CptS 428/528 Software Security and Reverse Engineering

Project Deliverable 3

(Due on Canvas)

Deliverable Description

After the first two milestones, you should now have a working product (albeit probably just a first prototype) with secure design and development built in (i.e., intended to have satisfied the security plan and requirements). In particular, recall that in Milestone 2/Deliverable 1 (i.e., Deliverable 2-1), you have conducted security planning for your project, and have defined security goals and metrics. Now, the next step is to check whether and how well your analysis, design, and implementation (the code) conform to your security plan, through security validation (e.g., security-focused code review, peer review, penetration testing, static/dynamic) techniques that **break** your software product's security.

In particular, for this milestone/deliverable, you will need to

- verify your software product against security goals and metrics
- develop attacks (e.g., vulnerability exploits) to *break* your software's security, showing the presence of security defects in your product
  - note that typically if you cannot break your software, it is more likely that you have not identified sufficient ways to break it than your software is too secure to be exploitable/attackable!
- Summarize any findings you obtained during the *break* process, including what means you used and how you succeeded in breaking your software's security.
- Update your security requirements and amend your security plan in light of the above findings (i.e., strengthen your security requirements to ensure it is more secure than before)

What to Submit

a. The attack/exploit code any facilities/utilities used for breaking your software's security, including any test inputs and exploitation scripts, used for testing the security of your product resulted from Deliverable 2-2, all committed to your project code repository.

b. The milestone report, where you summarize the findings during the process of attacks/exploitations that breaks your software's security, as well as how to set up the attacks/exploits for others to reproduce the process.

c. The enhanced/improved security plan/requirements against the security flaws/vulnerabilities you identified during the security break process.

* Submit (b) and (c) together as a single PDF to Canvas (again, as stated above, code only needs to go to your project repository).

Assessment Criterion

This assignment will mainly be evaluated based on the quality of the code, clarify of the report, and the substance of the enhanced/improved requirements/plan (e.g., objectively describing whether or not the security goals are met, where they fail, and what the actionable insights are to enhance/improve the software product's security).

In particular, for the security improvement plan, the improvements are expected to include clarification and/or elaboration of relevant security goals, selection of *alternative/additional* security measures, and the choices/use of relevant techniques/tools to validate the security enhancements.