02244 Logic for Security
Project on Security Protocols

- Hand-out: Feb 5, 2024
  Hand-in: via DTU Learn until Mar 11, 2024 **noon**
- We allow to work and hand-in in groups of up to 3 students.

  Each report must indicate which students are part of the group.

  The reports must be divided into sections, and each section must have one group member designated as author. This should reflect a fair distribution in report writing among the group members. Any section without such a marking of one single author will count as not submitted.
- The report must indicate which resources have been used to perform the work. This includes text books, research papers, information found on the web, the use of any AI tools, detailed suggestions from teachers, and results of discussions or cooperation with other students.
- Page Limit: 15 pages (you may submit AnB files as attachments)
- Presentation: In the meeting on Mar 11, we ask groups to present the protocols they have designed, so we can discuss the different solutions, common mistakes, and insights gained.

A Secure Ticket Platform

Scenario: a consortium of theaters, opera houses, cinemas, and concert venue providers want to make their own independent mobile app for customers to buy tickets online and show them at the respective venue to get entrance. Of course, the consortium wants for the app as much security and privacy friendliness as possible with reasonable means.

The task of this assignment is a proposal for the design of this ticket platform which would consist the following entities:

- One (or more) servers of the venue providers that is hosting information about available seats for the different venues.
- The app of the customers.
- The devices at each venue provider that check the ticket upon entrance.
- Possibly some identity providers like MitID.
- Possibly some payment services like the bank who issued customer's credit cards or who implements other payment services like Mobile Pay.

More in detail, your report should contain a description of the following:

- The considered entities: e.g. including trust assumptions and initial distribution of keys.
- The security protocols spoken between these entities: These protocols should be formalized in the AnB language and verified with OFMC for two sessions. The report should contain:

  - the development: what flaws are found in first attempts and thus and argument for all security designs in the protocols (e.g., why is a particular encryption necessary etc.)
  - design decisions: what designs were considered, what's the reason for choosing a particular design
  - modeling in AnB: what simplifications had to be made
  - The security goals: what one wants to achieve and how this was encoded into AnB goals.

- An informal argument for privacy: that no party is learning or storing personal information that they actually do not need for their task.
- Finally, to manually analyze accountability: neither can a customer reasonably deny that they bought a ticket after they did, nor can a venue provider charge them for a ticket that they actually did not buy.

It is highly recommended to document your development while you are at it: whenever you have a version of a protocol that is syntactically accepted by OFMC and either verified or falsified with a reasonable attack[1], then make a copy of that file and a short description (e.g., what attack is found, what is the problem). In other words: keep a "lab logbook"!

Special Considerations Here are some additional questions and challenges your report should answer:

- Initial knowledge: you may well assume that all servers have a public/private key pair and that everybody knows all public keys. However you must not assume that customers initially have a public/private key pairs.
- Channels: in the course we introduce a channel notation and you can use this channel notion, but then you have to argue how this channel can be implemented without any additional knowledge.
- Trusted Parties: is it necessary to assume that some role in this protocol cannot be played by intruder without breaking the security? Justify by describing attacks that would work in this case and why without further assumptions one cannot prevent this.
- Accountability: Suppose a dishonest customer wants to deny an order/payment they have performed. Could the payment gateway or the merchant prove that the customers has performed the order in a legal court by revealing the messages exchanged?
- Typing: prove whether your protocol is satisfying the conditions of well-typedness from the lecture. If not, is there an easy way to make it well-typed? Describe this precisely in your report.

[1] If OFMC finds an attack that is simply due to a modeling mistake and does not provide any insight, one can of course skip describing that attempt.