Security Compliance in Agile Software Development: A Systematic Mapping Study

Fabiola Moyón[†], Pamela Almeida[*], Daniel Riofrío[*], Daniel Mendez[‡] and Marcos Kalinowski[§]
[†]Technical University of Munich (TUM) and Siemens CT. Munich, Germany.
Email: fabiola.moyon@tum.de
[*]Universidad San Francisco de Quito (USFQ). Quito, Ecuador.
Emails: pealmeida@estud.usfq.edu.ec, driofrioa@usfq.edu.ec
[‡]Blekinge Institute of Technology and fortiss GmbH, Karlskrona, Sweden.
Email: daniel.mendez@bth.se
[§]Pontifical Catholic University of Rio de Janeiro. Rio de Janeiro, Brazil.
Email: kalinowski@inf.puc-rio.br

***Abstract*—Companies adopting agile development tend to face challenges in complying with security norms. Existing research either focuses on how to integrate security into agile methods or on discussing compliance issues of agile methods but independently of the regulation type, in particular of security standards. A comprehensive overview of this scattered field is still missing and we know little about how to achieve *security compliance in agile software development*. Existing secondary studies (mapping studies and literature reviews) analyze publications on secure agile development, but they do not analyze implications of security standard compliance, e.g., integration of specific standard requirements or compliance assessments. To close this gap, we report on a systematic mapping study. Starting with a set of 2,383 papers, our work distills 11 relevant publications addressing security compliance in agile software development. With this study, we contribute by describing the maturity of the field, as well as domains where security compliant agile software engineering was investigated. Moreover, we make explicit which phases of a secure development process are covered by the field and which agile principles are analyzed when aiming at compliance with international security standards, country-specific security regulations, industry-specific security standards, and other well-known security frameworks.**

***Index Terms*—Systematic Mapping Study, Secure Software Engineering, Security Compliance, Agile Software Engineering**

I. Introduction

Agile methodologies are being applied not only for traditional systems, but also to develop critical systems where compliance with standards and regulations is a strong driving force [1], [2]. In fact, companies are often forced to deal with a competitive environment where customers' needs evolve and change rapidly. Agile methods and their customer-centric focus promise to be the key for such environments [3]. Hence, by shortening the development life-cycles and keeping a simple design, companies hope to have early feedback and adaptation cycles.

One of the challenges we focus on in our research is how to achieve compliance with security standards, against which companies are often measured, when employing agile software development (ASD) approaches. Although, previous studies (both primary and secondary studies) exist on the general notion of compliance in ASD and security in ASD, contributions still treat both fields in isolation: either compliance is analyzed independently of the type of regulation [4] or security aspects are integrated *ad-hoc*, i.e. they are not derived from specific security standards [5]. This renders the understanding on how to achieve security compliant agile development environments (and products) cumbersome.

In particular, existing contributions miss a comprehensive view on both, where security standards describe a broad notion of security that protects product (technology aspect) and its development life-cycle (process and people aspects) [6]. Our analysis therefore goes beyond technological aspects and relates more to agile values like team collaboration and working product flow. This fits well with a major goal of secure software engineering [7] which is not yet achieved for agile and lean software development [8].

In this paper, we report on a secondary study yielding 11 selected publications that treat security standards in context of ASD. Despite appearing to be low in number, in first sight, we deem the secondary necessary already at this stage to allow to synthesise the isolated fields early on. A particular focus of our study is to gain a quantitative overview of the publication landscape and its characteristics (with a systematic mapping study) which we use to distill relevant publications for a more detailed analysis. The overall objective is to contribute to a better understanding about the contemporary state of reported evidence in the field of *security compliance in ASD*. This allows us to discuss possible directions for researchers and implications for practitioners.

II. Background and related work

In the following, we introduce the relevant background in security compliance before discussing related work. We assume the reader is familiar with the notion of agile software development including the agile values as described in the agile manifesto [3] as well as in specific methodologies and practices such as Scrum [9] or the Scaled Agile Framework (SAFe) [10].

A. Background

Compliance requirements are stated in: standards, regulations and/or frameworks. In the following, we briefly describe the security compliance norms that are mentioned by the selected publications:

1) International Security Standards:

ISO 15408 Common Criteria (CC) treats specifications and evaluation of security attributes on software products [11].

ISO/IEC 62443-4-1 describes a secure development lifecycle for products used in industrial automation and control systems [12].

ISO 21827 Secure Systems Engineering Capability Maturity Model (SSE-CMM) consists of base practices, used as a checklist for measuring an organization's capability to develop secure systems [13].

2) Local Security Regulations:

VAHTI is a Finnish security regulation for information systems that connect to government systems. It is based on ISO/IEC 27001/2 Information Security Management System and US Laws: SOX and HIPPA [14]. SOX enforces internal control for accounting information [15] while HIPPA concerns to patient information protection [16].

NIST-800 is a US framework for improving security of critical infrastructure. It was released in 2014 and updated continuously [17].

3) Industry-specific Security Standards:

PCI-DSS (Payment Card Industry - Development Security Standard) provides requirements to securely store, process or transmit cardholder data or sensitive authentication data [18].

4) Other security best practices and frameworks:

Microsoft Secure Development Lifecycle SDLC is a secure software development process to help developers resist attacks by addressing common security activities [19].

Cigital Touchpoints is a process that improves product quality by introducing security engineering activities like code review, risk analysis, security and penetration testing, abuse case development and security requirements [20].

CLASP (Comprehensive, Lightweight Application Security Process) provides best practices, activities and project roles to help software-development teams to build security in an structured, measurable and repetitive way.

[21].

SAMM (Software Assurance Maturity Model) is an open framework to evaluate software security practices, build an assurance program, track improvements and define and measure security-related activities [22].

B. Related Work

To the best of our knowledge, there are no secondary studies on security compliance in agile software development which would allow to provide a big picture of the state of reported evidence. Nonetheless, there are systematic mapping studies and systematic literature reviews on security in agile. Despite contributing valuable insights to the field, security standards and regulations are not in scope in terms of their compliance aspects.

In our own research, we identified 76 relevant papers referring to security in agile development (please refer to figure 1). Among them, the following secondary studies are relevant to our own study described here.

Villamizar et al. [23] conducted a systematic mapping study on security in agile requirements engineering published in 2018. They identified 21 papers and 5 types of solution options to handle security requirements, and they discussed limitations in each approach.

Ouestali et al. [24] conducted a systematic literature review of the challenges of developing secure software using the agile approach in 2015. Ten papers were identified and 20 challenges related to agile development or developing secure software were identified. The study identifies the validity of security management challenges and security assurance and calls for researchers to address these challenges. Even in this study compliance was not in their scope.

Mohan and Othmane [25] conducted a mapping study on Security in DevOps. They identified 5 publications and 3 presentations, and they described security activities, process automation, and security problems throughout the process. They stated that SecDevOps implies team collaboration and encourages researchers to address the identified challenges. Security compliance is identified as one particular challenge, but no specific standards were mentioned.

In conclusion, our study extends existing literature by bringing the attention to the community in terms of adopting secure software development as part of an agile practice that generates products compliant with standards or government regulations

III. Study Design

In this section, we describe our research questions and the techniques applied to search, classify, and select relevant papers for further analysis. Our resulting data set is the product of filtering and analyzing contributions that treat security standards and government regulations in terms of achieving security compliance with such regulations throughout an agile product life-cycle. This undoubtedly reduced the universe of papers considered in this study.

We followed proven techniques for mapping studies like the methodology proposed by Kitchenham [26] and pragmatical recommendations for data collection and classification proposed by Kuhrmann et al. [27]. In addition, we consider as our guide the reporting structure presented by Villamizar et al. in a related study [23].

A. Research Objectives and Questions

To describe the current state of reported evidence, we analyze the context of current publications, how they deal with security standards, and the implications for agile methods. To steer our investigation, we formulate the following research questions (RQ):

RQ1. What is the profile of contributions referring to security compliance in agile software development? This question analyses the publications' context, their maturity and application area. We identify the involved research and practitioners groups as well as the domains and industry sectors which the research aims at. Finally, we determine how mature the research field is.

RQ2. Which aspects of security compliance are referred to by the publications?
This question aims at identifying which security standards are analyzed in relation to agile methods. We mapped contributions on known phases of a secure development process and defined to which extent these phases are covered. In addition, we determined if contributions focus solely on the integration of compliance requirement or also on assessment aspects.

RQ3. Which aspects of agile software engineering are analyzed from the perspective of security compliance?
This question considers the agility perspective. First, we want to understand which agile methods are in scope of the contributions including if they focus on scaled or on large-scale environments. Secondly, since security standard objectives are perceived as contrary to agile values (e.g. regarding contracting or documentation), we analyze if publications describe ways to fit standard requirements with agile values.

B. Search Strategy

We defined a set of keywords related to our research topic. We profit from the experience of the authors to determine through discussions the wording and establish 10 specific keywords to be used in the search strings: *continuous*, *agile*, *security*, *compliance*, *development*, *requirements*, *scrum*, *kanban*, *extreme programming*, *secure*. Our group of authors consists of people with experience in security compliance and agile software development. In particular, the second and the fourth author helped to provide the background in security compliance and in the development of empirical research methodologies.

The keywords aim at broadening the search scope within reason. In fact, different approaches are considered, e.g. *agile* is a term used by practitioners but *continuous* is often used by researchers to refer to the same practice. Hence, we included well known agile methodologies such as *Scrum*, *Extreme Programming* and *Kanban*. We prepared a set of logically connected search strings to capture the essence of our research (please refer to Table I). These search strings were tested in *Scopus*. These trials were used to verify the strings accuracy and to determine a preliminary number of hits related to the domain of our study, if any.

For our study, we selected digital libraries specific for software engineering. Such libraries were also proposed by similar studies [23], [27], namely: *IEEE*, *ACM* and *SpringerLink*.

The primary search results were ordered by relevance and the first 100 results from each search string where selected for subsequent analysis. After 100 hits, the results did not relate to the study field.

C. Relevant Papers Selection

Applying the search strings yielded 2,383 papers, not necessarily unique ones, that contain the search strings in titles, abstracts, and/or keywords. An overview of the study selection is summarized in figure 1. Subsequently, we describe the selection steps.

TABLE I
SEARCH STRINGS

N^o	**Search String**
1	(continuous OR agile) AND security AND compliance
2	secure AND development AND compliance
3	continuous AND security AND agile AND development
4	agile AND requirements AND security
5	security AND (scrum OR extreme programming OR kanban)
6	security AND scrum
7	security AND extreme programming
8	security AND kanban
9	security AND agile

To this dataset, we applied filters; first, based on the exclusion and inclusion criteria 1 and 2 shown on Table II, and secondly, by removing papers according to the exclusion criteria 3-8. The size of the set was reduced to 914.

TABLE II
INCLUSION AND EXCLUSION CRITERIA.

N^o	**Inclusion (I) / Exclusion (E)**	**Criterion**
1	I	Title, Keywords or Abstract related to security compliance for agile software development
2		The paper is related to computer science or software engineering.
3	E	The paper is not written in English
4		Title or Keywords are not related to security compliance for agile software development.
5		The paper's full text is not available to download.
6		The paper is not included on the first 100 results of the query
7		Paper occurs multiple times in the result-set.
8		The paper is a workshop summary.

We then *voted based on title and abstract* (see Fig. 1). We either voted for inclusion or removal of the single papers based on the information provided by the tittle and the abstract. The tool contained the whole set of publications and presented individual publications randomly to voters to prevent bias. Only those papers found in divergence were presented for validation. Two authors made an initial vote, later the topic's most experienced author voted to decide about the paper's relevance to yield a majority vote. A set of 98 papers was selected through majority voting. This set includes different clusters related to the voting criteria detailed on Table II.

At this point, a second filtering based on title and abstract took place (see Fig. 1). Three authors reviewed each cluster of publications. After this revision, 10 publications were consider relevant to the field of security compliance for agile software development [P1]–[P10].

Finally, we performed snowballing on references of these 10 publications, identifying 5 possible relevant papers ([28]–[31] [P11]). These papers' abstracts, full-text bodies, and conclusions were then reviewed. Four of them were discarded eventually, because either they refer security compliance but not agile practices or they considered compliance in agile but

TABLE III
VOTING CRITERIA

N^o	**Voting Criteria**
1	Security Compliance for Agile Software Development
2	Security Compliance in Software Development
3	Security Standard
4	Security and Software Development
5	Security and Agile Software Development

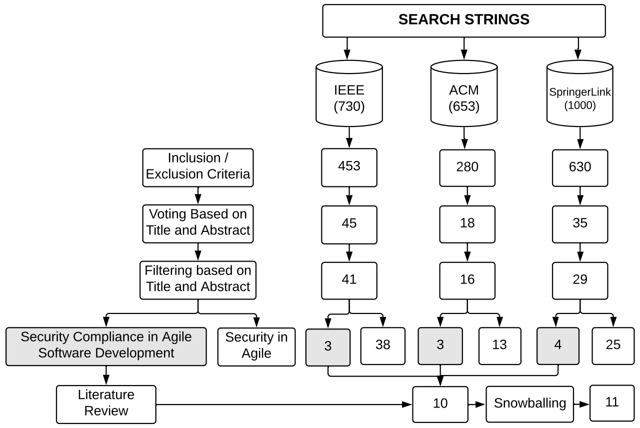
SEARCH STRINGS
IEEE (730)
ACM (653)
SpringerLink (1000)
Inclusion / Exclusion Criteria
453
280
630
Voting Based on Title and Abstract
45
18
35
Filtering based on Title and Abstract
41
16
29
Security Compliance in Agile Software Development
Security in Agile
3
38
3
13
4
25
Literature Review
10
Snowballing
11

Fig. 1. Study Selection Process. In gray, papers filtered for our study.

not in the area of security. Snowballing thus resulted in one additional paper which we added to the final result set [P11].

The final set consisted of **11 *selected publications*** referring specifically to security norms and ASD.

D. Data Extraction and Classification Scheme

From each of the 11 selected papers, we reviewed titles, abstracts and conclusions to extract data and answer research questions. Further detail and the relationship with our RQs can be found in Table IV.

TABLE IV
DATA EXTRACTION FORM

Information	**Description**
Study Metadata	Include the paper title, abstract, information about authors, venue, type of contribution and year of publication.
Approach Description	Short description of the approach.
(RQ1) Profile	Context, research type facets and application area of each publication.
(RQ2) Aspects of Security Compliance	Identify security standards, other frameworks, compliance perspective and the phases of SDL considered.
(RQ3) Agile from a Security Compliance Perspective	Agile methods, agile context and research type facets.

IV. STUDY RESULTS

The selection process covers 11 relevant papers on security compliance in agile software development. Table V presents a chronological summary of each paper with a summary of their contributions (kindly refer to Appendix A for their references).

TABLE V
RELEVANT PAPER CONTRIBUTIONS.

Year	**ID**	**Contribution**
2004	[P2]	This paper addresses how XP takes into consideration activities and security requirements engineering derived from standards.
	[P9]	This paper examines how conventional security assurance adapts to agile methodologies for intensive software systems development. It classifies software security assurance methods and techniques with respect to the impact into agile development.
2011	[P11]	This paper presents an agile development process based on professional impressions about security activities in security engineering processes. It includes those activities that do not block significantly the agile process.
2013	[P6]	This paper compares CLASP, Microsoft SDL, Cigital Touchpoints and Common Criteria with agile processes and applies a survey in order to understand a professional perspective about current security practices in agile software development.
2015	[P3]	This paper enhances Scrum, XP and Kanban methodologies using Microsoft SDL elements. These enhanced methods are evaluated for compliance with respect to VATHI requirements.
2016	[P8]	This paper proposes a strategy to achieve standard compliance and trust in the use of DevOps environments.
	[P4]	This paper reports experiences in agile development for secure regulated environments and it identifies best practices, and also, obstacles in a use case.
2017	[P7]	This paper presents a method, AgileSafe, in order to select agile practices for software development projects limited by assurance requirements resulting from safety and/or security standards.
2018	[P1]	This paper proposes a method for continuous security compliance that maps security requirements to an agile model.
	[P10]	This paper proposes a framework in order to align security engineering with software development. It suggests ways to overcome potential difficulties during the alignment.
2019	[P5]	This paper proposes an evaluation methodology for secure agile processes based on SAFe and IEC 62443-4-1.

We analyzed these relevant papers in detail. The analysis sheet is made available in our online material at https://doi.org/10.6084/m9.figshare.11984259.

The following items summarize the answer to the RQs while a discussion of implications is given in the next section.

A. RQ1. What is the profile of contributions referring to security compliance in agile software development?

1) What is the maturity level of research in security compliance for agile software engineering?: As a markup to determine the maturity of the field, we rely on the *research type facets* as described by Wieringa et al. in [32]. Most publications are of philosophical nature [P2], [P3], [P6], [P9], [P10] distributed along a decade (see Fig. 2). In other words, these papers discuss already existing concepts from different angles. Solution proposals in turn, namely the papers [P1], [P5], [P7], [P11], appear from 2011 until the current date. Opinion pieces, [P8], and evaluation reports, [P4], are rather isolated. Experience reports are yet completely missing. This indicates to a rather low state of evidence where scholars and industry practitioners would otherwise discuss experiences from applying certain approaches and techniques or general challenges encountered in this field.

2) Which type of contributions exist in the field?: Relevant contributions were published from 2004 to 2019 (see Table

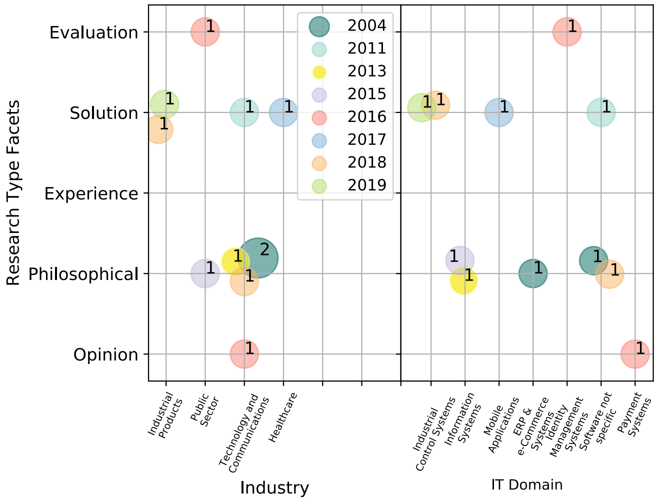

Fig. 2. Selected publications according year, maturity, industry and domain.

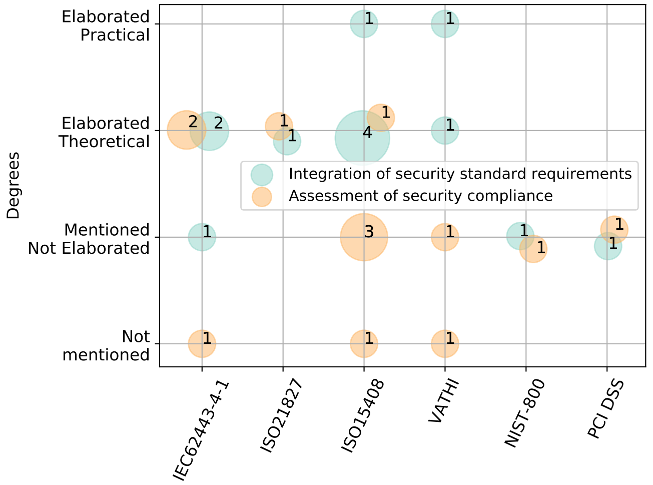

Fig. 3. Selected publications according to security norm, compliance perspective and degree of analysis.

V). Main locations of researchers affiliations are: Finland [P3], [P4], [P10], Sweden [P6], [P8], [P11], and Germany [P1], [P5].

The field captures the interest from both researchers and practitioners. However, most of publications contain only research affiliations, one paper contains only industry affiliations [P8], and 3 have ties to both industry and academia [P1], [P5], [P7].

As compliance is rather an industry-dependent issue, we also analyzed the related industry sectors (see Fig. 2). Technology and Communications industry is the context for 6 papers [P6]–[P11], Industrial Products [P1], [P5] and Public Sector [P3], [P4] have also more than one publication each.

B. RQ2. Which aspects of security compliance are referred to by the publications?

1) Which security norms are analyzed for publications referring to agile development?: Relevant publications refer to several security norms, although through the content, they do not elaborate on all of them. We excluded from the analysis those security norms that are described only as related content, e.g. ISO 27001, which although could be considered to be the top security norm. We identified 10 security norms distributed as: international standards, local regulation, and industry-specific standards and security frameworks, see table VI. The ISO 15408, also known as Common Criteria (CC), is the international standard that most publications address considering both philosophical and solution proposals (see Fig. 4). The IEC 62443-4-1 standard for secure product development in industrial systems gets the second place with existent solution proposals. The only mentioned local regulation is VATHI while Microsoft SDLC is the preferred security framework.

We analyzed in more detail a security compliance perspective (see Fig. 3). Either publications refer to the integration of compliance requirements or they describe aspects of security compliance assessment. Two papers elaborate and describe pragmatically how to integrate security standard requirements [P4], [P11] while 7 discuss it from a theoretical perspective [P1]–[P3], [P6], [P7], [P9], [P10]. None of the publications describe pragmatic ways to approach security compliance assessments in agile software development, 3 papers provide theoretical descriptions only [P2], [P5], [P7], and 3 do not mentioned it specifically [P1], [P4], [P6], among them the only evaluation paper [P4]. The rest of publications mention this aspect, but do not elaborate further.

TABLE VI
OVERVIEW OF SECURITY NORMS REFERRED BY RELEVANT PUBLICATIONS

Type	Norm	Relevant Papers
International Standard	ISO 21827 ISO 15408 ISO/IEC 62443-4-1	[P2] [P2], [P6], [P9]–[P11] [P1], [P5], [P7]
Local Regulations	VAHTI NIST-800	[P3], [P4] [P8]
Industry Specific Standards	PCI/DSS	[P8]
Other Security Best Practices	Microsoft SDLC OWASP SAMM CLASP Cigital Touchpoints	[P3], [P6], [P10], [P11] [P10] [P6] [P6], [P11]

2) Which stages of a standard secure development life cycle are analyzed by agile software engineering publications?: To answer this question, we mapped publications to the phases of a secure development life-cycle. Phases with more publications may be recognized as the challenging areas. The phases correspond with the practices of the IEC 62443-4-1 standard for secure product development (c.f. [12]). We changed the standard practice Security Guidelines for Security Hardening. This phase then refers to those activities that ensure secure configuration of development and deployment environment. Such aspects were mentioned by publications [P4], [P10], [P11] as part of agile development but are also part of a complete cycle to deliver a working product. The analysis of phases occurs independently of the security norm since the norms refer to different aspects of the life-cycle, e.g., the

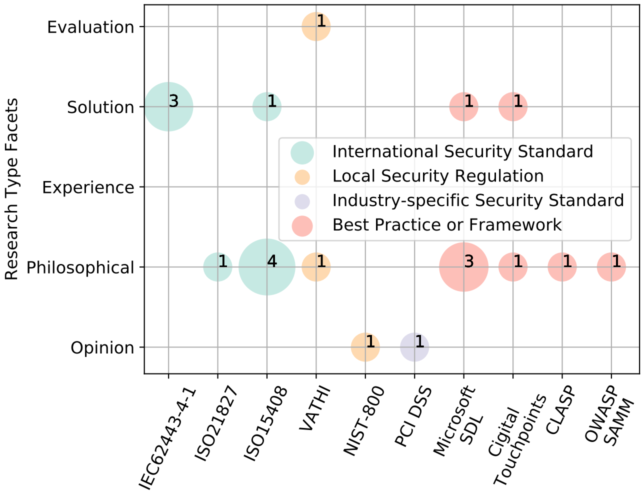

Fig. 4. Security norms that are depicted by the selected publications.

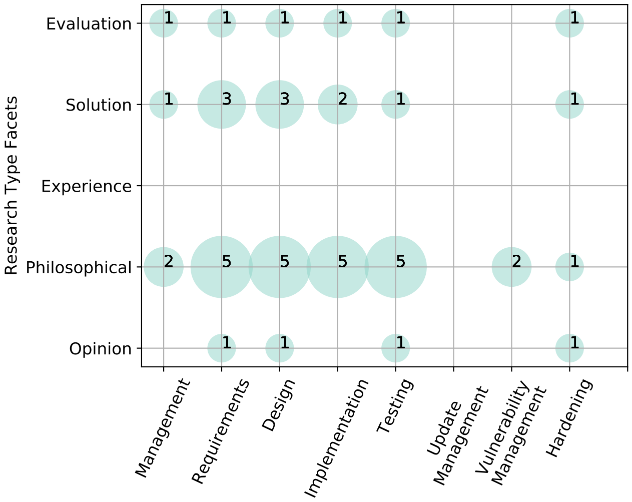

Fig. 5. Phases of a secure development life cycle that are addressed by the selected publications.

62443-4-1 refers to a secure development life-cycle while the ISO 15408 refers to aspects of the product.

Security requirements and design have the highest number of publications (see Fig 5) [P1]–[P4], [P6]–[P11]. Publications refer to how to cope with typical demands of security norms like threat modeling, security architecture design e.g. including them as user stories.

For *secure implementation*, publications describe agile methods as very compatible. Practices like peer-review are adapted to include in the review a security expert. They have the task to check code for secure coding aspects while transferring knowledge to developers. Evidence of these tasks are relevant to prove compliance [P4], [P10], [P11].

In relation to *security testing*, publications describe automation as a good opportunity to cover compliance testing requirements [P8], [P11]. However, how and when in the process to generate testing plans are not described.

Phases like *security management, update and vulnerability management* lack attention. Some publications include activities like *security training* or *incident response planning*, but do not elaborate specifically on the topic. The phase of *security hardening* although mentioned, is not completely depicted. Publications limit to mention that related activities are compatible with agile methods like Scrum or XP.

C. RQ3. Which aspects of agile software engineering are analyzed from the perspective of security compliance?

1) Which agile values are in scope of publications on security compliance?: All of the relevant publications refer to that security compliance influences the ability to generate a *valuable product increment* (see Fig. 6). Since security compliance requirements are part of customer needs, satisfying this aspect implies value for the customer. Six publications describe the influences of security compliance into the aim of *decreasing cycle times* [P2]–[P4], [P9]–[P11]. The main concern is that security compliance requires documented evidence which increases the delivery time. Teams need extra effort to generate such documentation or even to involve security experts and assessors to evaluate the accuracy of the documentation. Five publications describe how security compliance influences *team collaboration*. This includes involving security experts to analyze or better refined compliance requirements [P1]–[P4], [P11]. Team ownership of security is also mentioned. Suggestions are specific training or having a secure developer in Teams to do code review or support specific compliance demands like threat modeling. 4 publications mentioned *dealing with change* although do not elaborate on them [P2], [P9]–[P11].

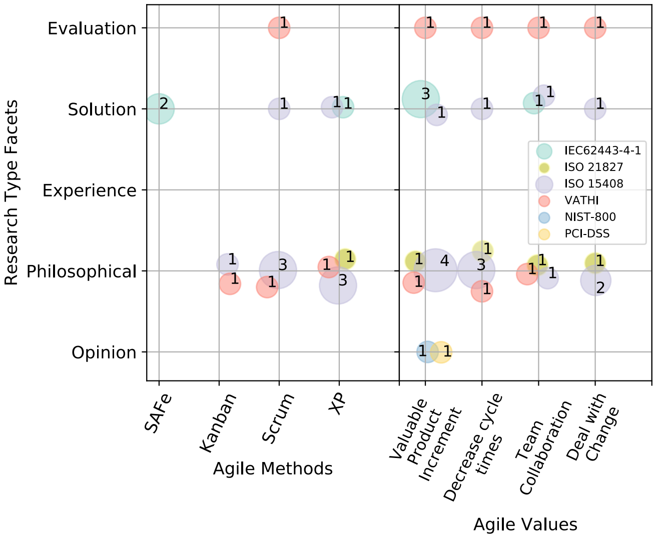

Fig. 6. Security norms and relation to agile methods and values.

2) Which are the agile methods addressed by publications on security compliance?: Relevant contributions describe the following as the agile methods that either are adapted or have adopted security compliance: Scrum [P3], [P4], [P6], [P9]–

[P11], XP [P2], [P3], [P7], [P9]–[P11], Kanban [P3], [P9] and the Scaled Agile Framework [P1], [P5]. Although not an agile methodology in the nearer sense, one publication [P8] refers to security compliance in DevOps, often seen as an evolutionary step in agile software development. Scrum, probably the most popular agile approach, can be compliant with VAHTI and the ISO 15408 standard (see Fig. 6). XP can be compliant with the previous norms plus ISO 21827 and ISO/IEC 62443-4-1. The Scaled Agile Framework can be compliant with IEC 62443-4-1.

V. Discussion

Our results show that publications in the field of *security compliance in agile development* are still rare in comparison to contributions referring to security in ASD (where security aspects are not based on requirements from security norms). After the voting process and forward snowballing (see Fig. 1), we identified in total 11 publications describing security aspects as stated by security standards, regulations, or best practices & frameworks. In the following, we summarize key points, extracted from the analysis of the selected publications.

Security compliance in ASD is a concern for practitioners and researchers since a few years after the appearance of the manifesto [3]). In 2004, authors started naming difficulties in the field ([P2], [P9]). By 2011, a first solution appeared [P11].

Authors describe two approaches to achieve security compliance in ASD: either security norms requirements are adapted to fit agile methods, e.g. [P1], [P9], or agile methods are adapted to fulfill security norms, e.g. [P2], [P7]. The only *evaluation* paper [P4] followed the first approach.

Reported evidence is still rare and lacks so far experience reports on the current state of the practice or that would otherwise underline any empirical figures to the adoption of techniques or approaches.

Authors focus more on describing how to integrate standard requirements than in how to assess their compliance or achieve certification. Few assessment publications are limited to theoretical discussions e.g. [P5], therefore experience reports are missing.

Challenges in the field are grouped into: problematic development areas and how to keep agility. The problematic development areas seem to be the same as for security in linear processes. Most publications concentrate on the phases of Requirements, Design, Implementation, and Testing (see Fig. 5). In these stages, most of the security compliance evidence is generated, e.g. security requirements baselines, threat models, secure coding standards, or security test plans. In the context of agile development, strict and long documentation is an issue. Moreover, there are phases not well covered like security management, updates, and vulnerability management. For practitioners, this is relevant as they need solutions to manage security issues in an agile way e.g. continuous vulnerability detection and patching.

Overall, publications conclude that **agile methods (Scrum, XP, and Scaled Agile Framework) are compatible with the requirements of security norms**. Authors see pitfalls for agility when generating security compliance evidence or integrating security experts and assessors. These aspects influence agile aims like reducing cycle times and team collaboration.

Finally, they also describe benefits like keeping the customer centric-view. Customers require compliant and secure products. The **product gains value through a seamless integration of security standard requirements into the agile process** (starting with the backlog e.g. adapting user stories).

Threats to Validity

As every study design, also this bears certain threats to validity for which we identified countermeasures.

First, our secondary study (as any other) might have a certain incompleteness in the search results; in our case, because we limited the results to the first 100 library results. To minimize this threat, we developed the search strings iteratively and adopted them to the libraries, and, moreover, we used a hybrid search strategy complementing library searches with snowballing.

We are confident in having analyzed key contributions. A possible threat affecting the reliability stems from selection bias as some results have been co-authored by authors of the study at hands. We mitigated this threat by having the first, independent author performing the overall data collection and having three authors do the voting procedure including a majority vote approach.

As a final measure, we also disclose our data sets to increase the transparency and reproducibility. This disclosure also strengthens the external validity in the long-run as other researchers not involved in our study can further update and replicate our secondary study.

VI. Conclusion

Publications in the field of *security compliance in agile software development* are rare. After processing an initial subset of 2,383 papers, we identified and analyzed 11 contributions that specifically treat security norms and compliance to them (see Fig. 1).

Contributions do not yet describe practical strategies on how to assess compliance but concentrate mostly on theoretical discussion, with limited empirical evidence, on how to integrate security norms requirements into agile methods.

To describe the field, this study analyzed not only its maturity, but also which aspects of security compliance and agile development are described by publications. This improves the knowledge of the field both for researchers and practitioners.

After mapping the selected publications according: research type, security norm, agile method and related agile value (see Fig. 6), results show interest into adapting Scrum, XP, the Scaled Agile Framework and Kanban to comply with security norms of the following clusters: international security standards (the 62443-4-1, the 21827 and the 15408), country security regulations (VAHTI, NIST800), industry-specific security standards (PCI-DSS) and well-known security frameworks (Microsoft SDLC, CLASP, OWASP SAMM, Cigital's Touchpoints).

Security compliance relates to all agile values. Compliant software satisfies customer needs and pose advantages with competitors, therefore aligned with *delivering a valuable product increment* and *dealing with change*. On the other hand, challenges arise to *decrease cycle times* due to a huge effort to achieve compliance and assess compliance evidence.

Our study results suggest possible directions for researchers and practitioners. Researchers are encouraged to address evaluation studies, i.e. to describe benefits/drawbacks of the applicability of their proposed solutions. Practitioners can gain an overview of this difficult and still fuzzy field, and they could further contribute to it with experience reports describing challenges and success stories. One aspect can be to gain clarity on regulators opinions about the validity of compliance evidence. For both, researchers and practitioners, this study mainly reveals the need to explore not only the integration of security standard requirements but also how to assess security compliance.

As further work, we consider to explore non-scientific literature. Our hope is to find in particular practitioner reports describing their experience on dealing with regulators and compliance gap assessors. Moreover, we look forward to develop such experience report in the context of the relevant methods here stated.

APPENDIX A. LIST OF STUDIES SELECTED FOR THE REVIEW

[P1] F. Moyon, K. Beckers, S. Klepper, P. Lachberger, and B. Bruegge, "Towards continuous security compliance in agile software development at scale," in *4th International Workshop RCoSE*. Sweden: ACM, 2018.

[P2] J. Wäyrynen, M. Bodén, and G. Boström, "Security Engineering and eXtreme Programming: An Impossible Marriage?" in *Extreme Programming and Agile Methods-XP/Agile Universe*. Germany: Springer, 2004.

[P3] K. Rindell, S. Hyrynsalmi, and V. Leppänen, "A comparison of security assurance support of agile software development methods," in *Proceedings of the 16th International on CompSysTech*. Ireland: ACM, 2015.

[P4] K. Rindell, S. Hyrynsalmi, and V. Leppanen, "Case Study of Security Development in an Agile Environment: Building Identity Management for a Government Agency," in *11th ARES*. Austria: IEEE, 2016.

[P5] S. Dännart, F. Moyón, and K. Beckers, "An Assessment Model for Continuous Security Compliance in Large Scale Agile Environments: Exploratory Paper," in *Advanced Information Systems Engineering*. Switzerland: Springer, 2019.

[P6] T. Ayalew, T. Kidane, and B. Carlsson, "Identification and Evaluation of Security Activities in Agile Projects," in *Secure IT Systems*. Germany: Springer, 2013.

[P7] J. Górski and K. Łukasiewicz, "Meeting Requirements Imposed by Secure Software Development Standards and Still Remaining Agile," in *Computer Network Security*. Switzerland: Springer, 2017.

[P8] J. R. Michener and A. T. Clager, "Mitigating an Oxymoron: Compliance in a DevOps Environments," in *40th COMPSAC*. USA: IEEE, 2016.

[P9] K. Beznosov and P. Kruchten, "Towards agile security assurance," in *Proceedings of Workshop on New security paradigms - NSPW*. Canada: ACM, 2005.

[P10] K. Rindell, S. Hyrynsalmi, and V. Leppänen, "Aligning security objectives with agile software development," in *19th International Conference on Agile Software Development-XP*. Portugal: ACM, 2018.

[P11] D. Baca and B. Carlsson, "Agile development with security engineering activities," in *2nd Workshop on SESENA '11*. USA: ACM Press, 2011.

REFERENCES

[1] R. Hoda, N. Salleh, and J. Grundy, "The Rise and Evolution of Agile Software Development," *IEEE Software*, vol. 35, no. 5, pp. 58–63, Sep. 2018.

[2] VersionOne, "13th Annual State of Agile Survey," May 2019.

[3] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, "Manifesto for agile software development," 2001. [Online]. Available: http://www.agilemanifesto.org/

[4] Z. Zhioua, Y. Roudier, and R. B. Ameur, "Formal Specification and Verification of Security Guidelines," in *22nd International Symposium on Dependable Computing (PRDC)*. New Zealand: IEEE, 2017.

[5] X. Ge, R. F. Paige, F. Polack, and P. Brooke, "Extreme Programming Security Practices," in *Agile Processes in Software Engineering and Extreme Programming*. Berlin, Heidelberg: Springer, 2007.

[6] E. Humphreys, "How to measure the effectiveness of information security," 2017. [Online]. Available: https://www.iso.org/news/2016/12/Ref2151.html

[7] K. Khan and M. Jaatun, *International Journal of Secure Software Engineering IJSSE*. IGI Global, 2017.

[8] B. Fitzgerald and K.-J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, 2017.

[9] K. Schwaber, "Scrum development process," in *Business Object Design and Implementation*. London: Springer London, 1997, pp. 117–134.

[10] D. Leffingwell, "Scaled agile framework," 2020. [Online]. Available: www.scaledagileframework.com

[11] International Organization for Standardization, "Information technology: Security techniques: Evaluation criteria for IT security 15408-1," 2014.

[12] International Electrotechnical Commission, "Security for industrial automation and control systems: Secure product development lifecycle requirements 62443-4-1," 2018.

[13] International Organization for Standardization, "Information technology: Security techniques: Systems Security Engineering : Capability Maturity Model® (SSE-CMM®)," 2008.

[14] Government and Ministries and Finnish Sweden english, "VAHTI," 2013. [Online]. Available: https://vm.fi/vahti

[15] S.-O. Act, "The Sarbanes-Oxley Act 2002," 2002.

[16] O. f. C. Rights (OCR), "Summary of the HIPAA Security Rule," Nov. 2009.

[17] National Institute Of Standards and Technology, *Special Publication 800-53 Information Security*, USA, 2011.

[18] PCI Security Standards, "Official PCI Security Standards Council Site," 2006. [Online]. Available: https://www.pcisecuritystandards.org

[19] M. Howard and S. Lipner, "The Security Development Lifecycle," p. 348, 2006.

[20] G. Mcgraw, "Software Security: Building Security In," 2006.

[21] D. Graham, "Introduction to the clasp process," Nov 2016. [Online]. Available: https://www.us-cert.gov/bsi/articles/best-practices/requirements-engineering/introduction-to-the-clasp-process

[22] OWASP, "Software Assurance Maturity Model (SAMM)," p. 72, 2017.

[23] H. Villamizar, M. Kalinowski, M. Viana, and D. M. Fernandez, "A Systematic Mapping Study on Security in Agile Requirements Engineering," in *44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. Prague: IEEE, 2018.

[24] H. Oueslati, M. M. Rahman, and L. b. Othmane, "Literature Review of the Challenges of Developing Secure Software Using the Agile Approach," in *10th International ARES*. France: IEEE, 2015.

[25] V. Mohan and L. B. Othmane, "SecDevOps: Is It a Marketing Buzzword? - Mapping Research on Security in DevOps," in *11th International (ARES)*. Austria: IEEE, 2016.

[26] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information and Software Technology*, vol. 55, no. 12, pp. 2049–2075, Dec. 2013.

[27] M. Kuhrmann, D. M. Fernández, and M. Daneva, "On the pragmatic design of literature studies in software engineering: an experience-based guideline," *Empirical Software Engineering*, 2017.

[28] H. Alaidaros, M. Omar, and R. Romli, "Towards an Improved Software Project Monitoring Task Model of Agile Kanban Method," 2018.

[29] W. H. M. Theunissen, D. G. Kourie, and B. W. Watson, "Standards and Agile Software Development," p. 11, 2003.

[30] B. Fitzgerald, K.-J. Stol, R. O'Sullivan, and D. O'Brien, "Scaling agile methods to regulated environments: An industry case study," in *35th (ICSE)*. USA: IEEE, 2013.

[31] K. Łukasiewicz and J. Górski, "AgileSafe a method of introducing agile practices into safety-critical software development processes," 2016.

[32] R. Wieringa, N. Maiden, N. Mead, and C. Rolland, "Requirements engineering paper classification and evaluation criteria: a proposal and a discussion," *Requirements Engineering*, 2006.