



Reto Lippuner



Marcel Wirth

# Web Application Security

Diplomanden	Reto Lippuner, Marcel Wirth
Examinator	Prof. Dr. Peter Heinzmann
Experte	Dr. Paul Schöbi, cnlab AG, Rapperswil-Jona SG
Themengebiet	Sicherheit
Projektpartner	cnlab AG, Rapperswil-Jona SG


Das Open Web Application Security Project (OWASP) hat sich zum Ziel gesetzt, solche Fälle zu vermeiden, indem man Applikationsentwicklern typische Verletzlichkeiten von Webapplikationen näherbringt. Zu diesem Zweck wurde die Lernsoftware WebGoat entwickelt. WebGoat simuliert typische Fehler von Webapplikationen. In über 60 Lektionen kann jedermann Angriffe durchführen und analysieren. So macht man sich mit typischen Angriffen vertraut und lernt Verletzlichkeiten vermeiden. Allerdings wurde die Weiterentwicklung von WebGoat stark vernachlässigt. Unvollständiger oder falscher Inhalt so-

wie viele Programmierfehler verunmöglichen ein effizientes Abarbeiten der Lektionen.

Ziel dieser Diplomarbeit war es, WebGoat wesentlich zu verbessern und zu erweitern. Dazu galt es zuerst die aktuell relevanten Schwachstellen von Webapplikationen zu identifizieren und die vorhandenen WebGoat-Lektionen im Detail zu studieren und auf Fehler und Unvollständigkeiten zu überprüfen. Im Rahmen der Diplomarbeit musste mehr als die Hälfte der bestehenden WebGoat-Lektionen wesentlich überarbeitet werden. Etwa 40 wesentliche Unzulänglichkeiten



Auswirkungen von Schwachstellen



[Logout ?](#)

## Session Fixation

Lesson Plan
←
Hints
→
Solution

**OWASP WebGoat V6.2**

- Introduction
- Admin Functions
- General
- Code Quality
- Concurrency
- Unvalidated Parameters
- Access Control Flaws
- Authentication Flaws
- Session Management Flaws
- [Snoop an Authentication Cookie](#)
- [Hijack a Session](#)
- [Session Fixation](#)
- Cross-Site Scripting (XSS)
- Buffer Overflows
- Injection Flaws
- Improper Error Handling
- Insecure Communication
- Insecure Storage
- Denial of Service
- Insecure Configuration
- Web Services
- AJAX Security
- Challenge

[Restart this Lesson](#)

STAGE 1: You are Hacker Joe and you want to steal the session from Jane. That is why you have to send a prepared mail which looks like an official mail from the bank to her. The mail is already prepared. Only thing missing is a Session ID (SID) in the Link. Alter the link to include a SID.

**You are: Hacker Joe**

**Mail To:** jane.plane@owasp.org  
**Mail From:** admin@webgoatfinancial.com  
**Title:**

<b>Dear MS. Plane</b> <br><br>During the last week we had a few problems with our database. A lot of people complained that there account details are wrong. That is why we kindly ask you to use following link to verify your data:<br><br><center><a href=http://localhost/WebGoat/attack?Screen=48&menu=3206SID=attack>Goat Hills Financial</a></center><br><br>We are sorry for the caused inconvenience and thank you for your cooperation.<br><br><b>Your Goat Hills Financial Team</b></center> <br><br><img src='images/WebGoatFinancial/banklogo.jpg'></center>

Created by: Reto Lippuner, Marcel Wirth

OWASP Foundation | Project WebGoat

von WebGoat wurden behoben. Die vorhandenen Lektionen wurden mit zusätzlichen Erklärungen und Musterlösungen ergänzt. Das Userinterface wurde vereinfacht und intuitiver gestaltet. Die neu erstellte «Introduction Sektion» dürfte dem Neuling den Einstieg in WebGoat wesentlich erleichtern. Zu neu aufgetauchten Verletzlichkeiten bzw. Fehlern von WebApplikationen wurden neue Lektionen erstellt. Es sind dies Lektionen zu den Themen Session Fixation, Multi-Level-Login-Fehler, Insecure Communication und Passwortstärken.

Der WebGoat-Projektleiter Bruce Mayhew hat die im Rahmen dieser Diplomarbeit entstandenen Änderungen in den Hauptentwicklungszweig eingespeist. Somit sollten in der nächsten Version von WebGoat die in dieser Diplomarbeit erarbeiteten Änderungen der gesamten «WebGoat-Community» zur Verfügung stehen.