OWASP WebGoat – Web Application Security Lessons

Professor Yeali S. Sun

Project Description

Web application security is difficult to learn and practice. Not many people have full experience in exploiting web vulnerability. WebGoat is a deliberately insecure web application maintained by OWASP. It is designed to teach web application security lessons. In this term project, you should install WebGoat and practice with WebGoat to learn how popular web attacks actually work. In each WebGoat lesson, you should demonstrate your understanding of a security issue by exploiting a real vulnerability in the WebGoat. It is intended to let you learn where vulnerability may exist and how exploitation occur, and most important of all, to think about how to defense such web attack.

The operation of the HTTP, the web browser and some web attacks will be covered in the lecture, and the first few lessons will be demonstrated in the class as well, including the basic usage of WebGoat and the tools for exploiting and testing. WebGoat v5.4 provides more than 50 web security lessons. Please follow the lesson instructions, hints, and solutions to complete as more lessons as possible on your own. If you are unable to comprehend a lesson, walkthrough videos are available on the Internet.

The final term report should include the 2 sections. In the first section, please detail how to install WebGoat v5.4 and the additional security tools you install. In the second section, please provide the detail solution to 6 designed lessons. A solution should include: 1) each steps to exploit the vulnerability in the lesson, 2) the reason why the exploitation work, and 3) your suggestion to stop such exploitation in detail.

This term project includes a 20-minute demonstration. You should demonstrate the solution to 3 random lessons (see Requirement and Notice below) to TA, and explain why your solution can successfully exploit the web vulnerability. The successfulness of you solution depends on the standard WebGoat scorecard. Please register your demonstration time in prior.

Grading

- Final term paper: 40%
  - Installation and tools: 10%
  - The solution to 6 designed lessons for the term paper: 30%
- Demonstration: 50%
  - The solution to 3 designed lessons for the term paper: 30%
  - Questions 20%

Requirement and Notice

- At most, two students form a team to do this term project.
- You should install stable WebGoat v6, not 7.x
- The 6 designed lessons for the final term paper are:
  - General, HTTP Splitting
  - Access Control Flaws, LAB: Role Based Access Control
  - AJAX Security, LAB: DOM-Based cross-site scripting
  - Cross-Site Scripting (XSS), LAB: Cross Site Scripting
  - Cross-Site Scripting (XSS), Reflected XSS Attacks
  - Injection Flaws, LAB: SQL Injection
- The 3 lessons for demonstration will be randomly picked in the following set and the 6 designed lessons in the final term paper:
  - AJAX Security, JSON Injection
  - Cross-Site Scripting (XSS), Cross Site Request Forgery (CSRF)
  - Session Management Flaws, Hijack a Session
  - Session Management Flaws, Session Fixation

Useful Links

- OWASP WebGoat Project, https://www.owasp.org/index.php/Category:OWASP_WebGoat_Project
- OWASP WebGoat Github, https://github.com/WebGoat
- OWASP WebGoat v5.4 Web Hacking Simulation WalkThrough Series, http://webappsecmovies.sourceforge.net/webgoat/