CORE
Internet-Draft
Intended status: Standards Track
Expires: March 8, 2015

S. Lemay
V. Solorzano Barboza
Zebra Technologies
H. Tschofenig, Ed.
ARM Ltd.

September 4, 2014

A TCP and TLS Transport for the Constrained Application Protocol (CoAP)

draft-tschofenig-core-coap-tcp-tls-01.txt

Abstract

The Hypertext Transfer Protocol (HTTP) has been designed with TCP as an underlying transport protocol. The Constrained Application Protocol (CoAP), which has been inspired by HTTP, has on the other hand been defined to make use of UDP. Reliable delivery, a simple congestion control mechanism, and flow control had been added to the CoAP protocol. UDP is a good choice for networks that do not perform any form of filtering and firewalling. There are, however, many deployment environments where UDP is either firewalled or subject to deep packet inspection. These environments make the use of CoAP brittle.

This document defines the use of CoAP over TCP as well as CoAP over TLS.

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 http://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 March 8, 2015.

Table of Contents

1. Introduction

The Internet protocol stack is organized in layers, namely data link layer, network layer, transport layer, and the application layer.

IP emerged as the waist of the hour glass and supports a variety of link layers and new link layer technologies can be added in the future, without affecting IP.

Combined with the end-to-end principle the hour glass indicates the level of protocol understanding intermediaries need to have in order to exchange forward IP packets between a sender and a receiver (absent any specific application layer entities, like proxies or caches). Having IP as the waist meant that anyone could extend the layers above the network layer in the way they wanted to communicate end-to-end, including defining new transport layer protocols (as it

was done with SCTP, and DCCP).

Unfortunately, deployments departed from this ideal architecture. When the Constrained Application Protocol (CoAP) [RFC7252] was designed it was assumed that many Internet of Things deployments

would be clean-slate. Today, we know that some deployments have to integrate well with existing enterprise infrastructure, where the use of UDP-based protocols is not well-received and firewalling use is very common.

To make IoT devices work smoothly in these demanding environments CoAP has to make use of a different transport protocol, namely TCP [RFC0793] and in some situations even TLS [RFC5246]. This document describes a shim header that conveys length information about the included payload. Modifications to CoAP are intentially avoided (e.g, to introduce optimizations).

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Shim Header

This specification defines a simple layer necessary to convey length information about the exchange payloads in a 32-bit length field indicating the number of bytes in the payload following that header.

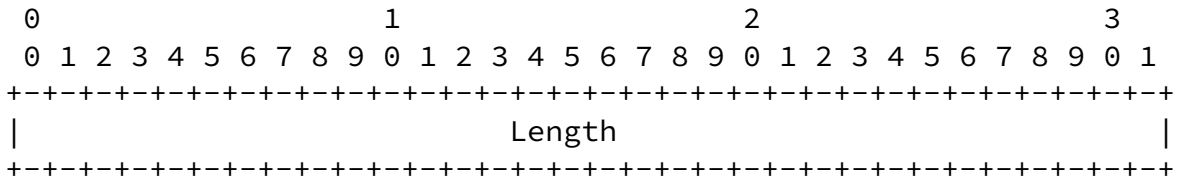

Figure 1: Shim Header.

4. Developer Considerations

The use of CoAP over a transport protocol offering reliable transmission already offers functionality that could be offered by CoAP itself. While a developer can re-use an already existing CoAP protocol stack, the use of TCP makes some CoAP features redundant.

Section 4.2 of [RFC7252] discusses the ability to convey messages in CoAP reliably as a "confirmable message", which always generates a response. It can be used without harm but does not add any value since all messages would be transmitted reliably already thanks to the features offered by TCP. A developer writing an application that runs CoAP over TCP or CoAP over TLS needs to be mindful about the changed semantic of CoAP. For example, the marking the message as

non-confirmable (see Section 4.3 of [RFC7252]) does not make the transmission unreliable but it instead saves the transmission of one CoAP message.

5. Security Considerations

This document defines how to convey CoAP over TCP and TLS. It does not introduce new vulnerabilities beyond those described already in the CoAP specification.

When CoAP is exchanged over TLS port 443 then the "TLS Application Layer Protocol Negotiation Extension" [I-D.ietf-tls-applayerprotoneg] MUST be used to allow demultiplexing at the server-side unless out-of-band information ensures that the client only interacts with a server that is able to demultiplex CoAP messages over port 443. This would, for example, be true for many Internet of Things deployments where clients are pre-configured to only ever talk with specific servers.

When CoAP over TLS is used then the use of the shim header that includes the length information is redundant since the TLS protocol headers already include length information. As such, the length header MUST be omitted when CoAP is exchanged over TLS.

6. IANA Considerations

This document requests a value from the "Application Layer Protocol Negotiation (ALPN) Protocol IDs" created by [I-D.ietf-tls-applayerprotoneg]:

Protocol: CoAP

Identification Sequence: 0x63 0x6f 0x61 0x70 ("coap")

Specification: This document.

7. Acknowledgements

We would like to thank Michael Koster, Zach Shelby, and Szymon Sasin for their feedback.

8. Normative References

[I-D.ietf-tls-applayerprotoneg]
Friedl, S., Popov, A., Langley, A., and S. Emile, "Transport Layer Security (TLS) Application Layer Protocol Negotiation Extension", draft-ietf-tls-applayerprotoneg-05 (work in progress), March 2014.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

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

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, June 2014.

Authors' Addresses

Simon Lemay
Zebra Technologies
820 W. Jackson Blvd.suite 700
Chicago 60607
United States of America

Phone: +1-847-634-6700
Email: slemay@zebra.com

Valik Solorzano Barboza
Zebra Technologies
820 W. Jackson Blvd. suite 700
Chicago 60607
United States of America

Phone: +1-847-634-6700
Email: vsolorzanobarboza@zebra.com

Hannes Tschofenig (editor)
ARM Ltd.
110 Fulbourn Rd
Cambridge CB1 9NJ
Great Britain

Email: Hannes.tschofenig@gmx.net
URI: http://www.tschofenig.priv.at