Refactoring Transport for the Next Century

Ken Calvert
University of Kentucky

November 2014

1 Context

The idea of "refactoring" transport protocol functionality and implementing it in a more flexible and extensible way has been the subject of a good deal of academic research, going back at least 20 years [4, 1, 3, 8, 13, 10].[1] Unfortunately, none of the proposed solutions has achieved significant deployment in the global Internet. (Although various frameworks for building web services—SOAP, TWISTED, etc.—are in pretty widespread use, they are generally language-specific and run "above" the application-layer protocol.) One plausible explanation is that the perceived value of the proposed framework(s) has never been sufficient to offset the cost of deployment. That may be changing, due to several factors including increasing diversity of application needs, more challenging operational environments, and evolving security considerations (e.g., pervasive monitoring).

As noted by the workshop organizers, a second barrier to evolving the stack is the presence of components in the infrastructure (middleboxes) that actively *enforce assumptions* about protocols above the Internet layer—effectively precluding the use of transport protocols other than TCP and UDP (and possibly SCTP). The prevalence of such elements in the network is now simply a fact of life, and may actually be increasing. However, this problem seems to have received less attention from the academic research community. The workshop organizers have it exactly right: the problem of "stack evolution" cannot be solved in the absence of a solution to the middlebox problem, even if evolution is confined to protocols *above* the network layer (and thus—theoretically—only affecting end-systems).

Middleboxes are deployed to enforce the policies of network stakeholders and to perform functions, such as NAT, related to the network layer. Their reliance on assumptions about protocols above the network layer, and their problematic behavior of terminating any unrecognized protocols, result naturally from the current architecture's lack of *explicit mechanisms* to support the stakeholders' policy goals. Although some protocols for detecting and controlling middleboxes have been standardized—MIDCOM, STUN, TURN, ICE, etc. [17, 16, 11, 15]—they were developed with limited contexts and/or specific applications in mind, and lack the generality needed for a 21st-century Internet.

2 Position Statement

1. *The stack evolution problem should be solved once.* A durable, robust approach should be application-independent, should allow for the addition of new transport functionality, and should enable implementation strategies to evolve for the foreseeable future.

[1] This is by no means an exhaustive list.

2. In particular: *the solution should support evolution below the transport level as well as above.* Development of new Internet architectures is an active area of research [20, 21, 22]; even though it may take many years, eventually there will be a new network layer. Some of these network layers support rather different service models (e.g., “pull” vs. “push”). At the same time, deploying an API that supports new transport capabilities is costly, even if it is backward compatible: applications must eventually be modified to take advantage of new capabilities, though that may not happen immediately. Designing for “waist evolution” will significantly ease deployment challenges for the next generation; the challenge is to determine what that really looks like, given the diversity of approaches being studied.

3. A key component of a stack evolution solution is a *general-purpose mechanism* for negotiating, determining, and verifying policy compliance of end-to-end network traffic. The relevant architectural principle here is “design for tussle” [5]: Middleboxes are the manifestation of a tussle among providers and users—one in which the providers currently hold all the power. A system allowing the policies of *all* stakeholders to be expressed in the selection of paths through the network is highly desirable, and could motivate deployment of new protocols.

Obviously such a system raises several very challenging issues. One is that the current architecture does not admit any notion of *path*; the closest thing is a source-destination address pair, a la multipath TCP. Another is the question of what kinds of policies can be expressed, and how they are represented in the system. A third consideration is the mode of operation of such a system and the overhead it imposes. For example, is compliance checked/negotiated in-band, or out-of-band? At “run-time”, or in advance? Are trusted third parties required? Finally, there is the important question of how providers could possibly be motivated to give up their control, and allow users’ policies to be considered?

As a thought experiment, consider a system with the following high-level architecture. Each autonomous system runs a policy-negotiation service, which knows about the middleboxes within that domain and the policies they enforce. Such policies might deal with traffic engineering, application throttling, access control, content distribution, etc. At flow initiation time, the originating end-host contacts the local service and presents its desired destination, together with its policies (which might include a description of the application plus “hints” about desired path characteristics). The policy service would first check its local policies to determine if the request complies, and if so, which neighboring domain the traffic would flow through (if any). It would then contact that neighboring domain’s policy service, and present the *conjunction* of the user request with its own policy to the neighboring service, which would repeat the process recursively. If the user request complies with all policies along the interdomain path, a positive response is returned. The response contains a token that can be used as “proof of policy-compliance” for middleboxes along the path. The originating user presents the token along with the flow (initially, or in every packet, e.g., as a flow-id), to prove to middleboxes the policy-compliance of the flow. (Similar schemes have been described by others for various purposes [9, 14].) If such a scheme were deployed in today’s Internet, it obviously would not include the in-band compliance check by middleboxes; they would continue to operate based on legacy protocol assumptions. However, the *application would already benefit* by early and explicit indication of whether its flow will conform to the policies along the path to the destination. Other capabilities—selection among alternative paths, in-band compliance checks—could be added incrementally.

The ChoiceNet project [23] aims to incent innovation in the network infrastructure through competition. The general approach is to develop mechanisms that support: (i) presentation and selection of alternatives in network services; (ii) more general and direct flow of compensation from users to service providers (an “economy plane” [18]); and (iii) verification of services received. We believe that a mechanism for direct flow of compensation from users to service providers can be especially important as an incentive for the latter to adopt a system in which users can express their policies [12] (and potentially even select paths [2]).

References

[1] M. B. Abbot and L. L. Peterson, "A Language-Based Approach to Protocol Implementation" *IEEE/ACM Transactions on Networking*, 1(1), August 2002, pp.4–19.

[2] O. Ascigil, K. L. Calvert, and J. N. Griffioen, "On the Scalability of Interdomain Path Computations", *Proceedings IFIP Networking 2014 Conference*, Trondheim, Norway, June 2014.

[3] N. T. Bhatti, M. A. Hiltunen, R. D. Schlichting, and W. Chiu, "Coyote: A System for Constructing Fine-grain Configurable Communication Services", *ACM Transactions on Computer Systems*, 17(4), Nov. 1998, pp.321–366.

[4] K. L. Calvert, "Beyond Layering: Modularity Considerations for Protocol Architectures", *Proceedings 1993 IEEE International Conference on Network Protocols (ICNP '93)*, San Francisco, October 1993, pp. 90–97.

[5] D. D. Clark, J. T. Wroclawski, K. R. Sollins, and R. Braden, "Tussle in Cyberspace: Defining Tomorrow's Internet", *IEEE/ACM Transacations on Networking*, 13(3), June 2005, pp.462–475.

[6] R. J. Clark, M. A. Ammar, and K. L. Calvert, "Protocol Discovery in Multiprotocol Networks", *Balzer/ACM Mobile Networks and Applications*, 2, 1997, pp.271–284.

[7] R. J. Clark, K. L. Calvert, and M. A. Ammar, "Multiprotocol Interoperability In IPng", Request for Comments 1683, August 1994.

[8] R. V. Clayton and K. L. Calvert, "A Reactive Implementation of the Tau Protocol Composition Framework", *Proceedings of IEEE OpenArch '98*, San Francisco, April 1998, pp.101–114.

[9] X. Zhang et al., "SCION: Scalability, Control, and Isolation On Next-Generation Networks", *Proceedings of 2001 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2011.

[10] H. Miranda, A. Pinto, and L. Rodrigues, "Appia, a Flexible Protocol Kernel Supporting Multiple Coordinated Channels, *Proceedings of The 21st Intl Conf. on Distributed Computing Systems (ICDCS-21)*, Phoenix, April 2001, pp.707–710. IEEE Computer Society.

[11] R. Mahy, P. Matthews, and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, April 2010.

[12] A. Nagurney, D. Li, S. Saberi, and T. Wolf, "A Dynamic Network Economic Model of a Service-Oriented Internet with Price and Quality Competition", in *Network Models in Economics and Finance*, Springer, September 2014, pp.239-264,

[13] R. van Renesse et al, "Horus: A Flexible Group Communications System", Technical Report TR95-1500, Department of Computer Science, Cornell University, April 1996.

[14] B. Raghavan, P. Verkaik, and A. Snoeren, "Secure and Policy-Compliant Source Routing", *IEEE/ACM Transactions on Networking*, 17(3), December 2008, pp.764–777.

[15] J. Rosenberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[16] J. Rosenberg, R. Mahy, P. Matthews, and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[17] P. Srisuresh, J. Kuthan, J. Rosenberg, A. Molitor, and A. Rayhan, "Middlebox communication architecture and framework", RFC 3303, August 2002.

[18] T. Wolf, J. Griffioen, K. Calvert, R. Dutta, G. Rouskas, I. Baldine, and A. Nagurney, "ChoiceNet: Toward An Economy Plane for the Internet", *ACM SIGCOMM Computer Communication Review*, 44(3), July 2014, pp.58–65.

[19] X. Wu and J. Griffioen, "Supporting Application-Based Route Selection", *Proceedings 2014 International Conference on Computer Communications and Networks*, Shanghai, August 2014, pp.1–8.

[20] Named Data Networking Project, www.named-data.net.

[21] Expressive Internet Architecture Project, www.cs.cmu.edu/ xia/.

[22] MobilityFirst Project, mobilityfirst.winlab.rutgers.edu.

[23] www.ecs.umass.edu/ece/wolf/ChoiceNet/.