Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 16, 2015

T. Hardie
Google

February 12, 2015

Use Cases for SPUD
draft-hardie-spud-use-cases-01

Abstract

SPUD is a prototype for grouping UDP packets together. This grouping allows on-path network devices, especially middleboxes such as NATs or firewalls, to understand some basic semantics and potentially to offer salient information about their functions or the path to the endpoints. This document describes basic use cases for sharing that semantic and for using the information shared.

Table of Contents

1. Introduction

SPUD [draft-hildebrand-spud-prototype] is a prototype for grouping UDP packets together. This grouping allows on-path network devices, especially middleboxes such as NATs or firewalls, to understand basic session semantics and potentially to offer salient information about their functions or the path to the endpoints. This document describes basic use cases for sharing that semantic and for using the information shared

1.1. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119 [RFC2119] and indicate requirement levels for compliant STuPiD implementations.

2. Application to Path Use Cases

The primary use case for application to path signaling is the indication of which packets traveling between two endpoints make up a an application-layer group, along with basic related semantics (start and stop). By explicitly signaling start and stop semantics, a flow allows middleboxes to use those signals for setting up and tearing down their relevant state (NAT bindings, firewall pinholes), rather than requiring the middlebox to infer this state from continued traffic. At best, this would allow the application to refrain from sending heartbeat traffic, which might result in reduced radio utilization (and thus greater battery life) on mobile platforms.

A use case suitable for experimentation might be the management of multiple UDP flows going between the same two endpoints. This occurs, for example, in WebRTC. There the application may be willing to disclose which UDP flows are media traffic rather than data channel traffic. Now middleboxes may now have to examine multiple encrypted packets in the SRTP packet train to infer which flows are media, so having an explicit indication might speed appropriate treatment by the network.

An application may also be willing to indicate ordinal priority among those flows which are not bundled, if it believes the network assigned priority might be inappropriate (bundling all media above all data may not, after all, match the application semantics for games or other applications). A more complex example would be the browser signaling whether it is using a particular congestion control algorithm (future RMCAT work vs. the "circuit breaker" baseline.)

Note that in none of these cases is the signaling between the application path mandatory; if elements along the path do not understand or choose to ignore these signals, the flow proceeds as before.

3. Path to Application Use Cases

The primary use case for path to application signaling is parallel to the use of ICMP [ICMP], in that it describes a set of conditions (including errors) that applies to the datagrams as they traverse the path. This usage is, however, not a pure replacement for ICMP but a "5-tuple ICMP". Since policy may cause different middleboxes to be on path for different application, the path for different applications may have both different elements and different constraints; this signaling would enable these different constraints to be transmitted to the sending application. A minimal set of such ICMP-like messages would be: the moral equivalent of "packet too big"; something like the "next-hop MTU" message; a notification of (near) congestion similar to ECN[RFC3168]; and an address-family conversion message.

A use case suitable for further experimentation might be the signaling of known network constraints. An on-path router or access point might, for example, indicate the upstream bandwidth when it would be surprising (e.g. when cellular backhaul is used).

Note again that in none of these cases is the signaling mandatory; if elements along the path do not send or the application choose to ignore these signals, the flow proceeds as before.

Because of the risk that an attacker with access to the path may send spurious signals, applications should in general "trust but verify" data received from the path. That is, the information received may form the basis of tests that confirm network conditions like the reported MTU.

4. Security Considerations

In addition to the security risks associated with spurious messages inserted by attackers noted above, it is important to note that the failure of this substrate should never result in a fallback to plaintext. For encrypted flows, if this substrate fails to perform correctly, the correct fallback is to fully encrypted flows like those carried by DTLS [RFC6347]

The privacy objective here is to enable UDP-based transports whose payload is fully encrypted to have very simple semantics exposed to the path elements which might otherwise required access to plaintext. Obviously, any exposure beyond the standard 5-tuple involves some information sharing which is not required for packet delivery. There are potential attacks that use start and stop semantics to infer known plain text for a common protocol, those they require cryptographic attacks or failures which are not common. Later versions of this document will explore the cases in which use of SPUD to expose those semantics is not appropriate.

5. IANA Considerations

This document makes no requests of IANA.

6. Acknowledgements

This document arose out of the IAB SEMI workshop. In particular, Joe Hildebrand and Brian Trammel guided the shape of the document.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2. Informative References

[ICMP] Postel, J., "Internet Control Message Protocol", September 1981, <https://tools.ietf.org/html/rfc792>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[draft-hildebrand-spud-prototype]
Trammel, B., "Session Protocol for User Datagrams Prototype", February 2015, <https://tools.ietf.org/html/draft-hildebrand-spud-prototype-01>.

Author's Address

Ted Hardie
Google

Email: ted.ietf@gmail.com