Introducing a Novel Security-Enhanced Agile Software Development Process

Martin Boldt, Blekinge Institute of Technology, Karlskrona, Sweden
Andreas Jacobsson, Malmö University, Malmö, Sweden
Dejan Baca, Ericsson AB, Karlskrona, Sweden
Bengt Carlsson, Institute of Technology, Karlskrona, Sweden

ABSTRACT

In this paper, a novel security-enhanced agile software development process, SEAP, is introduced. It has been designed, tested, and implemented at Ericsson AB, specifically in the development of a mobile money transfer system. Two important features of SEAP are 1) that it includes additional security competences, and 2) that it includes the continuous conduction of an integrated risk analysis for identifying potential threats. As a general finding of implementing SEAP in software development, the developers solve a large proportion of the risks in a timely, yet cost-efficient manner. The default agile software development process at Ericsson AB, i.e. where SEAP was not included, required significantly more employee hours spent for every risk identified compared to when integrating SEAP. The default development process left 50.0% of the risks unattended in the software version that was released, while the application of SEAP reduced that figure to 22.5%. Furthermore, SEAP increased the proportion of risks that were corrected from 12.5% to 67.9%, a more than five times increment.

KEYWORDS


INTRODUCTION

Agile methods were initially seen as ideal for non-critical product developments. However, during recent years there have been several initiatives for the adaptation of agile methods for domains also in critical areas, such as, medical devices (Fitzgerald, Stol, O’Sullivan, & O’Brien, 2013) (McHugh, McCaffrey, & Coady, 2014) and money transfer systems (Baca, Boldt, Carlsson, & Jacobsson, 2015). That is, in situations where security risks are prominent characteristics. Despite these risks, most existing agile development methods hold an improvement-potential with regards to security aspects, e.g. the work by Cruzes et al. (Cruzes, Felderer, Oyetoyan, Gander, & Pekaric, 2017). As a result, security is often added afterwards or included in the process by way of external resources. While there exist alternatives, such as, discrete security methods (such as checklists, management standards, etc.) that can supplement agile methods, few of these integrate into agile methods in a seamless, quality-enhanced yet cost-efficient manner.

In traditional software development processes, a security officer (also known as the product owner of security) performs audits or acts as a security advisor to the developers usually handles all matters related to security. However, in an agile development process, where iterations are short and
changes made by the minute, it is not always possible to include a security officer from the outside, neither to integrate complementary discrete security methods in the process.

In terms of security, a general drawback with agile methods is the lack of a complete and updated picture of how all of the software requirements are implemented, which often results in difficulties to grasp let alone analyze the risks. Moreover, security flaws are often either overlooked or handled in an insufficient way. Due to the general work method in agile development, it is typically not possible to integrate systematic risk analysis in such a way that usable yet probable results are yielded. It is therefore important to predict how different ways of adding security aspects to agile processes will improve the security of the final product, in order to make sure that any additional costs due to the addition of security aspects are motivated by increased product security.

The work in this paper uses quantitative measures from risk analyses on a software product developed using an agile software development process. Thus, the risks identified in the risk analyses are used as indicators of the software product’s security level, i.e., risk data was used as a way to estimate the end-product’s level of security. In addition, the risk analysis data also reflects the efficiency of the agile team members in identifying risks in the software, as well as addressing them at an early stage. Although there exist other indicators of the level of security in software being developed (e.g., measures from security test cases or code reviews) that also could be used, those are considered out of scope for the focus of this study. Further, there is also a correlation between software security and software quality that must be taken into account. More precisely, there is a need for cost-efficient processes, tools, and guidelines for developing security-critical software in an agile way. In this work, our first task and contribution is to find a way to systematically integrate security management (including risk analysis) in agile software development project while at the same time meeting all the other software requirements with both quality and cost-efficiency. While our second task is to evaluate the security effects of the enhanced agile process in the development of a security-critical software product in an industrial setting.

Computer systems, network applications, and web services are vulnerable to attacks and need both protection of the involved assets and a way to deal with the consequences of bad, hazardous, or malicious practices. The work documented in this paper demonstrates how security features can be integrated into an agile software development method. The method has been tested at Ericsson AB, and in this setting and in accordance with the idea behind the method, a mobile-money transfer system that handles large amounts of monetary transactions has been developed. The mobile-money transfer system, which is a web-based service, intended to run as a stand-alone application on mobile units, is similar to online banking services in that it must be considered a highly attractive target for an abundance of malicious acts, such as, money laundering schemes, embezzlement attempts, and other criminal activities. In other words, security of the product is a highly prioritized concern.

This paper introduces a new security-enhanced agile software development process, hereinafter referred to as SEAP. A shorter version of this investigation along with a brief description of SEAP has previously been outlined by the authors in (Baca, Boldt, Carlsson, & Jacobsson, 2015). Matters related to security are, in the case of SEAP, handled both as a process of integrating security in the agile development process and as a way of introducing specific security tools usable for the development teams. The introduction of integrated security tools facilitates a faster security analysis, a more detailed examination, and increased integration between risk analysis and security-enhancing measures, as well as, the overall software development. A main contribution in this work demonstrates how security can be integrated into an agile method in a real industrial setting, together with recommendations on how SEAP can be implemented in the development processes.