Enhancing Proactive Control Mobile and Web Software Security Education with Hands-on Labware

Hossain Shahriar, Kai Qian
Kennesaw State University
Marietta, GA, USA
{hshahria, kqian}@kennesaw.edu

Atef Shalan
Georgia Southern University
Statesboro, GA, USA
amohamed@georgiasouthern.edu

Fan Wu
Tuskegee University
Tuskegee, AL USA
fwu@tuskegee.edu

Abstract - **While the number of mobile and web applications is growing exponentially, the mobile and web security threat landscape is growing explosively. Malicious malware may attack vulnerable applications and obtain personal or enterprise confidential data anywhere and anytime. Most vulnerabilities should be addressed and fixed during the early stages of software development. However, many software development professionals lack the awareness of the importance of security vulnerabilities and the necessary knowledge and skills at the software development stage. This paper addresses the needs and challenges of the lack of pedagogical materials and real-world learning environment in ProActive Control for Software Security (PASS) through effective, engaging, and investigative authentic learning approaches.**

Keywords: Proactive control, Software security, OWASP, Output encoding, Vulnerability.

1. INTRODUCTION

The security threats to mobile and web applications are growing explosively. Software security has now become a wider security concept. Developing secure software is essential and crucial for Confidentiality, Integrity, and Availability of all software applications, such as mobile and web applications. Most malicious attacks take advantage of vulnerabilities in applications, such as sensitive data leakage via inadvertent or side channel, unsecured sensitive data storage, and data transmission. Most vulnerabilities should be addressed in the mobile software development phase. However, most development teams often have little to no time for security remediation, as they are usually tasked for the project deadlines. Even worse, many development professionals lack awareness of the importance of security vulnerability and the necessary secure knowledge and skills at the development stage.

Most developers did not learn about secure coding or cryptography in school. They lack critical core controls and implement code in an insecure way. Security vulnerabilities open the doors to security threats and attacks, which may be prevented at an early stage. The rapid security threat growth has resulted in a shortage of security personnel. Education for secure mobile and web application development is in big demand in IT fields. With more schools developing teaching materials on mobile application development, more educational activities are needed to promote security education especially for proactive secure software development. However, secure software development is not well represented in most schools' computing curriculum. It is necessary to build capacity for computing faculty in secure software development as an important and integral part of security education. The proactive controls are intended to provide software developers with initial awareness for building secure software. These controls can help software developers secure application coding consistently and throughout application development proactively rather than the patch-and-fix passive traditional security management approach.

We propose to build the capacity on ProActive Control for Software Security (PASS) through three venues: (1) curriculum development and enhancement with a collection of ten transferrable learning modules with companion hands-on labs on mobile and web software development (security requirement specification control, data store and database security control, data communication control, input validation control, output decoding control, access control, logging monitoring and exception handling control, framework API control, file inclusion control, and session control) which can be integrated into existing undergraduate and graduate computing classes that will be mapped to ISA KAs proposed in CSEC 2017 [3] and CC 2020 [4] curricula to enhance the student's secure mobile software development ability; (2) faculty expertise and partnership development with annual faculty summer workshops and webinar each semester, where the selected teaching materials will be demonstrated and hands-on exercises will be practiced during the workshops; and (3) The mobile and web hands-on learning modules and labs are designed based on the OWASP 2018 Top Ten Proactive Controls open source project with 10 most important security techniques that should be applied proactively at the early stages of software development to ensure maximum effectiveness. The overall goal of this PASS project is to address the needs and challenges of building capacity with proactive controls for software security development and the lack of pedagogical materials and real world learning environment in secure software development through effective, engaging, and investigative approaches. The PASS project will help students and faculty to know what should be considered or best practices during mobile and web software development and raise their overall security level for software development. Students can learn from the misuse vulnerability cases and insecure mistakes of other organizations. Simultaneously, such cases should be prevented by mitigation actions described in protection use cases for building secure software.

2. LABWARE DESIGN

Each module in the mobile and web PASS labware is designed based on a real world cybersecurity vulnerability and consists of three components: pre-lab, hands-on lab activity, and post-lab for student to add on new attacks and solutions. All proactive control learning modules are designed based on OWASP

recommended and real world cases. The project will provide ten transferable learning modules including Security Requirement Specification Control (Mobile, Web), Data Store and Database Security Control (Mobile, Web), Data Communication Control (Mobile, Web), Input Validation Control (Web), Output Encoding Control (Web), Access Control (Mobile, Web), Logging Monitoring and Exception Handling Control (Mobile), Framework API Control (Mobile), File Inclusion Control (Web), and Session Control (Web).

3. SAMPLE MODULE

Each learning module consists of the pre-lab, hands-on lab activity, and post-lab. Here is the module for output encoding. The learning objectives include understanding (i) the dangers of unsanitized output handling, (ii) the fundamental concepts of output encoding, and (iii) the basic defensive practice skills against malicious injection attacks in mobile software development.

The pre-lab in this module provides a overview of proactive control for a specfic vulnerability. It introduces how such vulnerability takes place, the consequence of such attack, and the proactive control strategy for such vulnerability in design and development phase.

The hands-on activity lab consists of insecure misuse case and secure use case labs. Students can follow the lab tutorial to understand the consequences of an attack in the misuse case lab and learn by self-doing and practice the secure case design to prevent such attacks. It will greatly improve the student learning effectiveness of concept learning and problem solving.

The screenshot of Figure 1 shows the consequence of not having output encoding implemented in a simple program that asks a user to provide an input so that the program can greet the user with a hello message in HTML in the insecure misuse case of the module. When a user enters a malicious script into the text field and presses “Insecure”, the script will be executed, resulting in the alert shown below (Figure 1).

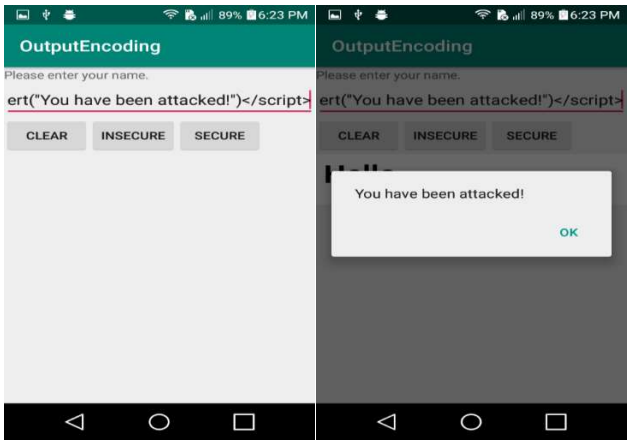

Figure 1: Insecure application

In the secure use case of the module, the vulnerability is located and then vulernability prevention and countermeasures are introduced. In the output encoding hands-on lab activity secure use case, the user is introduced to HTML encoding, and the vulnerability that was previously exposed is prevented. When the same malicious script is entered into the text field and “Secure” is pressed, the script will be HTML encoded and shown in the hello message, as shown in Figure 2.

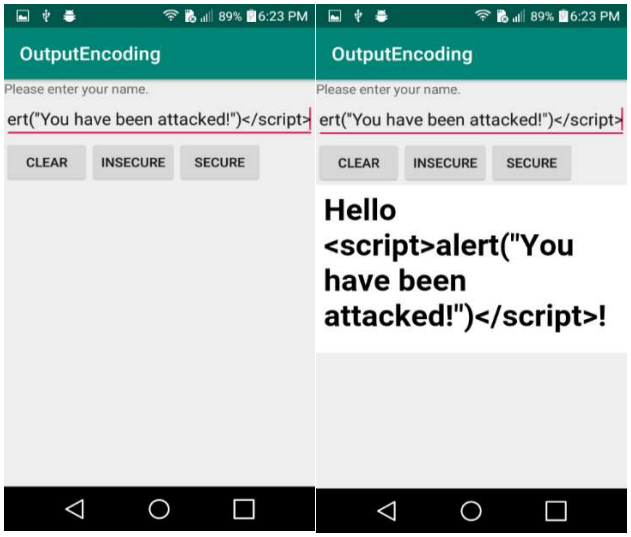

Figure 2: Secure application

In order to promote student active and creative learning, we provide a post-lab for student to explore variable attacks in this category and develop proactive control.solution guidelines themselves. The new attack formats and techniques may be changed and new guideline may need updating as well which will also promote student lifetime learning.

4. CONCLUSION

The presented labware engages students in active learning so that they are learn about security vulnerabilities in both mobile and web applications, such as security requirement specification control, data store, database security control, data communication control, input validation control, output encoding control, access control, logging monitoring, exception handling control, framework API control, file inclusion control, and session control. It will foster students’ innovation ability in the reflection of today’s reality and prepare our undergraduate students for entrance into the industrial workforce.

ACKNOWLEDGMENTS

The work is partially supported by the National Science Foundation under award#1723578. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Anton, K., Manico, J., Bird, J., OWASP Proactive Controls for Developers, 3.0., 2018, Retrieved from https://www.owasp.org/images/b/bc/OWASP_Top_10_Proactive_Controls_V3.pdf

[2] Williams, J., Manico, J., Mattatall, N., Output Encoding Rules Summary. Retrieved from https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet#Output_Encoding_Rules_Summary, 2020.

[3] Cybersecurity Curricula Guidelines, 2017, https://cybered.hosting.acm.org/wp/

[4] Computing Curricula, 2020, https://cc2020.nsparc.msstate.edu/