

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: June 10, 2018

B. Trammell  
M. Kuehlewind  
ETH Zurich  
December 07, 2017

**The Wire Image of a Network Protocol**  
**draft-trammell-wire-image-01**

Abstract

This document defines the wire image, an abstraction of the information available to an on-path non-participant in a networking protocol. This abstraction is intended to shed light on current discussions within the IETF on the implications on increased encryption has for network functions that use the wire image.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 10, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## **1. Introduction**

A protocol specification defines a set of behaviors for each participant in the protocol: which lower-layer protocols are used for which services, how messages are formatted and protected, which participant sends which message when, how each participant should respond to each message, and so on.

Implicit in a protocol specification is the information the protocol radiates toward nonparticipant observers of the messages sent among participants. Any information that has a clear definition in the protocol's message format(s), or is implied by that definition, and is not cryptographically confidentiality-protected can be unambiguously interpreted by those observers.

This information comprises the protocol's wire image, which we define and discuss in this document. It is the wire image, not the protocol's specification, that determines how third parties on the network paths among protocol participants will interact with that protocol.

Several documents currently under discussion in IETF working groups and the IETF in general, for example [[QUIC-MANAGEABILITY](#)], [[EFFECT-ENCRYPT](#)], and [[TRANSPORT-ENCRYPT](#)], discuss in part impacts on the third-party use of wire images caused by a migration from protocols whose wire images are largely not confidentiality protected (e.g. HTTP over TCP) to protocols whose wire images are confidentiality protected (e.g. H2 over QUIC).

This document presents the wire image abstraction with the hope that it can shed some light on these discussions.

## **2. Definition**

More formally, the wire image of a protocol consists of the sequence of messages sent by each participant in the protocol, each expressed as a sequence of bits with an associated arbitrary-precision time at which it was sent.

## **3. Discussion**

This definition is so vague as to be difficult to apply to protocol analysis, but it does illustrate some important properties of the wire image.

Key is that the wire image is not limited to merely "the unencrypted bits in the header". In particular, interpacket timing, packet size, and message sequence information can be used to infer other



parameters of the behavior of the protocol, or to fingerprint protocols and/or specific implementations of the protocol; see [Section 3.1](#).

An important implication of this property is that a protocol which uses confidentiality protection for the headers it needs to operate can be deliberately designed to have a specified wire image that is separate from that machinery; see [Section 3.3](#). Note that this is a capability unique to encrypted protocols. Parts of a wire image may also be made visible to devices on path, but immutable through end-to-end integrity protection; see [Section 3.2](#).

Portions of the wire image of a protocol that are neither confidentiality-protected nor integrity-protected are writable by devices on the path(s) between the endpoints using the protocol. A protocol with a wire image that is largely writable operating over a path with devices that understand the semantics of the protocol's wire image can modify it, in order to induce behaviors at the protocol's participants. This is the case with TCP in the current Internet.

Note also that the wire image is multidimensional. This implies that the name "image" is not merely metaphorical, and that general image recognition techniques may be applicable to extracting patterns and information from it.

### [3.1](#). Obscuring timing and sizing information

Cryptography can protect the confidentiality of a protocol's headers, to the extent that forwarding devices do not need the confidentiality-protected information for basic forwarding operations. However, it cannot be applied to protecting non-header information in the wire image. Of particular interest is the sequence of packet sizes and the sequence of packet times. These are characteristic of the operation of the protocol. While packets cannot be made smaller than their information content, nor sent faster than processing time requirements at the sender allow, a sender may use padding to increase the size of packets, and add delay to transmission scheduling in order to increase interpacket delay. However, it does this at the expense of bandwidth efficiency and latency, so this technique is limited to the application's tolerance for latency and bandwidth inefficiency.

### [3.2](#). Integrity Protection of the Wire Image

Adding end-to-end integrity protection to portions of the wire image makes it impossible for on-path devices to modify them without detection by the endpoints, which can then take action in response to



those modifications, making these portions of the wire image effectively immutable. However, they can still be observed by devices on path. This allows the creation of signals intended by the endpoints solely for the consumption of these on-path devices.

Integrity protection can only practically be applied to the sequence of bits in each packet, which implies that a protocol's visible wire image cannot be made completely immutable in a packet-switched network. Interarrival timings, for instance, cannot be easily protected, as the observable delay sequence is modified as packets move through the network and experience different delays on different links. Message sequences are also not practically protectable, as packets may be dropped or reordered at any point in the network, as a consequence of the network's operation. Intermediate systems with knowledge of the protocol semantics in the readable portion of the wire image can also purposely delay or drop packets in order to affect the protocol's operation.

### **3.3. Engineering the Wire Image**

Understanding the nature of a protocol's wire image allows it to be engineered. The general principle at work here, observed through experience with deployability and non-deployability of protocols at the network and transport layers in the Internet, is that all observable parts of a protocol's wire image will eventually ossify, and become difficult or impossible to change in future extensions or revisions of the protocol.

A network function which serves a purpose useful to its deployer will use the information it needs from the wire image, and will tend to get that information from the wire image in the simplest way possible. A protocol's wire image should therefore be designed to explicitly expose information to those network functions in an obvious way, and to expose as little other information as possible.

However, even when information is explicitly provided to the network, any information that is exposed by the wire image, even that information not intended to be consumed by an observer, must be designed carefully as it might ossify, making it immutable for future versions of the protocol. For example, information needed to support decryption by the receiving endpoint (cryptographic handshakes, sequence numbers, and so on) may be used by the path for its own purposes.

Since they are separate from the signals that drive an encrypted protocol's mechanisms, the veracity of integrity-protected signals in an engineered wire image intended for consumption by the path may not be verifiable by on-path devices; see [[PATH-SIGNALS](#)]. Indeed, any



two endpoints with a secret channel between them (in this case, the encrypted protocol itself) may collude to change the semantics and information content of these signals. This is an unavoidable consequence of the separation of the wire image from the protocol's operation afforded by confidentiality protection of the protocol's headers.

#### **4. Acknowledgments**

Thanks to Martin Thomson and Thomas Fossati for discussions that have improved this document.

This work is partially supported by the European Commission under Horizon 2020 grant agreement no. 688421 Measurement and Architecture for a Middleboxed Internet (MAMI), and by the Swiss State Secretariat for Education, Research, and Innovation under contract no. 15.0268. This support does not imply endorsement.

#### **5. Informative References**

##### [EFFECT-ENCRYPT]

Moriarty, K. and A. Morton, "Effect of Pervasive Encryption on Operators", [draft-mm-wg-effect-encrypt-13](#) (work in progress), October 2017.

##### [PATH-SIGNALS]

Hardie, T., "Path signals", [draft-hardie-path-signals-02](#) (work in progress), November 2017.

##### [QUIC-MANAGEABILITY]

Kuehlewind, M. and B. Trammell, "Manageability of the QUIC Transport Protocol", [draft-ietf-quic-manageability-01](#) (work in progress), October 2017.

##### [TRANSPORT-ENCRYPT]

Fairhurst, G. and C. Perkins, "The Impact of Transport Header Encryption on Operation and Evolution of the Internet", [draft-fairhurst-tsvwg-transport-encrypt-04](#) (work in progress), September 2017.

Authors' Addresses





Brian Trammell  
ETH Zurich  
Gloriastrasse 35  
8092 Zurich  
Switzerland

Email: [ietf@trammell.ch](mailto:ietf@trammell.ch)

Mirja Kuehlewind  
ETH Zurich  
Gloriastrasse 35  
8092 Zurich  
Switzerland

Email: [mirja.kuehlewind@tik.ee.ethz.ch](mailto:mirja.kuehlewind@tik.ee.ethz.ch)

