Specifying and Verifying Security Protocols using Temporal Logics of Knowledge [Extended Abstract]

Clare Dixon, Mari-Carmen Fernández Gago, Michael Fisher and Wiebe van der Hoek
Department of Computer Science
The University of Liverpool,
Liverpool L69 7ZF, UK
{clare,mcarmen,michael,wiebe}@csc.liv.ac.uk

Improved communication infrastructures encourage parties to interchange more and more sensitive data, such as payment instructions in e-commerce, strategic information between commercial partners, or personal information in, for instance, medical applications. Issues such as authentication of the partners in a protocol, together with the confidentiality of information, therefore become increasingly important. Consequently, cryptographic protocols are commonly used to distribute keys and authenticate agents and data over hostile networks. Verifying that particular properties do hold for these protocols, so that, for example an intruder cannot gain sensitive information under certain assumptions is critical.

The Needham-Schroeder protocol with public keys [NS78] intends to establish authentication between an agent A who initiates the protocol and an agent B who responds to A. The complete protocol consists of seven messages, but here we describe a simplified version consisting of only three messages. The messages that we omit are those whereby the agents request other agent's public keys from a server.

The protocol can then be described as the three following steps:

Message	Direction	Contents
Message 1	$A \rightarrow B:$	$\{N_A, A\}_{pub\_key(B)}$
Message 2	$B \rightarrow A:$	$\{N_B, N_A\}_{pub\_key(A)}$
Message 3	$A \rightarrow B:$	$\{N_B\}_{pub\_key(B)}$

Note that message contents of the form $\{X, Y\}_{pub\_key(Z)}$ represent messages containing both X and Y but then encrypted with Z's public key. Elements of the form N_X are special items of data, called *nonces*. Typically, agents in the protocol will generate their own unique nonce (often encrypted) which is initially unknown to all other agents. We would like to be able to specify such security protocols including their underlying assumptions, for example that agents only know their own private keys, in a suitable logic. Then verification that particular properties hold would be carried out using proof within this logic.

The application of logical tools to the analysis of security protocols was pioneered by Burrows, Abadi and Needham. In [BAN89] and [GNY90] specific belief logics, collectively referred to as BAN logics, were proposed to deal with authentication issues. The rules in these logics are such as "if a principal is told a formula encrypted with a public key and he possesses the corresponding private key he then is considered to have been told the decrypted contents of that formula". These logics have been re-formulated and given a possible world semantics in [AT91].

In general BAN logics do not deal explicitly with time but there is an implicit assumption, part of the semantics, that beliefs cannot decrease over time. Several authors have tried to explicitly incorporate time into BAN logics, for example [Syv93], which allows simple past time operators *always in the past* and its dual *sometime in the past* to impose some order on events. This paper uses the temporal operators to specify the order messages are sent in a particular protocol, for example an agent may only send the second message in the protocol if he received the first message. A potential attack is discussed.

We aim to show the suitability of temporal logics of knowledge to specify security protocols. This is the fusion of the multi-modal logic S5 for knowledge with propositional linear time temporal logic.

We show how the above Needham-Schroeder protocol can be specified in this logic in [DGFvdH03]. For example the following axiom states that agent A never forgets his own private key.

$$K_A val\_priv\_key(A, V) \Rightarrow \bigcirc K_A val\_priv\_key(A, V)$$

Here K_A is an epistemic operator meaning agent A knows, $val\_priv\_key(X, V)$ is satisfied if the value of the private key of X is V and $\bigcirc$ is the temporal operator meaning *in the next moment in time*. We then prove properties of the protocol using resolution based proof for temporal logics of knowledge [DFW98]. For example we can prove simple statements such as "*once B receives the nonce of A encoded by B's public key then B knows the nonce of A*". Until now proofs have been carried out by hand however, as part of the EPSRC project (GR/R45376), we hope to use the resolution theorem prover, TRP 2.0, for monodic first-order temporal logics [HK03], to carry out the proofs automatically by translating the modal part into a fragment of first-order logic. Further, we hope to specify and verify other security protocols, thus building up a set of axioms describing the environment as well as the differing axioms relevant to each protocol.

One of the advantages of using a standard combination of temporal and modal logics is that there is a clear semantics for this logic. This was a problem with the early BAN logics. Secondly there is an existing bank of work relating to axiomatisations, complexity, proof methods etc that can be applied. Some existing approaches that use particular combinations of temporal and modal logics (for example [GMP92] which uses a combination of a simple branching-time temporal logic, and modal logics for knowledge permission and obligation) can specify protocols in their language but have no proof method for the relevant logic and reasoning about protocols is carried out using semantic arguments. Finally the temporal logics that have been used to specify security protocols for example [Syv93, GMP92] only allow the temporal operators always and sometime (or their past-time equivalents) and don't have the richer temporal language we use. We have yet to see whether these additional operators are really useful in specifying security protocols.

References

[AT91] M. Abadi and M. R. Tuttle. A Semantics for a Logic of Authentication. In L. Logrippo, editor, *10th Annual ACM Symposium on Principles of Distributed Computing*, pages 201–216. ACM Press, August 1991.

[BAN89] M. Burrows, M. Abadi, and R. Needham. A Logic for Authentication. In *Proceedings of the Royal Society of London*, volume 426, pages 233–271, 1989.

[DFW98] C. Dixon, M. Fisher, and M. Wooldridge. Resolution for Temporal Logics of Knowledge. *Journal of Logic and Computation*, 8(3):345–372, 1998.

[DGFvdH03] C. Dixon, M.-C. Fernandez Gago, M. Fisher, and W. van der Hoek. Using temporal logics of knowledge in the formal verification of security protocols. Technical Report ULCS-03-022, University of Liverpool, Department of Computer Science, 2003. `http://www.csc.liv.ac.uk/research/techreports`.

[GMP92] J. Glasgow, G. MacEwen, and P.Panangaden. A Logic to Reason About Security. *ACM Transactions on Computer Systems*, 10(3):226–264, August 1992.

[GNY90] L. Gong, R. Needham, and R. Yahalom. Reasoning about Belief in Cryptographic Protocols. In *Proceedings of the IEE Computer Societey Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society Press, 1990.

[HK03] U. Hustadt and B. Konev. TRP 2.0: A temporal resolution prover. In *Proceedings of 19th International Conference on Automated Deduction (CADE)*, LNCS. Springer, July/August 2003.

[NS78] R.M. Needham and M.D. Schroeder. Using Encryption for Authentication in Large Networks of Computers. *Communications of the ACM*, 21:993–999, 1978.

[Syv93] P. Syverson. Adding Time to a Logic of Authentication. In *Proceedings of the First ACM Conference on Computer and Communications Security*, pages 97–101. ACM Press, 1993.