Graphical animations of authentication protocols

1910414 Thet Wai Mon

Two authentication protocols Identity-Friend-or-Foe protocol (IFF) and Needham-Schroeder-Lowe Public Key protocol (NSLPK) are graphically visualized with SMGA and formally verified with CiMPG. The former is a simple protocol used to check if a principal (or an agent) is a member of a group. The latter is the revised version of Needham-Schroeder Public Key protocol (NSPK) by Lowe.

State machine graphical animation tool (SMGA) is used to graphically visualize the behaviors of the two authentication protocols. The main purpose of SMGA is to help human users be able to visually perceive non-trivial characteristics of protocols/systems by observing their graphical animations. We use Maude as a formal specification language to formally specify the two authentication protocols as state machines. Maude is based on rewriting logic. Firstly, state picture designs are carefully made for the authentication protocols to produce good graphical animations with SMGA and some finite state sequence input files are prepared based on the formal specifications of the protocols. SMGA takes a finite state sequence input file and a state picture designed for each protocol, and then produces its graphical animations by regarding the state sequence as a movie film. Observing such a graphical animation allows us to guess the characteristics of the state machine formalizing the protocol.

We confirm whether the state machine enjoy guessed characteristics because such characteristics may or may not be true properties of the protocols. One possible way to do so is model checking. Maude is equipped with model checking facilities (a reachability analyzer and an LTL model checker). We use the Maude reachability analyzer (or the search command) to model check that all characteristics conjectured are invariant properties with respect to the state machines formalizing the two authentication protocols and no counterexample is found for each characteristic at a specific depth. However, it does not guarantee that the two authentication protocols surely enjoy the desired properties because standard model checking techniques cannot handle state machines in which there are an arbitrary number of entities, such as principle. To make sure that the guessed characteristics are invariant properties of the protocols we conduct the formal verification of the two authentication protocols with CafeInMaude Proof Assistant (CiMPA) and CafeInMaude Proof Generator (CiMPG) provided by CafeInMaude, the world's second implementation of CafeOBJ in Maude, where CafeOBJ is another formal specification language and a sister language of Maude. Thus,

the two protocols are also formally specified in CafeOBJ as observational transition systems (OTSs), a kind of state machines.

CiMPA allows users to write proof scripts manually to prove invariant properties of OTSs based on their CafeOBJ specifications. The manual written proof scripts are executed with CiMPA; if CiMPA can discharge all goals successfully, then the proofs are correct and the properties are proved; otherwise we need to revise the proof scripts and make the verification attempt again until it is successful. However, writing proof scripts is not easy for non-expert users whereas working with proof scores is easier and flexible, where proof scores are programs written in an algebraic specification language, such as CafeOBJ, to conduct formal proofs. The disadvantage of working with proof scores is that proof scores are subject to human-errors because users may overlook some cases and may lead to incorrect proofs. CiMPA can address this issue but the weakness is lack of flexibility.

To overcome this weakness, CiMPG can be used. CiMPG allows users to combine the flexibility of proof score approach with the reliability of CiMPA. Once we have complete proof score, it is preferable to use CiMPG to automatically generate proof scripts for CiMPA. We need to minimally annotate proof scores to use CiMPG. If CiMPA cannot discharge all goals successfully, the proof scores are something wrong and we need to revise the proof scores, making verification attempt again. We conduct two case studies in which the two authentication protocols are formally verified using CiMPA and CiMPG that IFF enjoys the identifiable property and NSLPK enjoys the nonce secrecy property and the one to many correspondence property (a kind of authentication property).

In this thesis, we summarize some lessons learned through two case studies on how to design state picture, and how to conjecture characteristics based on our state pictures of two authentication protocols and the advantages and disadvantages of using CiMPG and CiMPA in the formal verification. We have confirmed that the characteristics we observe by observing graphical animations are the lemmas that we use in the formal verification of the two protocols which demonstrates SMGA has potential to help human users conjecture lemmas needed to conduct formal proofs. This thesis also demonstrates that SMGA can be applied to a wider class of systems/protocols, authentication protocols in particular and confirms the effectiveness of CiMPG in the formal verification of the two authentication protocols.