Chapter 58

Measuring Security: A Step Towards Enhancing Security of System

Shruti Jaiswal
Department of Computer Science and Engineering, Delhi Technological University, Delhi, India

Daya Gupta
Department of Computer Science and Engineering, Delhi Technological University, Delhi, India

ABSTRACT

The researchers have been focusing on embedding security from the early phases of software development lifecycle. They have researched and innovated a field of Security Engineering where security concerns are embedded during requirement, design, and testing phases of software development. Efforts were made in developing methods, methodologies, and tools to handle security issues. Various methods are present in the literature for eliciting, analyzing and prioritizing the security requirements. During the design phase based on prioritized requirements, environment parameters and attribute a suitable security algorithm mainly cryptography algorithms are identified. Then a question arises how to test the effectiveness of chosen algorithm? Therefore, as an answer to the issue in this paper, a process for Security Testing is presented that evaluates the selected security algorithms. Evaluation is done by generating the test scenarios for functionalities using sequence diagram representing the threats at vulnerable points. Then, checking the mitigation of potential threats at identified vulnerable points. A security index is generated which shows the effectiveness of deployed/ chosen security algorithm. The process ends with the generation of a test report depicting the testing summary. For a clear understanding of the process, the proposal is illustrated with a case study of the cloud storage as a service model.

DOI: 10.4018/978-1-7998-7705-9.ch058

INTRODUCTION

Field of Security Engineering has emerged focusing on embedding security in the software using methods, processes, and tools. The term coined by Gupta and Prakash (2001) deals with a systematic approach to develop the secure software system. Firstly during requirements engineering phase, security requirements which emerge from potential threats are elicited, analyzed and prioritized. During the design phase, suitable cryptography algorithm that mitigates threats is selected and deployed during implementation. Finally, testing is done to validate that system is secure from potential threats. Researcher Firesmith (2003) defines Security Requirements as the high-level requirement that gives a detailed specification of system behavior which is not acceptable. In the literature, various proposals that address security requirements identification and analysis are found such as secure tropos extension of the Tropos methodology (Mouratidis, Giorgini, Manson, & Philp, 2002), an intentional anti-model extension of the KAOS methodology (Lamsweerde, 2004). The proposal by researcher N. Mayer, Heymans, and Matulevicius (2007), SQUARE methodology (Mead & Stehney, 2005) prescribes different phases to develop the secure system such as Elicitation, Categorization, and Prioritization of Security Requirement. These proposals mainly focus on eliciting the security requirements and assume that a suitable security mechanism can be deployed to mitigate the identified threats. However, they do not validate the deployed algorithm to check whether the identified threats are mitigated. It means besides conventional testing for functionality and quality factors, testing for potential attacks is also necessary.

Security testing is the process of ensuring/ analyzing whether the selected security algorithms are mitigating all possible threats to the system. Our research shows that Security Testing is much more different from traditional functional testing. Because security testing requires a tester having detailed knowledge of cryptology, to ensure that developed system can be protected from potential attacks. One of the proposals of security testing described in Arkin, Stender, and McGraw (2005) is known as penetration testing, in this tester needs to think like an attacker/ intruder and performs various attacks to identify the existing threats. Recently testing technique showing remarkable results is a model- based security testing (MBST) (Felderer, Zech, Breu, Bchler, & Pretschner, 2016), (Schieferdecker, Grossmann, & Schneider, 2012). In this method, test cases are generated from a set of models (architectural, functional, risk) depicting the behavior expected from the system and its environment. Test cases are generated with the intent of identification of potential vulnerabilities by checking the deviation from expected system behavior. Security testing as presented in Mouratidis and Giorgini (2007) is a novel scenario-based method that develops a scenario for testing potential attacks by identified malicious actors. They are constructing a security attack scenario (SAS) template which describes the sequence of possible attacks on resources by malicious actors. These scenarios are then used to verify whether deployed security mechanism mitigates the attacks.

It is impossible for a cryptography algorithm to mitigate all possible threats. So the cryptography algorithm is evaluated by generating a metric that estimate the risk of non-mitigated threats by deployed security algorithm. The metric value is then compared to the predefined epsilon; comparison result would act as a guide for a software developer to enhance/ revise the existing cryptographic algorithm. Epsilon value shows the tolerable value of risk in the system. Therefore, in this paper, a proposal for testing security of the system is presented that will end up specifying the security index showing gap in the security of deployed security algorithms.