

Term Project

Why a Project?

This course covers a very large discipline, and—perhaps more so than many other areas of computer science—the discipline of computer security runs through many other areas. Because the class has a very limited amount of time, we will only touch the surface of many topics. The project is to give you an opportunity to explore one of these topics, or some other area or application of computer security that interests you, in some depth.

The Ground Rules

The project can be a detailed research paper or survey, or a programming project that focuses on validating or working with some formalism. It can be a formalism, a model, or something else theoretical that we do not cover in class. In any case, check with me before beginning to be sure it is a reasonable project and no-one else has chosen it. Please select something that interests you!

You may work individually, or in groups of up to 3 people (if you want to have more than 3, please come see me). Of course, the larger the group, the more I will expect from it.

Some Suggestions for Project and Report Topics

Below are some suggestions for projects. If you pick one of these, you will need to refine it or limit the scope of your project. You may also think of a project on your own.

- Develop a model of information flow through a network using the Take-Grant Protection Model, and demonstrate its utility by analyzing a situation of your choosing.
- Examine some of the extensions of SPM and TAM, and report on what has been learned.
- Present a survey of confidentiality models other than the Bell-LaPadula Model.
- Compare some of the secure development life cycle models such as SDLC from Microsoft, or BSIMM, with integrity models to see how well they preserve integrity constraints.
- Examine the composition problem, and focus on advances in the nature of composition and restrictiveness.
- Create a model for a specific problem, such as electronic voting, and use it to reason about properties of the desired systems.
- Insert information flow analysis into a compiler or assembler and use it to detect flows that violate a policy specifying security/integrity levels for a program or system.
- Develop a formalism or model for analyzing some aspect of the “insider problem”.
- Build a run-time system that detects flows that violate a policy specifying security/integrity levels for a program or system.
- Develop a covert channel analyzing tool and use it to analyze a subsystem or some other entity.

What Is Due and When

Please submit the following on the dates indicated:

1. *Project selection*: due on Friday, January 20; 10% of project score. Submit a write-up with your team members consisting of a one-line title of your project, a one-paragraph description, and the names of all team members. If you’re doing a programming project, state the problem you want to solve and the requirements for a solution.
2. *Progress report*: due on Wednesday, February 15; 20% of project score. Submit a one-page progress report, and a bibliography of references that you have used or plan to use.
3. *Completed project*: due on Monday, March 19 (this is the last day of instruction); 70% of your project score. Turn in your final project.

In all cases, submit the project to SmartSite as described in **All About Homework**. If a team has multiple members, only one need submit the material, and the others can simply submit a note saying who submitted the final project.