Titel

Bachelor's Thesis Proposal
by Ella Vahle

Arbeitsgruppe
Codes und Kryptographie

1 Introduction

This template contains some hints of what to write in each section. *This is just an example.* There are good reasons to deviate from it. These are just some ideas to get started.

For the introduction:

- What general topic are we interested in, why does it matter (e.g., where is it used, what problem does it solve)?

- Explain some more background for the topic, help the reader (who may have never heard of the topic) understand what you are talking about.

(Tip: make such a bullet point list for yourself (with more concrete talking points) for all sections, *then* start writing text)

Secure multiparty-computation (MPC) is a concept for securely computing functions with multiple parties involved. In this context security refers to the different parties' private inputs. The goal is to collectively execute a protocol and as a result obtain the wanted output without the necessity to reveal private data.

Secure multiparty-computation yields many practical areas of application, one being stable matching algorithms. Matching algorithms are used to match members of two different sets to each other with regards to their respective preferences. Such algorithms are used to for example match residents to residency programs or students to public schools [TODO Zitat]. A stable matching is achieved when there are no two members (one from each set) that aren't matched to each other, such that they would rather be matched to each other than to their assigned matches. Obviously the best solution would be that every member is assigned to their first preference, but in reality this is rarely possible. Stable matchings are a desirable solution in such cases to still achive a high level of member satisfaction and in some sense fairness.

One solution for computing a stable matching is the Gale-Shapley algorithm, developed by David Gale and Lloyd Shapley [TODO Zitat].

2 Current state of research

Explain what the current state of the topic is. This depends a lot on the concrete topic. For example, you could write about:

- What is a trivial (not suboptimal) way to solve x?

- How is x usually solved in practice?

- What are typical ingredients/techniques to construct the systems we are interested in?

- How does [**?**] solve the problem?

- What does the algorithm in [**?**] do (roughly)?

- . . .

3 Goals of the thesis

- What's the problem with existing solutions? (i.e. with the stuff you explained in the previous section)

- What's the goal of the thesis in a nutshell? (Probably: solve that problem/analyze the solution suggested by Eti [**?**]/compare potential solutions/. . .)

- Then go into detail. What are the specifics of what you are going to do? Explain intermediate goals.

- Explain optional goals.

With regard to optional topics I see (there are) two possible scenarios. The first one is that in the course of working with the presented protocol and it's proof of security, it becomes clear that the assumption about the protocols security in context of the presented security model doesn't hold. In this case the first step would be to examine and explain the problems that arise. The insights gained could be used to identify a different, weaker security model that solves the problems the initial protocol implied. With such an altered security model a proof can then again be attempted.
The second one is that the proof turns out to be more straight forward than expected. A reasonable course of action could be to go a step further and investigate how the proven security transfers to other security models, such as an active attacker scenario.

4 Preliminary outline of the thesis

Show your current plan for the outline ($\approx$ table of contents) of your thesis. A hierarchical list of section names suffices. Example:

1. Introduction

2. Definitions and notation

3. First topic

    a) First subsection

    b) ...

4. ...

After reading the *Goals* section, the reader should be able to understand what you're planning here.

5 Work plan

How do you plan to spend your 21 weeks of work on the thesis? For a fancy Gantt chart, consider using the `pgfgantt` package. If done well, a bullet point list is also sufficient.

Date:

Prof. Dr. Johannes Blömer	Ella Vahle