Homework 1

Model a simple city bus system using Proverif

Deadline: 8.10.2020

1 Problem statement

The bus system can be designed as follows. The user first buys a ticket (a digital token) from the server. The user then goes to a bus validator and registers the ride. The validator confirms from the server whether the ticket is valid. If it is, then green light is emitted. If it is not, then red light is emitted.

- For simplicity, assume that a ticket is being issued once for eternity (no expiration time).
- The attacker has access to all communication channels, but we can use a pre-shared symmetric key for communication between the Bus and the Server.
- It is fine to assume a private channel between an honest user and the validator (but the model should allow the attacker to interact with the validator as well).
- The buying transaction and the light emissions are just events.
- If needed, we can assume preshared public keys that confirm Alice’s identity (which is reasonable in practice e.g. if Alice buys the ticket through the bank).
2 Requirements

The following properties can be verified by putting appropriate events into the model.

- If the ticket $T$ has not been registered, the validation of $T$ should not pass.
  - That is, on the server side, every successful validation should be preceded by a successful registration.
  - Note that registration and validation are separate processes on Server side, since the server does not keep the session established during the registration.

- If the user presents a valid ticket $T$, the validation should pass.
  - This is hard to model in ProVerif since the attacker is always able to do any kind of denial of service.
  - For simplicity, we can just check whether the “green light” event is reachable for an honest user.

- The attacker should not be able to get a free ride using an honest party’s ticket, e.g. using replay attacks.
  - If the validator says ”yes” for a ticket that belongs to an honest user Alice, then Alice should indeed have touched the validator.