Poznan University of Technology
Faculty of Computing and Telecommunications

michal.apolinarski[at]put.poznan.pl

Course: Application Security – laboratories

Lecturer: Michał Apolinarski, Ph.D.

Topic: User login, session management and password reset process

Duration (on site): 240 min.

Participants: groups of max. 2 persons

PREREQUISITES:

General knowledge of computer networks, operating systems, and databases. Basic programming skills in any language. Familiarity with forms, hashing, tokens, database design, and UML modeling[1]. Completed previous laboratory with working registration module.

GOALS:

The purpose of this laboratory is to design and implement secure mechanisms for:

- user login and logout (session destroy),
- server-side session management (or token-based session management),
- password reset ("forgotten password") feature.

Optional components features (for extra grade):

- password strength meter / advanced password policy,
- CSRF protection for all forms,
- rate limiting or account lockout mechanisms, CAPTCHA,
- device/session management (view and revoke active sessions),
- security event logging (failed logins, invalid tokens, lockouts),
- multi-factor authentication,
- enhanced transport security (forcing HTTPS with secure cookies and HSTS) – self signed certs acceptable,
- … your idea

[1] https://www.visual-paradigm.com/guide/

INSTRUCTIONS (tasks for a group of max. 2 persons)

PART A (design):

1. Using your existing project from the registration lab, extend the documentation to cover **login**, **session management**, and **password reset feature.** The document[2] should include:
    - full details of the student group, course, and exercise,
    - updated short description of the complete authentication module incl. security assumptions
    - updated functional and non-functional requirements[3] for the new features,
    - updated database structure,
    - (at least one :-)) UML sequence diagrams for: login, logout, reset request and reset completion – including validations and alternative paths.
2. Send updated draft[4] documentation to the lecturer for review.
3. Present and discuss your documentation with the lecturer.

PART B (implementation):

1. Extend your existing application to implement **login**, **session management**, and **password reset process** according to your design.
2. Prepare and send to lecturer the improved, final[5] documentation, add:
    - screenshots,
    - explanations of key implementation choices,
    - description of security mechanisms,
    - conclusions.
3. Demonstrate the working functionality (show a complete authentication module, explain your security-related decisions).

REPORT:

- Include a title page with full details of the student's group, course and exercise.
- Should be carefully edited and provide evidence of the completion of all exercises (screenshots, answers, and conclusions).
- **A complete reports must be submitted to the lecturer at least two days before the next class in which it will be presented.**

---

[2] For diagrams it's recommended to use: Draw.io, https://app.diagrams.net

[3] All communication between the client and the server that involves credentials (passwords, tokens) must be protected using HTTPS in a real deployment. In this laboratory environment, HTTPS configuration is not strictly required, but the design and documentation must clearly assume the use of HTTPS in production.

[4] Include suffix "_draft" in report filename.

[5] Include suffix "_final" in report filename.