Formal Verification of Communication Protocols (Extended Abstract)

Mark A. S. Smith*
mass@lcs.mit.edu
Laboratory for Computer Science, MIT

June 19, 1996

The original motivation for this work was to do a formal verification of an experimental transport level protocol called T/TCP. This protocol, by Braden and Clark [5, 3, 4], is designed to be a *unified transport protocol* in that it should work well for both transactions and streaming. A transaction is typically a request from a client and a response from a server. Streaming on the other hand is the sending of significant amounts of data. The idea behind the design of T/TCP is to extend the Transmission Control Protocol (TCP) to make it efficient for transactions (hence the name T/TCP).

TCP is the most commonly used transport level protocol on the Internet. The basic service that it provides is reliable *end-to-end* delivery of data between application programs. On the Internet packets sent from one user to another may get duplicated, lost, or arrive out of order. TCP ensures that these packets are delivered to the application programs without duplication, without loss, and in the correct order. While TCP works well for data streaming, it does not work well for transactions because it has an open phase (the three-way handshake protocol) that forces two round trips across the network for a client to send a request and get a response from a server. Ideally we would like the request and response to be done in one round trip across the network.

The designers of T/TCP believed their protocol was correct since it is based on TCP, but the changes they made were sufficiently complex to make them uncertain. Therefore, they thought a formal correctness proof would be useful [5]. Our initial plan of attack for verifying T/TCP was to assume the correctness of TCP and leverage off this correctness in the verification of T/TCP. However, we could not find any work that verified TCP in sufficient generality to use in our work. Other works have verified parts of TCP or protocols similar to TCP. In [11], Søgaard-Andersen, Lampson, and Lynch formally verify the correctness the five packet handshake protocol of Belsnes [2] which forms the basis of the open and close phase of TCP and ISO-TP4. However, this work does not verify enough of TCP for us to use directly in the verification of T/TCP. Murphy and Shankar in [9] also specify and verify a connection management protocol for the transport layer, but the protocol is of their own design, not TCP.

The informal specification of TCP [10] is quite complicated, and an important contribution of this work is the presentation of a precise specification of the transport level problem TCP is supposed to solve. The specification can be viewed as a "black box", which has a user interface that gets all the inputs that the protocol receives and sends out all the outputs that we want the protocol to produce. The specification defines a relationship on the inputs and outputs that gives precisely the desired behavior any protocol solving the problem should have. The user interface for TCP and our specification, S, is shown in Figure 1. In our formal presentation of TCP we do make some simplifications. For example, we do not include security parameters, and the congestion control aspect of

*Mark Smith is advised by Nancy Lynch and is supported by Air Force Contract AFOSR F49620-92-J-0125, NSF contract 9225124CCR, and ARPA contracts N00014-92-J-4033, F19628-95-C-0118, and DABT63-94-C-007.

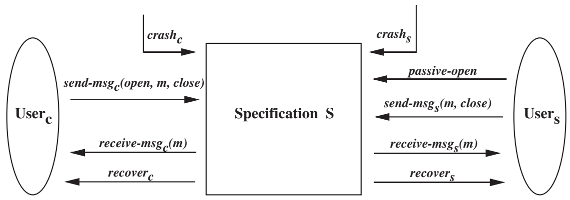

Figure 1: The user interface for TCP/IP transport level protocols.

TCP. We also assume a client/server model which means one side is always active and the other passive, whereas in full TCP either side can initiate communication. Even with these simplifications we know of no other work that attempts to verify TCP at the level of generality we do in this work. After specifying the problem and formally verifying TCP, the next step in our verification of T/TCP was to show that it implements TCP. However, we discovered that under certain circumstances T/TCP does not behave the way TCP does, and in fact does not satisfy the specification we have for TCP. More specifically, when there is a crash T/TCP may deliver duplicate data. In fact we are able to show in our work that it is impossible for any protocol to give the performance T/TCP was designed to achieve without the possibility of delivering duplicate data. Murphy [8] has also found a situation different from the one we discovered where T/TCP does not behave as TCP. The designers of T/TCP and other network protocol designers that we have spoken to do not seem to think that this behavior of T/TCP is necessarily bad. Therefore, in our work we formulate a weaker specification that allows these behaviors.

We use invariant assertion and simulation (refinement) techniques to verify TCP. We use the formalization of simulations developed in [7, 6] by Lynch and Vaandrager. These methods are used for proving trace inclusion relationships between concurrent systems. The methodology is developed in the context of a very simple and general automaton for both untimed [7] and timed [6] systems. Simulation techniques are known to be quite useful in the verification of concurrent systems, and many other researchers use this method in their work [1, 9, 11]. Our work is closest in scope to the work in [11] and in fact builds on that work.

References

[1] M. Abadi and L. Lamport. The existence of refinement mappings. *Theoretical Computer Science*, 82(2):253–284, 1991.

[2] D. Belsnes. Single message communication. *IEEE Transactions on Communications*, 24(2), February 1976.

[3] R. Braden. Extending TCP for transactions – concepts. Internet RFC-1379, July 1994.

[4] R. Braden. T/TCP – TCP extensions for transactions – functional specification. Internet RFC-1644, July 1994.

[5] R. Braden and D. Clark. Transport protocols for transactions and streaming. Unpublished manuscript, March 1993.

[6] N. Lynch and F. Vaandrager. Forward and backward simulations - part II: Time-based systems. To Appear in Information and Computation.

[7] N. Lynch and F. Vaandrager. Forward and backward simulations - part I: Untimed systems. Technical Memo MIT/LCS/TM-486, M.I.T., May 1993.

[8] S. Murphy. Private communication, March 1996.

[9] S. Murphy and A. Shankar. Connection management for the transport layer: Service specification and protocol verification. Technical Report UMIACS-TR-88-45.1, University of Maryland, June 1988. Revised December. 1989.

[10] J. Postel. Transmission conrol protocol - darpa internet program specification (internet standard STC-007). Internet RFC-793, September 1981.

[11] J. Søgaard-Anderson, N. Lynch, and B. Lampson. Correctness of communications protocols, a case study. Technical Report MIT/LCS/TR-589, M.I.T., November 1993.