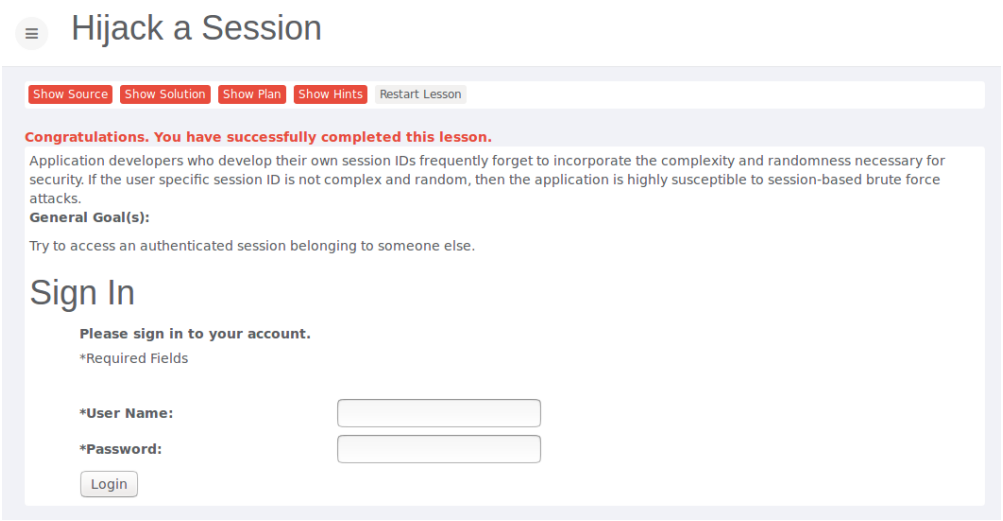
# Web Security
# Assignment 2

Christoph Schmidl
s4226887
Data Science
c.schmidl@student.ru.nl

April 18, 2017

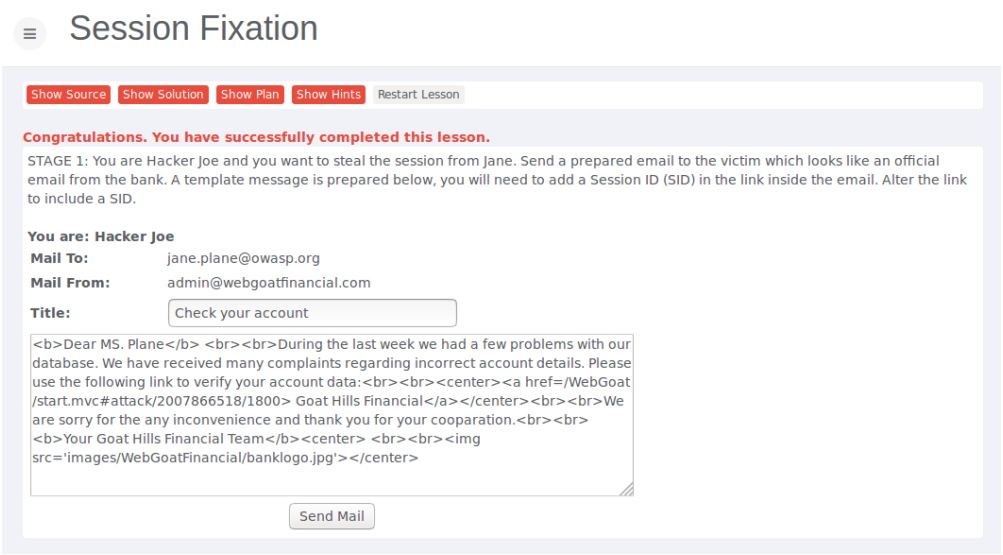
1. Do the following two lessons in WebGoat (you do not have to submit any notes regarding these exercises to blackboard):

   - Session Management Flaws - Hijack a Session




     Session hijacking, sometimes also known as cookie hijacking is the exploitation of a valid computer session - sometimes also called a session key - to gain unauthorized access to information or services in a computer system. In particular, it is used to refer to the theft of a magic cookie used to authenticate a user to a remote server. In this particular lesson the value of the cookie named "WEAKID" can be bruteforced by using a tool like JHijack. To find a value to start with you can use the SessionID Analysis Tab of WebScarab. You do not know the identity of the user's session you are going to hijack.

   - Session Management Flaws - Session Fixation




2. Inspect 5 websites for which you got login credentials

   - Do these website use HSTS (HTTP Strict Transport Security)?
   - Does the session ID cookie of these websites use *Secure* and/or *HttpOnly*?

   Make a table of these websites and their support for HSTS, the name of the cookie and the usage of *Secure* and *HttpOnly* for that cookie.
   **Solution:**

| Website | HSTS | Cookie Name | Secure | HttpOnly |
|---|---|---|---|---|
| facebook.com | yes | multiple cookies (c_user, xs, sb) | yes, yes, yes | no, yes, yes |
| mail.ru.nl | no | cadata | yes | yes |
| linkedin.com | no | li_at | yes | yes |
| github.com | yes | _gh_sess | yes | yes |
| gmx.net | yes | 905a7a5991a00898953878290a55d118 | yes | yes |

Table 1: Website Information