# UDP Encapsulation: Framework Considerations

## Position Paper
## SEMI 2015: IAB Workshop on Stack Evolution in a Middlebox Internet

David L. Black, Ph.D. (EMC Corporation)

## Abstract

UDP encapsulation has emerged as an important vehicle for new transport protocol innovation, experimentation and deployment because UDP has little functionality of its own that could interfere with new protocols and UDP generally passes through middleboxes (e.g., NATs, firewalls) that pose obstacles to direct usage of new protocols. However, specifying the use of UDP encapsulation in new IETF protocol standards may not be easy, due to considerations involving congestion control and the conditions under which it is safe to use zero UDP checksums with IPv6.

The author is one of the leaders of a design team that is working through these concerns for two UDP-encapsulated protocols, MPLS-in-UDP and GRE-in-UDP; based on progress to date, the design team is likely to succeed in completing these protocol specifications by the time of the workshop. Nonetheless, the design team's experience should serve as a warning for the IETF; the amount of time and effort expended and the level of talent involved are unlikely to scale up to the widespread use that is likely to want to be made of UDP encapsulation in the IETF. This position paper describes the design team's experience to date, explains concerns with scaling up that approach to many more protocols, and makes suggestions for how this situation could be improved.

This position paper represents solely the views of the author, which may not be shared by other members of the design team, the author's employer or anyone else.

## 1. UDP Background

Middleboxes and limited operating system protocol support are facts-of-life of the current Internet, and there are a limited number of transport protocols that are widely available to applications and provide ubiquitous (or nearly so) reachability in this environment. TCP and UDP are two of the most important such protocols, and SCTP (dating back to RFC 2960, October 2000) is an example that demonstrates the difficulties in widely deploying a new transport protocol directly over IP.

In contrast, UDP encapsulation has a long history of being able to get other protocols from point A to point B (for nearly arbitrary A and B) in the Internet; this dates back to at least the 2005 use of UDP for IPsec NAT traversal (RFC 3947 and RFC 3948, January 2005). To a first approximation, anywhere that IP traffic can be sent, UDP is likely to work, and UDP does very little other than port-based mux-demux of packets. UDP encapsulation of SCTP is now a key enabler for the expanding usage of SCTP, e.g., by the WebRTC data channel (draft-ietf-rtcweb-data-channel-12).

The importance of UDP encapsulation to SCTP (as well as other protocols like DTLS) is testament to the suitability of UDP as a substrate for more interesting transport protocol work. UDP encapsulation is an

important technique that can be expected to be widely used in new protocol designs, and hence ought to be easy for protocol designers to use; the author's recent experience unfortunately suggests that this is not currently the case for IETF protocol specifications.

## 2. MPLS-in-UDP

In January of 2014, a relatively short (6 pages of actual protocol specification) and seemingly straightforward draft on encapsulation of MPLS in UDP (draft-ietf-mpls-in-udp) was sent to IETF Last Call. That Last Call failed in a dramatic fashion with two primary areas of concern where consensus (rough or otherwise) was not achieved: congestion control (or lack thereof), and use of zero UDP checksums with IPv6.

The congestion control concerns stem from a combination of a) UDP's ability to reach almost anywhere in IP-based networks and b) MPLS's ability to carry significant amounts of traffic that is not congestion controlled. In principle, that combination could carry significant amounts of non-congestion-responsive traffic into congested (or potentially congested) areas of networks that rely on transmitter backoff responses to congestion signals (packet drops or ECN marking). See RFC 2914 for a longer discussion of why this could be problematic, and RFC 5405 for current guidelines that are intended to avoid these problems (both RFCs are Best Current Practice documents).

The UDP checksum concern arises from the absence of an IPv6 header checksum, as IPv6 relies on link-layer and transport layer checksums or other integrity checks to protect its header. This raises concerns for IP forwarding in routers, where there is no link-layer; the initial result was to require use of the UDP checksum with IPv6 (RFC 2460). At high data rates, computation and verification of the UDP checksum may entail significant implementation costs, so the requirement for use of the UDP checksum has recently been relaxed for UDP-based tunnels (RFC 6935) that meet a significant set of requirements (RFC 6936) intended to prevent packet mis-delivery based on header corruption. The MPLS-in-UDP draft that was sent to IETF Last Call did not require UDP checksum usage with IPv6 nor did it indicate how it met RFC 6936's requirements.

## 3. The Design Team

As this position paper is being written, about 9 months later, these concerns are (finally) well on their way to resolution, (see draft-ietf-mpls-in-udp-07), based on significant work by a cross-area (Routing and Transport) design team organized by the author. That design team also addressed the same concerns for the GRE-in-UDP draft (draft-ietf-tsvwg-gre-in-udp-encap); while the -03 version of that draft represents significant progress, it does not (yet) completely address these concerns.

This position paper was written because the design team effort has taken too long with too much work by too many talented people; a process that requires such a level of effort and expertise will not scale up to what is likely to be needed for UDP encapsulation to become an effective protocol design tool. Specifically, beyond the draft authors (Xiaohu Xu and Lucy Yong were the lead authors for the MPLS-in-UDP and GRE-in-UDP drafts, respectively), significant time and effort was invested by three WG chairs (Ross Callon [mpls], David Black [tsvwg] and Gorry Fairhurst [tsvwg]) with key inputs from both the Routing and Transport ADs, as well additional experts.

After some important groundwork was laid by the ADs during the London IETF-89 and Toronto IETF-90 meetings, the design team was formed at the Toronto IETF-90 meeting, and took the better part of three months to work through these concerns, resulting in revised drafts becoming available for the Honolulu IETF-91 meeting.

## 4. Why is this Hard?

At first blush, one could assign responsibility for the concerns and their resolution to the draft authors. After all, one of the IETF's middle names is "Engineering", there is extensive guidance on what needs to be done and why in RFCs 2914, 5405, 6935 and 6936, and there is reasonably broad familiarity with both congestion control and header corruption concepts and concerns in the IETF community as a whole.

The author of this position paper strongly disagrees with that assignment of responsibility based on the design team's experience in working through those concerns. The following factors provide strong indications that the actual situation is more complex:

- The protocol specification contents of the MPLS-in-UDP draft have increased in size by over 50% just to address these two concerns (which are not central to the protocol design);
- The design team work involved took months (not just a couple of weeks) despite the level of expertise applied; and
- The MPLS-in-UDP results are not immediately applicable to GRE-in-UDP, suggesting that this is not a one-time effort.

RFCs 5405, 6935 and 6936 are written for application protocol designers. From a strict standpoint, neither MPLS nor GRE is an application protocol; both are effectively encapsulations, so the MPLS-in-UDP and GRE-in-UDP drafts specify dual encapsulations (or tunnel-in-tunnel designs). In particular, these RFCs assume that the protocol designer has significant knowledge of the carried traffic; that is not generally true for tunnel encapsulations, which usually specify the tunnel independent of the traffic that it carries. That makes these RFCs more difficult to apply to these two UDP encapsulations.

The design team considered (on multiple occasions) whether to simply require (i.e., specify "MUST use") congestion control and/or use of the UDP checksum with IPv6; the decision was not to do so because:

- There are important uses of these protocols with traffic that is not congestion-controlled or that is not known to be congestion controlled.
- There are interaction risks in applying congestion control to congestion controlled traffic; the two control loops could interact in undesirable ways.
- There are implementation advantages (optimization) in not using the UDP checksum with IPv6.
- There is solid operational experience with non-UDP-encapsulated traffic that is not congestion-controlled and/or does not use an additional checksum or integrity check above the IPv6 header.

In full generality (for the public capital-I Internet as a whole) these concerns do not justify setting aside the four RFCs noted above. The design team eventually settled on an approach that rejected that "full generality" premise and focused on applicability of MPLS-in-UDP (and GRE-in-UDP) to network

environments in which traffic engineering mitigates much of the congestion control concern and the quality of network management and operation helps mitigate the corruption concern. Beyond that, the design team specified additional checks on the contents of received packet headers to increase the likelihood that packets with corrupt headers will be detected and discarded instead of mis-delivered (the goal was that a single corrupt header field be insufficient to cause mis-delivery - an offsetting corruption, or more than one, is necessary elsewhere, which is much less likely to occur).

New protocol designs may have more flexibility than tunnel-in-tunnel protocols based on existing (inner) encapsulations, but these two design concerns will remain significant. Congestion control is a fundamental aspect of the Internet architecture [RFC 2914], and coexistence of new congestion control techniques with existing ones will be of concern to designers of new transport protocols. Beyond that, the efficiency motivation to not use the UDP checksum is unlikely to ever go away.

## 5. How Could This Situation Be Improved?

Looking back on the design team's experience, the key RFCs that provide design guidance, RFC 5405 and RFC 6936 are both complete, but not easy to use in protocol design. Both RFCs assume that protocol designers are willing to do significant transport-related protocol engineering work - at some level that's reasonable, as both congestion control and use of UDP checksums with IPv6 are effectively "SHOULD use" requirements, so these RFCs are comprehensive explanations of: "the full implications must be understood and carefully weighed before choosing a different course" (text quoted from RFC 2119 definition of "SHOULD"). On the other hand, that's a lot to ask of protocol designers in areas other than transport. Towards more effective use of UDP encapsulation, here are some suggestions for IETF actions:

1) Additional guidance on use and application of RFC 5405 and RFC 6936. Both RFCs contain lists of guidelines; decision trees on combinations may better help designers consider their options.
2) A shorter more accessible introduction to at least these areas of concern could be useful, perhaps draft-ietf-tsvarea-udp-encap-for-dummies? Such a draft could make it clear why congestion control and UDP checksum usage with IPv6 are both "SHOULD use" requirements.
3) Documented reusable solutions or design approaches. The IANA Considerations RFCs (RFC 5226 and RFC 2434 before it) made it much easier to write good IANA Considerations by providing procedures that could be referenced. Perhaps a similar approach to acceptable designs would help here, especially for use of zero UDP checksums with IPv6.
4) Some good examples. The author would like to hope that both MPLS-in-UDP and GRE-in-UDP would be on that list.

## Author

David L. Black, Ph.D.
EMC Corporation
176 South Street
Hopkinton, MA 01748

Email: david dot black at emc dot com