Lesson 9:
Web Application Security.
Client interaction state management

Lesson goal

This module is intended to introduce the most important web application security principles, issues and tools. The security concerns span from the client side, through the communication link, to the server-side and its environment, taking also into account the human factor. In most cases, a given threat applies to a combination of several abovementioned elements.

Hence, we are going to make an overview of the main threats to a web application, using the OWASP ranking as the primary reference. Additionally, since some of the issues is related with the interaction state, we are going to take a look at various means of the interaction state management – including cookies, session variables and browser's local storage.

Important details to investigate

- Understanding the user interaction state maintaining mechanisms (regular cookie, session cookie, HttpOnly cookie, session variables, localStorage, sessionStorage)
- Understanding all the OWASP Top 10 threats, especially:
- Knowledge of various kinds of the injection threats – and the means of user input sanitizing provided by frameworks like Rails
- Understanding the XSS and CSRF threats

External resources in English

1) OWASP project website: https://www.owasp.org/
2) Secure Node JS Apps: https://medium.com/@tkssharma/secure-node-js-apps-7613973b6971
3) IBM Software Group: Discovering the Value of Verifying Web Application Security Using IBM Rational AppScan http://www-07.ibm.com/events/au/hacking/i/AppScan_POT_v02.pdf (rather old material, but a few the core threats are nicely visualized)
4) J. Kallin, I. L. Valbuena: Excess XSS http://excess-xss.com/ (a more detailed description of the XSS threats)

Assignment

Due to the introductory nature of this module, we are not going to delve into details of the Node.js/Express security features. Instead, let us summarize and test the main means of maintaining the client-server interaction state.

Using the skills gained in the previous exercises (including form and controller construction using Node/Express), prepare a form, which will allow submitting a number of parameters (e.g. textual or numeral attributes) and processing them (on client or on server appropriately), to demonstrate four ways of storing data:

a) long-term cookie;
b) session cookie;
c) session variable;
d) localStorage;

To confirm the successful storage, prepare a view that will present that data in a page.