Proceedings of the Federated Conference on Computer Science and Information Systems pp. 875–878
DOI: 10.15439/2019F213
ISSN 2300-5963 ACSIS, Vol. 18

Security-oriented agile approach with AgileSafe and OWASP ASVS

Katarzyna Łukasiewicz
Gdańsk University of Technology
ul. Narutowicza 11/12, 80-233
Gdańsk, Poland
Email: katlukas@pg.edu.pl

Sara Cygańska
IHS Markit
ul. Marynarki Polskiej 163, 80-868 Gdańsk, Poland
Email: sara.cyganska@ihsmarkit.com

***Abstract*—In this paper we demonstrate a security enhancing approach based on a method called AgileSafe that can be adapted to support the introduction of OWASP ASVS compliant practices focused on improving security level to the agile software development process. We also present results of the survey evaluating selected agile inspired security practices that can be incorporated into an agile process. Based on the survey's results, these practices were used as an input to AgileSafe method as well as to demonstrate their potential to comply with OWASP ASVS requirements.**

I. Introduction

THE concern for providing secure systems has become increasingly important throughout the years. With the rapid progress in the IT domain, expansion of the internet solutions and the level of general computer science knowledge, the problem with security affects multiple domains. At the same time, the changing markets and need for flexibility encourages many companies to adopt agile approach [1].

The goal of the research described in this paper was to identify security-focused agile practices, evaluate their usability and impact so that the positively assessed practices could be incorporated into an OWASP ASVS [2] compliant process, as a part of AgileSafe method [3].

II. Background

A. Agile methods

Ever since the announcement of the Agile Manifesto [4], the agile methods such as Scrum [5], eXtreme Programming [6] or Kanban [7] have been growing increasingly in popularity. The reports of the benefits experienced by numerous companies [8][9] encouraged the trend to shift from traditional, plan-driven methods to the agile ones. What is important is that this shift has not only concerned small and evolving companies which are considered a target of the agile approach. Bigger organizations with larger teams or corporate structures have also sought ways to incorporate agile approach, which resulted in methods such as SAFe [10] or DevOps [11].

B. OWASP ASVS

The name of the OWASP Application Security Verification Standard (OWASP ASVS) comes from the organization with same name, which created it - The Open Web Application Security Project [12]. Its two main goals are to help creating and maintaining secure software and help in defining requirements between service providers and their clients.

OWASP ASVS has been chosen for this research due to its versatility, open access and popularity among practitioners [13]. The domain of web applications is at the forefront of security issues, with frequent news about major security breaches [14]. For this reason, catering a solution that would allow combining agile security practices with OWASP ASVS requirements could be of interest to many organizations.

III. AgileSafe

In the safety context, quite similarly to the security one, norms and standards are vital to ensure the level of trust and quality of high-integrity systems. In order to enable safety-critical software companies to adopt hybrid agile approach while satisfying the regulatory requirements of applicable standards, AgileSafe [15] method has been proposed. It presents a framework for collecting and suggesting the most suitable agile practices for a given project, as well as the means for managing and monitoring conformance with the applicable regulatory requirements.

A. Overview

As an input to AgileSafe takes the characteristics of a project in which the new approach will be implemented (Project Characteristics) as well as a list of regulations (Regulatory Requirements), which the project needs to comply with.

Based on this information, the user is guided through the process of practices suggestion as well as the process of preparing a set of assurance arguments [16] that will help the user to maintain conformance with given norms and standards. As a result, the user obtains a tailored Project Practices Set, which would best suit a project with given characteristics and regulation restrictions as well as a set of

assurance arguments to monitor compliance with the chosen regulations.

B. Practices Knowledge Base

The information about practices available in AgileSafe, their capability to answer given Project Characteristics and Regulatory Requirements, is kept in the Practices Knowledge Base. Each practice is described using the same template that is then translated into OWL and managed using Protégé [17].

A. Assurance arguments

In order to ensure that the Regulatory Requirements will be met when applying the new agile approach, AgileSafe uses a set of assurance arguments. The highest level of abstraction is represented by Practices Compliance Argument. It is created separately for each standard added to the method and collects all of the practices from Practices Knowledge Base that have a potential to answer the standard's requirements. Such practices are arranged accordingly in the argument structure for a given standard requirements.

In this particular research, we focused on the most general Practices Compliance Argument for OWASP ASVS and the security-oriented practices identified in the course of this research, to keep it independent from any particular software project.

II. Security-oriented agile practices

In order to propose agile security practices that could extend the Practices Knowledge Base of the AgileSafe method, a review of the scientific literature and articles on blogs and industry portals was carried out.

A. Identification of security-oriented agile practices

While there are many well-known security-oriented practices such as threat modelling or attack trees, in this research we wanted to expand this list and focus on less obvious, agile inspired practices, to enrich the Practice Knowledge Base of AgileSafe method.

A literature review has been performed and as a result 10 articles were selected to be used in further work [18][19][20][21][22][23][24][25][26][27].

B. Selected practices description

Based on the articles identified in the research, 10 hybrid agile security-oriented practices were identified:

Abuser Stories. They describe, using a form similar to regular User Stories, how the system might be attacked and how assets might be put in risk. They should be estimated in accordance to how much damage they may potentially cause and probability of a successful attack. [19]

Evil user stories. This practice describes actions of malicious user (e.g. "As a hacker I want to steal payment information of other clients, so I can sell it."). They may be used as a starting point for threat modelling. [20]

Misuse cases. They are negative use cases. They illustrate behavior not wanted in the system, that can cause a security breach and can be described using UML diagrams. [21]

Protection poker. This is a software security game intended to create a list of each requirement relative security risk. It derives form Planning Poker technique of estimation. [22]

Second delivery. This is a process, that aims to integrate security related solutions to the project that already satisfies functional requirements. It is based on XP methodology. [23]

Security engineer. It calls for adding an expert role, that brings up-to-date security knowledge to developers' team. His insight is useful during multiple phases and actions in project.

Security Sprint. This is a practice inspired by Scrum. It's similar to regular Sprint except that it focuses on security issues. [24]

Security-focused code reviews. Such reviews should be performed for every story separately – no story can be completed without security review, fixing findings from review and then passing re-review. [25]

S-Mark and S-Tag. Originating from Secure Scrum, they are a way to document identified security issues in Scrum Backlog by creating system of tags (security issues) and markings for stories related to respective tags. [18][27]

Spikes. They are a way to include security analysis and design within Scrum. They accommodate activities that don't produce customer-valued product, like security analysis or system designing. [26]

III. survey

In order to evaluate the usability and accessibility of the selected security-oriented agile practices in projects with high security requirements a survey was conducted. It tackled 10 specific agile security-oriented practices, asking the respondents to rate their respective ease of use and security enhancement potential.

Subjects chosen to participate in the survey were 24 IT practitioners (both development and operations) from 7 different software companies, ranging from small to corporate ones, from Poland and UK. The questionnaire was distributed mostly by email and direct messages in social networks, eliminating probability of acquiring responses from random, unrelated to the field respondents. The respondents were also provided with the practices detailed descriptions.

A. Results

For each practice two closed questions were asked about its ease of use and if it's improving security in the project. In total, 15 of all the participants made their choices in those questions. Also, each practice was open to comments from the respondents. The results are presented in the Fig. 3 and Fig. 4.

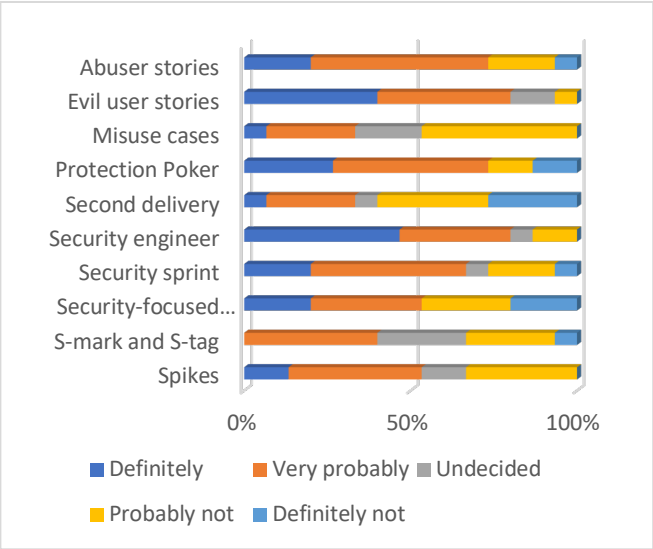

Fig. 3 Is this practice ease to use?

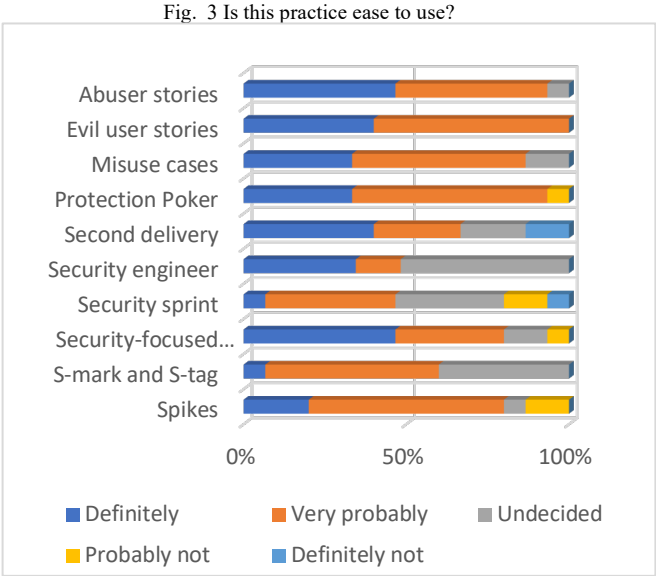

Fig. 1 Does this practice improve security in the project?

Abuser Stories. None of respondents chose negative answer for this practice security improvement potential and not many had doubts about its positive influence. But 26,67% believed it would not be easy to use - as the reason they mostly described difficulty in estimating attack probability. Despite this fact, this practice has potential benefits in the projects wanting to comply with OWASP.

Evil user stories. This practice was also positively rated in terms of security improvement. What's more, only 20% expressed doubts or were undecided about its ease of use. Those results categorize it as both efficient and easy to get started with. Respondent commented on possible threat to project agility in case of creating a large number of evil user stories.

Protection Poker. Majority of respondents found this practice easy to use – among the benefits they listed possible automation of prioritization. The doubts were similar to those for Abuser Stories practice – difficulty in estimation of attack ease and probability. Another noticed difficulty is the necessity for security experts to participate in the process. Despite that problems only 7% didn't rate the practice positively in terms of security.

Second delivery. This practice didn't occur as easy to use to most respondents. A lot of them were concerned about the need to re-implement huge parts of system in order to satisfy security requirements. 67% of answers in question about security were positive, but considering its difficulty, this practice might not cause some problems in actual development process. Also, a significant problem with security was noticed, that during the first development unexpected security flaws might be introduced to the system that are not addressed in the second delivery.

Security engineer. Most of respondents rated this practice positively in terms of ease of use, as it wouldn't require additional amount of work from the team and it would be beneficent to have an expert that is not writing the code himself. Among listed problems were difficulty in finding the suitable person for this role and risk of putting all of responsibility for security on one person. Despite those issues, rating in security improvement area was positive, with only 7% of participant undecided and none rating it negatively.

Security Sprint. The majority of respondents rated this practice as easy to use, but doubts were expressed that it could lead to development work duplications. Also, the question was asked about the case in which not enough security tasks are defined to fill the whole sprint. 47% of answers were positive in terms of security improvements, but as much as 33% of participants were undecided. This can indicate that practice description should be clarified when added to the AgileSafe Knowledge Base.

Security-focused code reviews. Opinions on this practice's ease of use are divided – the results for "Definitely" and "Definitely not" are equal (20%). Among mentioned problems were difficulty with finding a suitable expert and a lot of additional effort required for conducting such reviews. Despite that, most of respondents decided that this practice improves security in the project (80%). But the expected improvement seems not to be worth the effort required.

S-Marks and S-Tags. None of the respondents found this practice definitely easy to use, and 40% decided it's probably easy to use. Considering amount of answers "Undecided" in both questions, this practice might be too complicated to take up without previous training. Practice gained no negative rating in terms of security, but concerns were raised that it might be possible to lose track of some tags and marks and therefore omit some security issues in development. Also, the question was asked about support in existing project management tools, which could solve tracking problem.

Spikes. Although the majority of respondents (53%) rated this practice as easy to use, 33% doubted it – some commented that it's difficult to understand. However, in terms of security, most of participants expressed no concern

about its influence on project security. A question was also asked about other practices that can be used in security projects development. Only two answers were provided – bug bounty and security hackathon. This shows that it's not a common knowledge among developers.

The results show that, although not all practices are easy to use, most of them serve their purpose well by explicitly requiring some security assurance activities. Some of those that scored lowest in terms of easiness might be improved by description clarification, training or providing supporting tools.

IV. OWASP Assurance Argument

Because of the positive results of practices security assurance evaluation, the next step was to add them to the Practices Knowledge Base. The selected practices were analyzed according to the AgileSafe practice description template and incorporated into the knowledge base. Newly added security practices were assessed with respect to their OWASP conformance potential.

OWASP ASVS requirements has been added to the method and based on the Practices Compliance Assurance Argument Pattern, were mapped to the Practices Compliance Assurance Argument using NOR-STA tool [28].

All of the OWASP ASVS requirements were successfully mapped into the structure. The practices that were able to answer specific requirements were attached with a relevant rationale in the NOR-STA tool. None of the requirements were left without a practice that might be able to provide conformance.

It is worth noting that there was not one practice that would sufficiently address all of the OWASP ASVS requirements, which means that in a project wishing to comply with the standard, implementing a combination of the analyzed practices would be needed.

The prepared Practices Compliance Argument has been accepted as a part of the AgileSafe potential extension for security assurance domain. Based on this argument, depending on a given project's Project Characteristics, a new hybrid approach with OWASP ASVS compliance potential could be suggested.

V. Conclusions

During the literature review, 10 security-oriented agile practices were identified. The practices were positively assessed in the conducted surveys and successfully enriched the Agile Practices Knowledge Base. The OWASP ASVS was mapped into the method and formed, along with the identified practices, the Practices Compliance Argument, which after updating it with all of the other applicable practices available in AgileSafe, might be further used to support practices selection in specific projects. A case study carried out with such projects, going through the whole practices selection process of AgileSafe might be performed as next step of the research.

References

[1] "VersionOne® Releases 11th Annual State of Agile Report", VersionOne, 2017. [Online]. Available: https://www.versionone.com/about/press-releases/versionone-releases-11th-annual-state-of-agile-report/

[2] J. Manico, "OWASP Application Security Verification Standard," 2015.

[3] K. Łukasiewicz, J. Górski, "AgileSafe – a method of introducing agile practices into safety-critical software development processes," *Proceedings of the Federated Conference on Computer Science,* vol. Vol. 8, pp. 1549-1552, 2016.

[4] Agile Manifesto,. *Manifesto for Agile Software Development*. 2001 [online] Available at: http://agilemanifesto.org.

[5] K. Schwaber and M. Beedle, *Agile software development with scrum*. Upper Saddle River, N.J: Prentice Hall, 2002

[6] K. Beck and C. Andres, *Extreme programming explained*. Addison-Wesley Professional, 2004.

[7] D. Anderson, *Kanban.* Sequim: Blue Hole Press, 2010.

[8] J. Drobka, D. Noftz and R. Raghu, "Piloting XP on four mission-critical projects". *IEEE Softw.*, 21(6), pp.70-75, 2004

[9] M. Lindvall., D. Muthig, A/ Dagnino, C. Wallin, M. Stupperich, D. Kiefer, J. May & T. Kähkönen. "Agile Software Development in Large Organizations*"* in Computer, 37(12), pp. 26-34, 2004.

[10] R. Knaster, D. Leffingwell, *SAFe Distilled: Applying the Scaled Agile Framework for Lean Software and Systems Engineering*. Addison-Wesley Professional, 2017.

[11] J Kim, G., Willis, J., Debois, P., Humble, J., Allspaw, J. *The DevOps Handbook*. Trade Select, 2016.

[12] OWASP, "OWASP,", [Online]. Available https://www.owasp.org/index.php/Main_Page.

[13] OWASP users [Online] Available: https://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project#tab=ASVS_Users

[14] World's Biggest Data Breaches & Hacks, 2019, [Online] Available: https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/.

[15] K. Łukasiewicz "Method of selecting programming practices for the safety-critical software development projects," Ph.D. dissertation, Dept. Soft. Eng., Gdańsk Univ. of Technology, Gdańsk, Poland, 2019.

[16] J Górski, J., Jarzębowicz, A., Leszczyna, R., Miler, J. and Olszewski, M. "Trust case: justifying trust in an IT solution". *Reliability Engineering & System Safety*, 89(1), pp.33-47. 2005

[17] Musen, M.A. "The Protégé project: A look back and a look forward". AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4), June 2015.

[18] J D. Mougouei, N. Fazlida, M. Sani, M. M. Almasi, "S-Scrum: a Secure Methodology for Agile Development of Web Services," *World of Computer Science and Information Technology Journal (WCSIT),* vol. 3, no. 1, pp. 15-19, 2013.

[19] J. Peeters, "Agile security requirements engineering." Symposium on Requirements Engineering for Information Security, 2005

[20] E. A. Fischer, "Federal Laws Relating to Cybersecurity: Overview of Major Issues, Current Laws, and Proposed Legislation," 2014

[21] G. Sindre, A. L. Opdahl, "Eliciting security requirements with misuse cases".

[22] L. Williams, A. Meneely, G. Shipley, "Protection Poker: The New Software Security "Game"".

[23] E. G. Aydal, R. F. Paige, H. Chivers, P. J. Brooke, "Security Planning and Refactoring in Extreme Programming"

[24] G. Boström, J. Wäyrynen, M. Bodén, K. Beznosov, P. Kruchten, "Extending XP Practices to Support Security Requirements Engineering"

[25] T. Nguyen, "Integrating Security into Agile Methodologies,", http://www.umsl.edu/~sauterv/analysis/F2015/Integrating%20Security%20into%20Agile%20methodologies.html.htm

[26] OWASP, "Agile Software Development: Don't Forget EVIL User Stories,", https://www.owasp.org/index.php/Agile_Software_Development:_Don%27t_Forget_EVIL_User_Stories.

[27] C. Pohl, H.-J. Hof, "Secure Scrum: Development of Secure Software with Scrum," in *SECURWARE 2015 : The Ninth International Conference on Emerging Security Information, Systems and Technologies*, 2015

[28] NOR-STA project Portal . 2017. [online] Available at: www.nor-sta.eu