Typing and Compositionality for Stateful Security Protocols

On the Internet computers communicate using security protocols. For example, when using an online banking application a client may log in with NemID, ensuring that the bank can authenticate the client. Similarly, the TLS protocol provides a confidential communication channel between the client and the bank ensuring that the data they exchange are protected from eavesdropping and authenticates the bank to the client. A question is whether protocols are designed well enough so that they always achieve their security goals like confidentiality and authentication. If not, an attacker may be able to hijack and manipulate communication to break the goals, even when all honest participants only use the protocols as intended. To combat this, protocols are formally verified which provides strong guarantees that their goals cannot be violated even in the presence of dishonest participants and powerful adversaries that control the communication network. There is a rich body of research on the formal verification of security protocols, i.e., mathematically proving the correctness. However, significantly less literature exists on the composition of protocols: In the banking example the TLS protocol usually provides a secure channel over which NemID runs, that is, the protocols are used in a composed fashion. It turns out that such composed protocols might not be secure even when their component protocols have been verified in isolation. Composed protocols, however, can be rather difficult to verify since they are often too big for automated methods to verify in a reasonable amount of time. Worse still, a rather large amount of protocols are used on the Internet, and their number is growing; one cannot possibly verify the composition of all of them. To mitigate these problems, we prove several compositionality results that have the following basic form: given a number of protocols that are secure in isolation and that satisfy a number of simple prerequisites, then also their composition is secure. Hence compositionality reduces a complex protocol verification problem to easier ones.

Existing compositionality results are for relatively simple "stateless" protocols, where the behavior of the protocol participants only depends on the messages they receive within a single session. In this work we present the first compositionality result for stateful protocols, where each participant may additionally maintain databases, meaning that the behavior of the participants also depends on what is contained in the databases. For instance, a keyserver may maintain a database of currently valid public keys. As part of the protocol, participants can insert entries into, or delete entries from, their databases. Moreover, the databases may be used in more than one protocol, for instance, the keyserver may offer several different protocols for establishing new keys, and revoking and updating existing keys. The contribution of this thesis is to provide a very general result for the composition of such stateful protocols and how they coordinate the access to any shared databases. The proofs of compositionality results, however, are long and often contain subtle details. If there is a glitch in the proof of a compositionality theorem, then we may falsely rely on the security of a composition that is actually insecure. In fact, as part of this work we have found several such mistakes in existing compositionality results. The problem of flawed mathematical proofs due to their complexity is very old. A new way to mitigate this problem are proof assistants like Isabelle/HOL, that require the user to formulate a proof in such detail that a computer can check the proof. Since this relies only on the correctness of a very small core program that checks basic logical deductions, it is virtually impossible to conduct a flawed proof that is accepted by the proof checker.

Compositionality results are a lot easier to prove if one assumes a typed model where ill-typed attacks cannot happen. Typing results shows that this is a sound restriction for a large class of protocols. Such results are useful by themselves, in that they can make protocol verification faster for automated tools. They also combine very well with compositionality results, both in terms of methods and in terms of requirements on protocols. Therefore we use typing results as a basis for compositionality results. As part of this thesis we formalize an existing typing result in Isabelle/HOL and we establish and prove new compositionality results in Isabelle/HOL. This means that we can be sure that the compositionality results we present here are really correct. Moreover, if we have an Isabelle/HOL proof that a set of given protocols is each secure in isolation and satisfies the prerequisites of the compositionality theorem, then we immediately obtain a complete Isabelle/HOL proof that their composition is also secure.