Systematic Mapping of Security Patterns Research

Yurina Ito, Hironori Washizaki, Masatoshi Yoshizawa, Yoshiaki Fukazawa
Dept. Computer Science
Waseda University
Tokyo, Japan
tsunko@fuji.waseda.jp, yoshizawa-masa@asagi.waseda.jp, {washizaki,fukazawa}@waseda.jp

Takao Okubo
Graduate School of Information Security
Institute of Information Security
Yokohama, Japan
okubo@iisec.ac.jp

Haruhiko Kaiya
Faculty of Science
Kanagawa University
Kanagawa 259-1293, Japan
kaiya@kanagawa-u.ac.jp

Atsuo Hazeyama
Dept. of Information Science
Tokyo Gakugei University
Tokyo, Japan
hazeyama@u-gakugei.ac.jp

Nobukazu Yoshioka
Grace Center
National Institute of Informatics
Tokyo, 101--8430, Japan
nobukazu@nii.ac.jp

Eduardo B. Fernandez
Dept. of Computer and Elect. Eng. and Comp. Science
Florida Atlantic University
Boca Raton, FL, USA
fernande@fau.edu

***Abstract*— Security patterns (SPs) are reusable solutions to security problems. We study here research papers that use security patterns to build secure systems or analyze the nature of security patterns. The goal of this paper is neither listing nor direct mapping of existing over 200 SPs but finding about how SPs are being investigated within research works to guide future research targeting SPs. Although the number of SPs has recently grown, two critical problems remain due to the diversity in the results themselves and how they are shared. First, it is unclear whether or not the field is actively growing. Second, the trends in SP research (e.g., research content and their modeling methods of SPs) are uncertain. To elucidate the current trends, herein we classify 30 works on SPs using a technique called systematic mapping (SM), which reveals the following characteristics. As the frequency of less common patterns (e.g., reference monitor) increases, the amount of practical research (e.g., experimental evaluations) also increases; Regardless of SPs to be dealt with, the most common SP modeling method is UML followed by other modeling methods for specific purposes, demonstrating the importance of modeling methods complementing each other; Currently one the most common research topics is applying SPs, suggesting that the demand for efficient and reliable techniques to applying SPs is high; Future studies should examine other SPs in addition to access control to handle various threats as well as to investigate the analysis/requirement and test phases; Accumulated knowledge on SPs should improve research requiring precise modeling.**

***Index Terms*—security patterns, systematic mapping, software patterns.**

I. INTRODUCTION

Patterns, which are packaged reusable solutions to recurrent problems under specific contexts, are important tools in software engineering. Patterns include design, analysis, and security patterns (SPs). For example, the appearance of design patterns has been one of the most important developments in software engineering [1]. Developers are expected to achieve efficient software development by reusing patterns [2]. SPs including concrete security patterns and abstract ones [3] capture successful secure designs in a generic form that can be applied or instantiated to produce solutions with well-defined properties [4] to stop threats or correct vulnerabilities. Because SPs incorporate the knowledge of security experts, they provide guidelines to improve confidentiality, integrity, and availability of software development. SPs are usually described in terms of Structure, Context, Problem, Solution, and Consequences [5].

SPs are used to realize security features, leading to a more secure system than a functionally equivalent system without patterns [6], and are intended for use by developers who are not security professionals [7][8]. They enable developers and engineers to recognize, with relative ease, known vulnerabilities in their design and potential solutions [47]. Several security patterns have been reported by practitioners and researchers, and there are lively and ongoing discussions about the discovery, documentation and application of security patterns [9].

We study here research papers that use security patterns to build secure systems or analyze the nature of security patterns. The goal of this paper is neither listing nor direct mapping of existing over 200 SPs but finding about how SPs are being investigated within research works to guide future research targeting SPs.

Although the number of SPs has considerably increased [10], two problems remain. First, it is unclear that whether or not SP research is a steadily growing field. Second, research trends (e.g., research contents and modeling methods of SPs) are unclear.

Because security is involved with almost all aspects, technical elements, and stages of systems and software, numerous organizations are investigating SPs from different perspectives. Thus, the research results and how the results on SPs are shared (e.g., software conferences and journals,

system conference, quality conference, etc.) vary tremendously, which makes it challenging for developers and operators to determine the most suitable techniques. To ensure security, it is desirable that every stage of development and operation accurately consider and use SPs. In practice, identifying areas where research on SPs is lacking is problematic, which makes studies beyond elementary technology and the research stage, such as SP groups, extremely difficult. Thus, SP research must be analyzed.

Herein we classify 30 papers[1] on SPs using Systematic Mapping (SM) to analyze the trends, where SP research indicates papers that use existing SPs but excludes papers that propose a specific SP as a new concept because they have been already studied (see IV. Related Work). The contribution of this paper is that the results should guide future SP research as well as assist engineers in applying SPs.

This paper is organized as follows. Section II explains the mapping process. Section III provides the mapping results and discussion. Section IV describes related work. Finally, Section V summarizes this paper.

II. Mapping Process

SM is an existing classification method [11] used to quickly and easily identify the coverage area of a research topic by preparing a visual summary. SM involves six steps: defining research questions, reviewing the scope, conducting a search, screening papers, keywording using abstracts, and data extraction and the mapping process.

We use SM to analyze the trends in SP research. Below each mapping step is described.

A. Defining Research Questions (RQs)

In SM, RQs narrow the topic of interest. SPs are used to implement security features and lead to a more secure system than a functionally equivalent system without patterns [6]. Recently, the number of SPs has increased [50]. However, to facilitate use of SPs, more SP research and a better understanding of the research trends are desirable. This study poses three RQs.

RQ1: Is SP research an active and steadily growing field?

RQ2: What are the current trends of SP research?

RQ3: What are the future prospects of SP research?

B. Reviewing the Scope

In SM, review scope specifies the number of potential targets (hits) used in the mapping. Here the review scope is limited to the top 50 results for a keyword search using two different search engines where the results are sorted by relevance. Because the results are not necessarily related to security patterns, we actually read the papers in the hits. Hence, we reviewed 100 papers on November 2014.

C. Conducting a Search

In SM, conduct search sets the parameters in the mapping. Here we define our search engines as ACM Digital Library (ACM DL) [12] and IEEE Xplore [13] because these are famous search engines, and "security pattern" is the keyword. We retrieved 50 papers by ACM DL and another 50 by IEEE Xplore. As a result, there was one paper hit in both search engines.

D. Screening Papers

In SM, screening allows the results to be roughly evaluated and those inconsistent with the RQs posed in the first step to be excluded. Here we employ four exclusion conditions:

1. Papers that cannot be referenced from the search engines (3 papers)
2. Papers not related to security patterns (32 papers)
3. Papers proposing a specific SP after reading them (34 papers)
4. Papers that were a hit in both search engines (1 paper)

Hence, a total of 30 papers are analyzed in SM. We attached the list of these papers as Table 1 to this paper.

E. Keywording Using Abstracts

In SM, facets are defined as the evaluation axes by reading abstracts and identifying keywords in the 30 papers. The facets aim to answer the RQs. Here we define nine facets due to the features of SP research:

F1: Year published:

Because the number of SPs has grown recently [10], the number of SP research papers may have also grown. Confirming this speculation will provide insight on SP research.

F2: Research content

Research on SPs covers diverse topics [4]. Hence, revealing what topics have been studied will provide insight on SP research.

F3: SP modeling methods

Because various SP modeling methods exist [9], revealing the most common ones might provide a new direction for SP research, such as unification and selection of appropriate modeling method according to objectives (and SPs).

F4: SPs to be dealt with

Because there are various SPs [10], revealing the most common SPs to be dealt with will provide insight on SP research and the applicability to actual problems.

F5: Experimental evaluations

For practical implementation, SP research should not only propose but also conduct experimental evaluations (e.g., case study according to an example). Thus, whether a work includes experiments is crucial for practical applications.

[1] We did not try to select papers based on quality; we just followed the SM process and exclusion criteria described in Section II. Moreover we were not trying to be comprehensive since the number of papers reviewed is limited to 100. We intend to enlarge the scope of the paper in a subsequent version.

Table 1. Entire classification results of SPs

Ref.	F1	F2	F3	F4	F5	F6	F7	F8	F9
[16]	2009	application	UML	RBAC, Authentication, Logger, Firewall, Authorization, IDS	yes	6	yes	(iv)	no
[17]	2014	application	UML	SoSpa (System of aspect-oriented Security design Patterns)	no	1 (set)	yes	(ii)	no
[18]	2008	modeling	Petri nets	Sandbox and Message Secrecy	no	2	no	(iii)	no
[19]	2005	case study	NL	Compartmentalization, Distributed Delegation pattern , Trust Partitioning, Unique Atomic Chunks, Secure Pre-forking, Chroot Jail	no	6	no	(ii)	no
[20]	2011	detection	UML	Single Access Point	yes	1	-	(iii)	no
[21]	2012	validation	NL	Authentication, Logger, Firewall, Authorization, Single Access etc. (details are in [8])	yes	35	yes	(ii)	no
[22]	2013	application	SAM	Logger, Authorization	yes	2	no	(vi)	no
[23]	2008	case study	UML	ReferenceMonitor, Authorization, Authentication, Firewall etc. (details are in [23])	no	18	yes	(vi)	no
[24]	2005	modeling	HL7, JAHIS	RBAC	yes	1	-	unknown	no
[25]	2010	other	UML	RBAC, Authentication, Authorization, Reference Monitor	yes	4	yes	(v)	no
[26]	2013	validation	UML	RBAC, Password Design and Use, Prevent SQL Injection	yes	3	no	(v)	no
[27]	2009	application	UML	Property Certification, Secure Match, Mutual Key PoK	no	3	no	unknown	no
[28]	2013	application	UML	RBAC	yes	1	-	(iv)	yes
[29]	2009	other	UML	Authorization	no	1	-	(i)	no
[30]	2012	application	UML	Active Replication	yes	1	-	(iii)	no
[31]	2007	detection	UML	Authentication, Secure Pipe	yes	2	yes	unknown	no
[32]	2008	case study	XACML	Authorization	yes	1	-	(iv)	no
[33]	2008	selection	GRL, Prolog	RBAC, Single Access Point, Checkpoint	no	4	yes	(v)	yes
[34]	2011	other	NL	Payment, Anti Cross Site Request Forgery (CSRF)	yes	2	yes	(v)	no
[35]	2008	application	NL	Sensor Encryption, Data Decryption, Grant-Based Access Control	no	3	yes	(iii)	no
[36]	2009	selection	NL	patterns described in [7]	yes	46	yes	unknown	no
[37]	2013	application	unknown	patterns described in [7]	yes	46	yes	(ii)	no
[38]	2011	application	UML	Authorization, RBAC, Multilevel Security, Reference Monitor, Password Design And Use	no	5	no	(ii)	yes
[39]	2012	application	UML	Authorizationzation, RBAC	yes	2	no	(vi)	no
[40]	2010	application	unknown	Authorization, Reference Monitor, Log/Audit	no	3	yes	(v)	no
[41]	2012	case study	unknown	Input Validation Vulnerability Tests, Force Exposure Tests, Malicious File Tests, Malicious Use of Security Functions Tests, Dangerous URL Tests, Audit Tests	no	6	no	(vii)	yes
[42]	2014	application	unknown	Computer-Oriented Security Patterns (COSPs)	no	1	-	(vi)	no
[43]	2010	application	DLS	Authentication	no	1	-	unknown	no
[44]	2010	other	NL	Authentication, ServiceAccessControl	no	2	no	(vi)	no
[45]	2008	application	unknown	Audit Interceptor, Secure Logger, Authentication	yes	3	no	unknown	no

F6: Number of SPs in a paper

As the number of SPs has increased [10], SP research should not only deal with single SP but also investigate multiple SPs although some conferences could have paper size limitation[2].

F7: Relationships between SPs

Because related patterns can solve large problems [14], whether a paper considers the relationships between SPs is crucial information.

F8: Phases that SPs are applied

Because security must be considered for all phases of the lifecycle [15], the phase that the SP is applied is crucial information.

F9: Tooling

If tooling is fully implemented, the results will be independent of the individuals conducting the research, allowing the maturity to be evaluated. However, tooling implementation can be difficult when SPs or SP research is abstract. Thus, it is important to clarify how much tooling is actually implemented.

[2] PLoP conferences usually do not have paper size limitation.

F. Data Extraction and the Mapping Process

This step of SM visualizes the results for further analysis. Firstly we reviewed each paper in detail, classified it into a certain category for each facet and documented the classification result into a data table. Secondly we calculated the frequencies of papers in each category on the data table and created several systematic maps. The results of this study are discussed in the next section.

III. Results And Discussion

Using SM, we collected 30 papers [16]–[45] on SPs and analyzed the trends. Table 1 depicts the classification results of the papers.

A. Results from a Single Facet

F1: Year published

SP research has been conducted in the past most decade (Fig. 1). For 2014, the number of papers is as of Nov. 2014. Moreover, search engines may need some extra time to index published papers so that the number for 2014 is incomplete.

F2: Research contents

Based on a survey of SPs [5], we divided the research content into seven categories. The results are as follows: 46% (14 papers) apply SPs, 13% (four papers) conduct case studies, 7% (two papers) provide specific models for SPs, 7% (two papers) detect SPs, 7% (two papers) validate SPs, 7% (two papers) select SPs, and 13% (four papers) focus on other topics. It should be noted that a case study indicates implementation of SPs in certain scenarios.

The most common category is applying SP, suggesting that there is a large demand for efficient and reliable techniques to implement SPs.

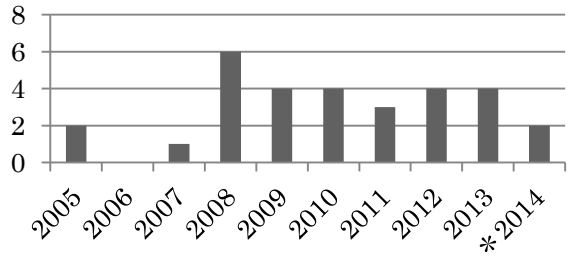

Fig. 1. Number of papers by year

F3: SP modeling methods

We divided the modeling methods into four categories. The results are as follows: 43% (13 papers) use UML, 20% (six papers) use a natural language ("NL"), 20% (six papers) use another language ("others"), and the modeling method is difficult to identify in the remaining 17% (5 papers).

The most common modeling method is UML, while "others" is the second most common. Others include Petri Nets [18], SAM (Serscis Access Modeller) notation [22], HL7 and JAHIS [24], XACML (eXtensible Access Control Markup Language) [32], GRL (goal-oriented requirement language) [33], and DSL (domain specific languages) [43], where each method has a specific purpose. Although modeling using a natural language mainly describes pattern characteristics, it is difficult to convert it into design and code. These findings demonstrate the importance of modeling methods complementing each other.

F4: SPs to be dealt with

Figure 2 plots the values when at least two papers examine the same SP. The three most common SPs are Authorization, RBAC, and Authentication. Access control is a common concrete SP in SP research and is likely in actual development. However, it needs to be further examined for other SPs in addition to access control to handle various threats. This may become a more common topic in the future. For example, access control could denote the combination of authorization and enforcement done by the reference monitor.

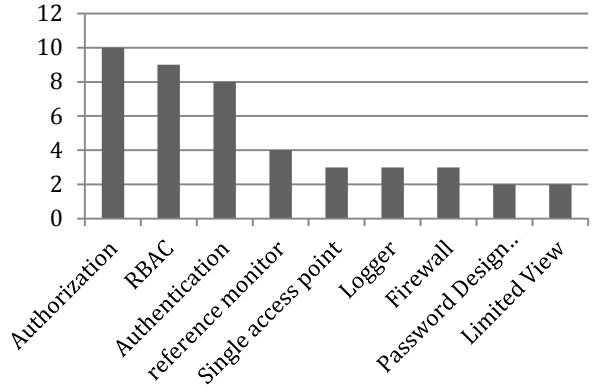

Fig. 2. Number of papers per each SP

F5: Experimental evaluations

Over half of the papers (53%) involve experimental evaluations. However, the other half only propose an approach; hence, the effectiveness and validity of half are untested. To strengthen the effectiveness and validity, researchers should conduct more experimental evaluations in the future.

F6: Number of SPs in a paper

Although some research uses more than 30 SPs, 47% use only one or two patterns. According to the facet F6 (Section II-E), future research could consider to address more SPs to handle various specific threats in a uniform way.

F7: Relationships between SPs

Only 40% of the papers consider relationships between SPs. According to the facet F7 (Section II-E), future research should consider relationships between SPs to handle larger problems.

F8: Phase where SPs are applied

We divided the development phases into categories: (i) analysis/requirement (one paper), (ii) design (five papers), (iii) implementation (four papers), (iv) analysis/requirement, design, and implementation (three papers), (v) analysis/requirement and design (five papers), (vi) design and implementation (five papers), and (vii) testing (one paper). The phases in the remaining six papers are unknown. Although there are fewer papers on the analysis/requirement and testing phases, the remaining phases are almost equally investigated.

F9: Tooling

Only 13% of the papers ([28][33][38][41]) involve tooling. Since SPs are know-how and knowledge, their handling is largely by human hands. To promote a

convenient treatment, tooling should increase and even be automated. This result indicates that SP research is still in its infancy in terms of automation.

B. Results from Facet Relations

The relationships between two or more facets are considered as bubble plots. We show only the characteristic results below where the number in the bubble denotes the number of papers.

F3 and F5

Figure 3 plots the relationship between F3 and F5. The variation in the plot indicates that whether an experimental evaluation is conducted (F5) is independent of the modeling method (F3). It could be because it is not always necessary to have formal and/or specific models for experimental evaluations; experiments could be conducted for SPs modeled even in natural language.

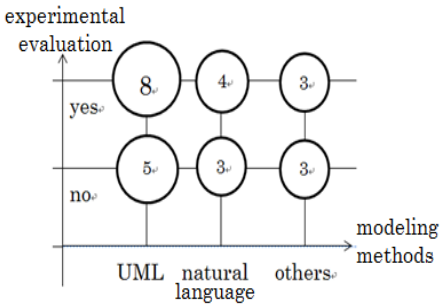

Fig. 3. F3&F5

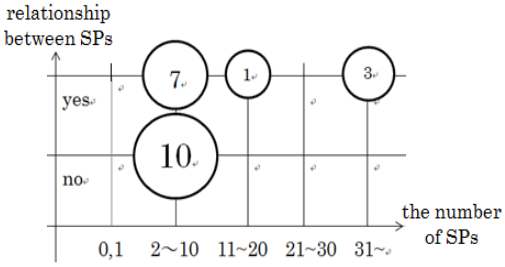

Fig. 4. F6&F7

F6 and F7:

Figure 4 shows the relationship between F6 and F7. Papers with a large number of SPs (F6) tend to consider relationships between SPs (F7). It seems quite natural since such papers focusing on SP relationships are likely to handle many SPs for explaining motivations and evaluations.

F3, F4, and F5:

Figure 5 shows the relations between F3, F4, and F5. On the F4 axis, papers with four or more SPs are arranged in descending order from left.

As shown in the upper part of Fig. 5 (F3 and F4), most SP research uses UML for modeling (F3) regardless of SPs to be dealt with (F4). It could be because UML is the most widely accepted formalism for the analysis and design of software [46]. Therefore, UML could be considered as SP modeling method first in further researches if there is no specific requirement for models.

As shown in the lower part of Fig. 5 (F4 and F5), research on less common patterns (right of F4 axis, e.g., reference monitor) is less likely to include an experimental evaluation (F5). It could be because the more patterns become popular, the more researchers are likely to pay attention to practical aspects of the patterns; and maybe vice versa. According to the facet F5 (Section II-E), whether experimental evaluations are part of the research is crucial to apply SPs in practice; encouraging researchers conduct experimental evaluations for less common SPs might result in more applications of these SPs.

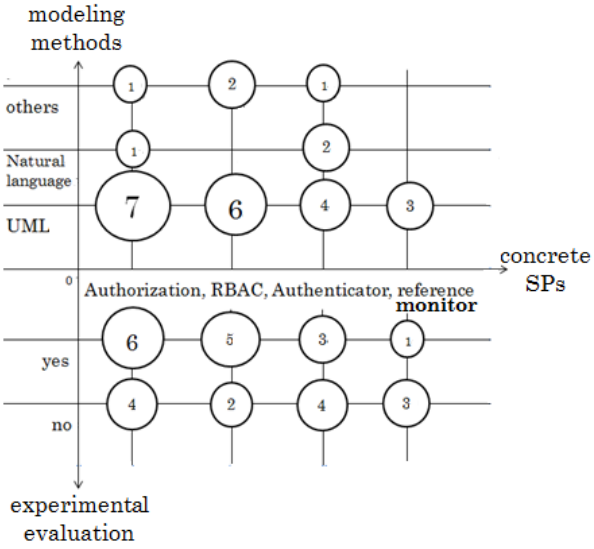

Fig. 5. Number of SPs in a paper

F1 and F5:

Figure 6 plots the relation between F1 and F5. For 2014, the number of papers is as of Nov. 2014. So the number for 2014 is incomplete. The number of experimental evaluations (F5) has increased recently (F1), indicating that research is shifting towards strengthening the effectiveness and validation of SPs.

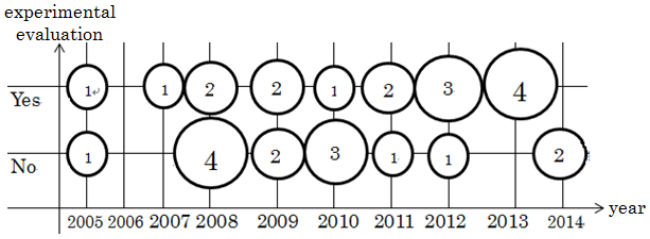

Fig. 6. F1 and F5

F3 and F6:

Figure 7 shows the relation between F3 and F6. Studies that contain many SPs (F6) tend to use a natural language

(F3) because it is difficult to precisely model many SPs due to lack of sharing reported results. Hence, accumulating knowledge should increase research requiring precise modeling.

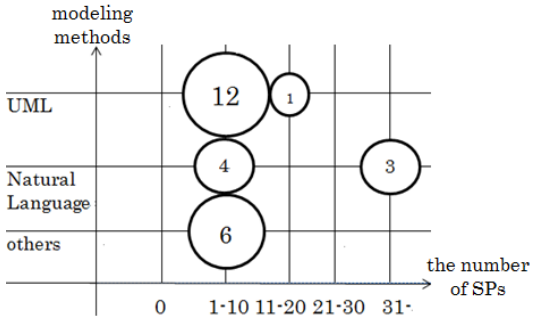

Fig. 7. F3 and F6

C. *Summary of Findings and Future Perspectives*

Our findings and future perspectives are as follows.

RQ1: Is SP research an active and steadily growing field?

SP research has been conducted constantly in the past decade (F1), confirming that the field is growing.

RQ2: What are the current trends of SP research?

The most common research topic is applying SPs (F2), demonstrating that efficient and reliable techniques to apply SPs are necessary to guarantee security.

The most common modeling method is UML (F3). "Others" where each method has a specific purpose is the second most common. To fully utilize each modeling method, it is crucial that modeling methods complement each other. Access control is a common concrete SP in SP research and is likely in actual development (F4). However, it needs to be further examined for other SPs in addition to access control to handle various threats.

About half of SP research involves experimental evaluations (F5). Because the effectiveness and validity of SPs are often untested, the amount of experimental evaluations should increase in the future. Similarly about half (47%) of SP research uses only one or two patterns (F6). To handle various threats in a uniform way, future research could consider to incorporate multiple SPs.

Although the facet F7 (Section II-E) states that future research should consider the relationships between SPs to handle larger problems, only 40% of research considers relationships (F7).

There are fewer reports on the analysis/requirement and test phases, but the other phases have a similar number of papers, suggesting that SPs are used in all phases (F8). To date, only 13% of the papers involve tooling (F9).

Tooling and automation promote the reasonable treatment of SPs. The lack of tooling indicates that SP research is in its infancy in terms of automation.

The relationships between the facets are important in determining the direction of SP research. This study reveals many interesting relationships. First, whether an experimental evaluation is conducted is independent of the modeling method (F3 and F5). It could be because it is not always necessary to have formal and/or specific models for experimental evaluations. Second, papers with numerous SPs tend to consider their relationships (F6 and F7). It seems quite natural since such papers focusing on SP relationships are likely to handle many SPs for explaining motivations and evaluations. Third, most research on SPs employs UML for modeling regardless of SPs to be dealt with (F3 and F4). It could be because UML is the most widely accepted formalism for the analysis and design of software; UML could be considered as SP modeling method first in further researches if there is no specific requirement for models. Fourth, research on more common patterns is more likely to include experimental evaluations (F4 and F5). It could be because the more patterns become popular, the more researchers likely to pay attention to practical aspects of the patterns; and maybe vice versa. Encouraging researchers conduct experimental evaluations for less common SPs might result in more applications of the SPs. Fifth, research involving experimental evaluations has increased recently (F1 and F5), indicating that research is strengthening the effectiveness and validation of SPs. However, studies using many SPs tend to use modeling methods in natural languages (F3 and F6) because modeling SPs precisely is difficult. As knowledge on SPs is accumulated, precise modeling should become easier, increasing the amount of research requiring precise models.

RQ3: What are the future prospects of SP research?

According to the answer of RQ2, there could be several future prospects aligned with the above-mentioned facets.

Although UML is the most widely accepted formalism, modeling methods (UML, natural language, etc.) should complement each other. Moreover, accumulated knowledge on SPs should improve research requiring precise modeling (F3).

Encouraging researchers conduct experimental evaluations for less common SPs (e.g., reference monitor) might result in more applications of the SPs. It needs to be further examined for other SPs in addition to access control to handle various threats (F4). For example, access control could denote the combination of authorization and enforcement done by the reference monitor.

There are only 40% of researches considering relationships among SPs (F7); future research should consider the relationships to handle larger problems. For example, pattern diagrams for describing SPs and relationships could be important.

Efficient and reliable techniques are necessary to apply SPs (F2). Moreover, SP research on analysis/requirement and test phases should increase (F8).

IV. Related Work

Although previous works have focused on SP classification, they differ from our study, which classifies SP research. One study classified SPs as quality analysis and coverage analysis after screening the literature [47]. Another, which aimed to identify the root causes of security violations, based the classification on security flaws along with other parameters to provide easy-to-use SPs [48]. In addition, a different study surveyed existing SP classifications [49].

V. Conclusion

Herein SP research is classified using SM to analyze trends. The results provide various insights on the future direction of SP research as we summarized in Section III-C. Our results should improve SP research and the effectiveness of SPs.

In the future, we plan to investigate detailed trends instead of summarizing trends on the whole. In addition, we plan to incorporate other sources as the targets because there are many sources for SP research in addition to the ACM digital library and IEEE explore (e.g., papers from conferences and other search engines).

Acknowledgement

Our thanks go to our shepherd Prof. Rosana Teresinha Vaccare Braga for her valuable comments and guides to improve our paper significantly.

References

[1] E. Fernandez, H. Washizaki, N. Yoshioka, A. Kubo, Y. Fukuzawa, Classifying security patterns, The 10th Asia Pacific Web Conference (APWeb2008), Shenyang, China, April 26-28, 2008.
[2] A. Kubo, H. Washizaki, A. Takasu, Y. Fukazawa, "Analyzing relations among software patterns based on document similarity," International Conference on Information Technology: Coding and Computing (ITCC 2005), 2005.
[3] E. Fernandez, H. Washizaki, N. Yoshioka, "Abstract security patterns," 15th Conference on Pattern Languages of Programs (PLoP 2008), 2008.
[4] A.V. Uzunov, E.B. Fernandez, and K. Falkner, "Engineering Security into Distributed Systems: A Survey of Methodologies," Journal of Universal Computer Science, Vol.18, No.20, pp.2920-3006, 2012.
[5] N. Yoshioka, H. Washizaki, and K. Maruyama, "A survey on security patterns," Progress in Informatics 5(2), 2008.
[6] S. T. Halkidis et al. "Quantitative evaluation of systems with security patterns using a fuzzy approach," OTM 2006.
[7] M. Schumacher et al. "Security Patterns: Integrating security and systems engineering." John Wiley & Sons, 2005.
[8] E. B. Fernandez, "Security Patterns in Practice: Designing Secure Architectures Using Software Patterns", John Wiley & Sons, 2013.
[9] A. Bandara, H. Shinpei, J. Jurjens, H. Kaiya, A. Kubo, R. Laney, H. Mouratidis, A. Nhlabatsi, B. Nuseibeh, Y. Tahara, T. Tun, H. Washizaki, N. Yoshioka, Y. Yu, "Security patterns: Comparing modeling approaches," in "Software Engineering for Secure Systems", pp.75-111, IGI Global, 2010.
[10] D. G. Rosado et al. "Comparison of security patterns," IJCSNS 6(2), 2006.
[11] K. Petersen et al. "Systematic mapping studies in software engineering," EASE 2008.
[12] ACM Digital Library, http://dl.acm.org/
[13] IEEE Xplorer, http://ieeexplore.ieee.org/Xplore/home.js
[14] H. Washizaki et al. "Network Analysis for Software Patterns including Organizational Patterns in Portland Pattern Repository." Agile 2014.
[15] P. T. Devanbu et al. "Software engineering for security: A roadmap," ICSE 2000.
[16] E. B. Fernandez et al. "On building secure SCADA systems using security patterns," CSIIRW 2009.
[17] P. H. Nguyen et al. "Model-driven security with a system of aspect-oriented security design patterns," VAO 2014
[18] V. Horvath et al. "From security patterns to implementation using petri nets," SESS 2008.
[19] M. Hafiz, "Security patterns and evolution of MTA architecture," OOPSLA 2005.
[20] M. Bunke et al. "An architecture-centric approach to detecting security patterns in software," ESSoS 2011.
[21] K. Yskout et al. "Does organizing security patterns focus architectural choices?" ICSE 2012.
[22] P. Rimba, "Building high assurance secure applications using security patterns for capability-based platforms," ICSE 2013.
[23] N. Delessy et al. "A pattern-driven security process for SOA applications," ARES 2008.
[24] E. Fernandez et al. "An analysis of modeling flaws in HL7 and JAHIS," SAC 2005.
[25] Y. Shiroma, H. Washizaki, Y. Fukazawa, A. Kubo, and N. Yoshioka, "Model-driven security patterns application based on dependences among patterns," ARES 2010.
[26] T. Kobashi, N. Yoshioka, T. Okubo, H. Kaiya, H. Washizaki, and Y. Fukazawa, "Validating security design patterns application using model testing," ARES 2013.
[27] A. Cuevas et al. "A security pattern for untraceable secret handshakes," SECURWARE 2009.
[28] R. Bouaziz et al. "An engineering process for security patterns application in component based models," WETICE 2013.
[29] M. Solinas et al. "Embedding security patterns into a domain model," DEXA 2009.
[30] R. Bouaziz et al. "Applying security patterns for component based applications using UML profile," CSE 2012.
[31] J. Dong et al. "Model checking security pattern compositions." QSIC'07.
[32] Busnel, Pierre, et al. "Achieving socio-technical confidentiality using security pattern in smart homes." FGCN'08
[33] M. Weiss et al. "Selecting security patterns that fulfill security requirements," RE 2008.
[34] T. Okubo, H. Kaiya, and N. Yoshioka, "Effective security impact analysis with patterns for software enhancement" ARES 2011.
[35] A. Cuevas et al. "Security patterns for capturing encryption-based access control to sensor data," SECURWARE 2008.
[36] S. Hasheminejad et al. "Selecting proper security patterns using text classification," CiSE 2009.
[37] P. Ruamjinda et al. "Framwork for information security standards storage and retrieval using security patterns," ICSESS 2013.
[38] M. Ratchakom et al. "A process model design and tool support for information assets access control using security patterns," JCSSE 2011.
[39] R. Bouaziz et al. "Secure component based applications through security patterns." GreenCom 2012.
[40] E. B. Fernandez et al. "Building secure systems: From threats to security patterns," SCCC 2010.
[41] B. Smith et al. "On the effective use of security test patterns," SERE 2012.
[42] J. Ruiz et al. "A security engineering process for systems of systems using security patterns," SysCon 2014.
[43] M. Menzel et al. "A pattern-driven generation of security policies for service-oriented architectures," ICWS 2010.
[44] A. Mourad et al. "A novel approach for the development and deployment of security patterns," SocialCom 2010.
[45] T. Heyman et al. "Using security patterns to combine security metrics," ARES 2008.
[46] D. Berardi, "Using DLs to reason on UML class diagrams," Workshop on Applications of Description Logic, 2002.
[47] T. Heyman et al. "An Analysis of the Security Patterns Landscape" IEEE Computer Society, 2007.
[48] Alvi, Aleem Khalid et al. "A natural classification scheme for software security patterns." DASC, 2011
[49] Alvi, Aleem Khalid et al. "A Comparative Study of Software Security Pattern Classifications." ARES, 2012.