Protocol composition and correctness *

Nancy Durgin John Mitchell Dusko Pavlovic

Computer Science Dept.
Stanford University
Stanford, CA 94305-9045
{nad,jcm}@cs.stanford.edu

Kestrel Institute
3260 Hillview Ave.
Palo Alto, CA 94304
dusko@kestrel.edu

April 10, 2000

1 Overview

There has been considerable research on formal analysis of security protocols, ranging from BAN logic and related approaches [BAN89, GNY90] to finite-state analysis [Ros95, MMS97] and proof methods based on higher-order logic [Pau97]. However, there appears to have been little work on the problem of composing useful protocols from standard building blocks. More specifically, those familiar with protocol design and analysis will understand that there are several common mechanisms for guaranteeing freshness, avoiding replay attacks, committing to an action without completing the action, and so on. Often, when a protocol error is discovered, the repair involves a change that is familiar from other protocols. Based on these observations, we believe that it will be useful to develop methods for composing complex protocols from simpler parts.

While we began our effort by attaching logical formulas to steps in protocol strands (based on the strand space model [FHG98]), we were soon led to a more flexible protocol formalism that includes explicit parameterization of protocol actions. We extend the strand space formalism with variables, the place holders for passing values, so that we can capture the value propagation which arises in actual communication. We distinguish static and dynamic binding operators on these variables, which regulate respectively the design time and the run time value passing.

The static parameters of our formalism are instantiated as part of protocol construction, allowing data from contiguous protocol steps to be combined. For example, a static parameter can allow data from one protocol step to be included in the next step, tying the two protocol steps together in a secure way. The static

*Partially supported by DoD MURI "Semantic Consistency in Information Exchange" as ONR Grant N00014-97-1-0505, and the Kestrel Institute.

binding also allows us to build a category, so that the protocol components (eg Guttman-Thayer's "authentication tests") can be formally composed as arrows.

Although it seems useful to assemble protocols from previously identified protocol parts, the main potential of our approach lies in the possibility of compositionally assembling protocol correctness proofs. More specifically, we present a method for attaching assertions to protocol actions, in a manner somewhat akin to Hoare logic for sequential imperative programs, so that the composition of the assertions associated with each action can provide the basis for a protocol correctness proof. The underlying logic is different from previous logics such as BAN and its descendants and from explicit reasoning about protocol and intruder as in Paulson's inductive method . In brief, the assertions associated with an actions will hold in any protocol execution that contains this action. This gives us the power to reason about all possible runs of a protocol, without explicitly reasoning about steps that might be carried out by an attacker. At the same time, the semantics of our logic is based on sets of traces of protocol execution (possibly including an attacker), not the kind of abstract idealization found in some previous logics.

Using these methods, we show how to construct a correct version of the Needham-Schroeder 2-way authentication protocol from two 1-way authentication sessions using a static parameter (one of the nonces) to tie the two protocols together. We use our logic to present a correctness proof for the composed protocol, and we also show how this same method fails to produce a proof for the original buggy version of the protocol.

References

[BAN89] M. Burrows, M. Abadi, and R. Needham. A logic of authentication. *Proceedings of the Royal Society, Series A*, 426(1871):233–271, 1989. Also appeared as SRC Research Report 39 and, in a shortened form, in ACM Transactions on Computer Systems 8, 1 (February 1990), 18-36.

[FHG98] F. Javier Thayer Fábrega, Jonathan C. Herzog, and Joshua D. Guttman. Strand spaces: Why is a security protocol correct? In *Proceedings of the 1998 IEEE Symposium on Security and Privacy*, pages 160–171, Oakland, CA, May 1998. IEEE Computer Society Press.

[GNY90] Li Gong, Roger Needham, and Raphael Yahalom. Reasoning About Belief in Cryptographic Protocols. In Deborah Cooper and Teresa Lunt, editors, *Proceedings 1990 IEEE Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society, 1990.

[MMS97] J.C. Mitchell, M. Mitchell, and U. Stern. Automated analysis of cryptographic protocols using Murφ. In *Proc. IEEE Symp. Security and Privacy*, pages 141–151, 1997.

[Pau97] L.C. Paulson. Proving properties of security protocols by induction. In *10th IEEE Computer Security Foundations Workshop*, pages 70–83, 1997.

[Ros95] A. W. Roscoe. Modelling and verifying key-exchange protocols using CSP and FDR. In *8th IEEE Computer Security Foundations Workshop*, pages 98–107. IEEE Computer Soc Press, 1995.