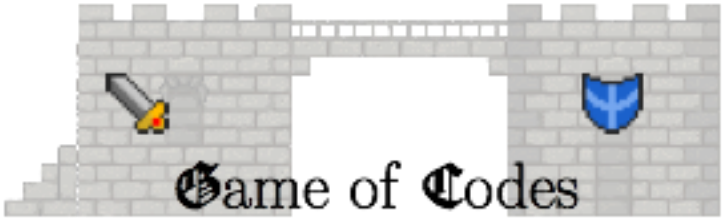
Secure Coding Project



Phase 5: Fixing vulnerabilities &
Final Report

Fixing vulnerabilities and features

Your client “the bank” wants you to:

- **Fix all the vulnerabilities** that were found by you and the team that tested your application in phase 4 (starting with the more critical vulnerabilities)

- **Fix and/or add the features from phase 1 and phase 3** which are not-working or vulnerable

Deliverables for Phase 5

- **Deadline** is (Friday) 29th January 2015 at 14:00
  → You must hand in via USB stick before this date at the office MI 01.11.040
  or on that date in the classroom (tutorial will start at 14:00 sharp)
  → There will be a 1% grade penalty for every minute after the deadline
- USB stick must contain the following files and folder:
  1. **SamuraiWTF-TeamX-Phase5.ova** (where X is your team number)
  2. Folder called **Credentials-TeamX-Phase5** contents of this folder explained on next slide
  3. **Presentation-TeamX-Phase5.pdf** (NO .PPT or .pages files)
  4. **OWASP-Checklist-TeamX-Phase5.xls (**Excel 97-2003**)**
  5. **Video-TeamX-Phase5.mp4** (NO .AVI, .MPEG or other formats)
  6. **Final-Report-TeamX-Phase5.pdf** (NO .DOCX or .pages files)

  The structure of the presentation, checklist and report are presented on the next slides
- Teams and students who present are picked randomly on the presentation date
  → If the picked student is not present s/he gets zero points for this phase (except if s/he can provide written evidence that the absence was justified)

Structure of Credentials Folder

The Credentials-TeamX-Phase5 should contain the following files:

- Non-password protected PDF files called **<username>.pdf** with the TAN numbers of existing users, which must have a transfer history and money in their accounts, where <username> indicates the username of the account to which the TANs belongs to.
- **general-info.txt** file should contain the following information:
  - the username and password of the OS user
  - the username and password of the MySQL database
  - with the username and passwords of all the existing users.
  - the username and password of the admin/employee user for your web-application
  - the URL to access your web application
  - the location and structure of the folder(s) where you have the source files of your web application.
  - the name and location of any third party libraries that you have used in phase 5.

Structure of Phase 5 Checklist

The checklist will not be presented during the lecture

- Custom version of the OWASP testing checklist https://docs.google.com/spreadsheets/d/1SAzI-y6R_XurIS7g6VUnM92Vo_w--kdBUHKZ2pPcb8s/edit?usp=sharing
- Extend the spreadsheet from phase 4 with columns H and I
- **Testing column** (H) should be filled out with information from the other team that tested your app in phase 4. Secure (green), Vulnerable (red), Not Tested (yellow), NA for Not Applicable (gray)
  - Insert note/comment for each cell that is Secure (green) or Vulnerable (red) with reference to the page in the WB testing report document where testing for this vulnerability is performed (example notes are provided for a few cells of the spreadsheet linked above)
- **The Fix column** (I) should be filled out with Fixed (green) or Not Fixed (red) values only where one or both the previous 2 columns indicate a vulnerability
  - Insert note/comment for each cell that is Fixed (green) or Not Fixed (red) with reference to the final report delivered in Phase 5, or an explanation including the source code file path and line(s) of code that were modified for this fix
- See the description of tests in the OWASP Testing Guide https://www.owasp.org/index.php/OWASP_Testing_Project

Structure of Phase 5 Presentation

MAXIMUM **5 MINUTES**

1. First slide: team number and team members (full-names)
2. Second slide: enumeration of use-cases that were **fixed / work / do not work**. At least the following use-cases should be there:
   a) All use-cases and vulnerabilities of phase 1
   b) All use-cases and vulnerabilities of phase 3
3. Third slide: enumeration of security features of your solution. Verbally describe what attacks they defend against
4. Remaining slides: **lessons learned**
   a) Summarize the things that you learned / liked during this project (especially if you think they will help you in real life if you work as a developer)
   b) These things can include: unexpected vulnerabilities, technical problems, project management issues or social issues
   c) Answer these 2 questions: Do you think your application is secure? Would you put your own money in the bank built by your team?

Video for Phase 5

MAXIMUM 5 minute VIDEO

This video will NOT be shown during the presentation!

Guidelines for making the video:

1.Showcasing the security features and other cool features of your solution

2.Make this video like you would want to sell your solution to a real bank.

3.Don’t just screencast client registration, be more creative and try to impress!

NOTE: This video will be posted on the website to help you with deciding who to vote in phase 5

NOTE: This video will be shown during the guest lecture from our sponsor (February 2nd 2016) in case you win one of the top 3 prizes. If so be prepared to give your presentation again + the video

Structure of Phase 5 Final Report

The report will not be presented during the lecture

NOTE: In the report you MUST ONLY talk about your app

1. First Page: Course title, Phase title, team number and team members (full names)
2. Second Page: executive summary
3. Third Page: table of contents
4. Fourth Page: time tracking table (info / student / task at hour granularity)

Tasks should be evenly distributed – project management part (this will be graded)

5. Application Architecture: next few pages should describe the architecture of your solution including UML diagrams, interaction with other systems and dependencies on external libraries
6. Security Measures: enumerate the security features your application uses and which attacks each feature defends against (specify which features you implemented and which you borrowed/use from other libraries)
7. Fixes: For each vulnerability discovered in Phase 4 by your team or the team that tested your application, you should summarize the fix (security measure) that you implemented:
   - Path(s) of file(s) which were modified for the fix
   - Line number(s) which were modified in each file
   - NOTE: You can use a diff utility but you should only report the changes relevant for one particular fix per vulnerability (not just a diff of all changes to phase 3).
   - Textual description of your countermeasures and why it fixes the problem (i.e. the effect that the modifications have against attack).