CORE   M. Boucadair
Internet-Draft   Orange
Intended status: Standards Track   T. Reddy
Expires: ~~February~~ **March** 17, 2019   McAfee
                                                           J. Shallow
                                                            NCC Group
                                                         ~~August 16,~~
                                                     September 13, 2018

             Constrained Application Protocol (CoAP) Hop Limit Option
                      ~~draft-boucadair-core-hop-limit-00~~
                      draft-boucadair-core-hop-limit-01

Abstract

   The presence of Constrained Application Protocol (CoAP) proxies may
   lead to infinite forwarding loops, which is undesirable. To prevent
   and detect such loops, this document specifies the Hop-Limit CoAP
   option.

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 ~~February~~ **March** 17, 2019.

Table of Contents

1. Introduction

   More and more applications are using Constrained Application Protocol
   (CoAP) [RFC7252] as a communication protocol between involved
   application agents. For example, [I-D.ietf-dots-signal-channel]
   specifies how CoAP is used as a distributed denial-of-service (DDoS)
   attack signaling protocol seeking for help from DDoS mitigation
   providers. In such contexts, a CoAP client can communicate directly
   with a server or indirectly via ~~a proxy.~~ **proxies.**

   When multiple proxies are involved, infinite forwarding loops may be
   experienced. To prevent such loops, this document defines a new CoAP
   option, called Hop-Limit, which is inserted **in particular** by on-path
   proxies. Also, the document defines a new CoAP Response Code to
   report loops together with relevant diagnostic information to ease
   troubleshooting.

2. Terminology

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

   Readers should be familiar with the terms and concepts defined ~~in~~
   [RFC7252].

Within this document, CoAP agent refers to both CoAP client and CoAP
proxy.

3. Hop-Limit Option

Hop-Limit option (see Section 4.2) is an elective option used to
detect and prevent infinite loops when proxies are involved. Only
one single instance of the option is allowed in a message.

The length of the
Therefore, messages carrying multiple Hop-Limit option is 1 byte. instances MUST
be rejected using 4.00 (Bad Request) error message.

The value of the Hop-
Limit Hop-Limit option is encoded as an 8-bit unsigned
integer (see Section 3.2 of [RFC7252]).

Each intermediate This value MUST be between 1
and 255 inclusive. CoAP messages received with a Hop-Limit option
set to '0' or greater than '255' MUST be rejected by a CoAP agent
using 4.00 (Bad Request) error message.

The Hop-Limit option is safe to forward. That is, a CoAP proxy which
does not understand the Hop-Limit option should forward it on.

If a CoAP proxy receives a request which does not include a Hop-Limit
option, it SHOULD insert a Hop-Limit option when relaying the request
to a next hop (absent explicit policy/configuration otherwise).

The initial Hop-Limit value SHOULD be configurable. If no initial
value is explicitly provided, the default initial Hop-Limit value of
16 MUST be used. This value is chosen to be sufficiently large to
guarantee that a CoAP request would not be dropped in networks when
there were no loops, but not so large as to consume CoAP proxy
resources when a loop does occur. Lower values should be used with
caution and only in networks where topologies are known by the CoAP
agent inserting the Hop-Limit option.

Because forwarding errors may occur if inadequate Hop-Limit values
are used, proxies at the boundaries of an administrative domain MAY
be instructed to remove or rewrite the value of Hop-Limit carried in
received messages (i.e., ignore the value of Hop-Limit received in a
message).

Otherwise, each intermediate proxy, which understands the Hop-Limit
option, involved in the handling of a CoAP message MUST decrement the
Hop-Limit option value by 1 prior to forwarding upstream if this
parameter exists.

CoAP messages MUST NOT be forwarded if the Hop-Limit option is set to
'0' after decrement. Messages that cannot be forwarded because of
exhausted Hop-Limit SHOULD be logged with a 5.06 TBA1 (Hop Limit Reached)
error message sent back to the CoAP peer. It is RECOMMENDED that
CoAP agents support means to alert administrators about loop errors
so that appropriate actions are undertaken.

To ease debugging and troubleshooting, the CoAP proxy which detects a
loop SHOULD include its information (e.g., server proxy name, server proxy alias,
IP address) in the diagnostic payload under the conditions detailed
in Section 5.5.2 of [RFC7252].

Each intermediate proxy involved in relaying a 5.06 TBA1 (Hop Limit
Reached) error message SHOULD prepend its own information in the
diagnostic payload with a space character used as separator. Only
one information per proxy MUST SHOULD appear in the diagnostic payload.

The initial Hop-Limit value SHOULD be configurable. If no initial
value is explicitly provided, the default initial Hop-Limit value of
16 MUST be used. Because forwarding errors may occur if inadequate
Hop-Limit values are used, proxies at the boundaries of an
administrative domain MAY be instructed
Doing so allows to rewrite limit the value size of Hop-
Limit carried in received messages (that is, ignore the value of Hop- TBA1 (Hop Limit received in a message). Reached)
error message, and to ease correlation with hops count.

4. IANA Considerations

4.1. CoAP Response Code

IANA is requested to add the following entries to the "CoAP Response
Codes" sub-registry available at https://www.iana.org/assignments/
core-parameters/core-parameters.xhtml#response-codes:

```
                    +------+------------------+-----------+
                    | Code | Description      | Reference |
                    +------+------------------+-----------+
                    | 5.06 TBA1 | Hop Limit Reached| [RFCXXXX] |
                    +------+------------------+-----------+
```

                          Table 1: CoAP Response Codes

4.2. CoAP Option Number

IANA is requested to add the following entry to the "CoAP Option
Numbers" sub-registry available at https://www.iana.org/assignments/
core-parameters/core-parameters.xhtml#option-numbers:

```
+--------+---+---+---+---+-----------------+-----------+
| Number | C | U | N | R | Name            | Reference |
+--------+---+---+---+---+-----------------+-----------+
|   2 TBA2  |   | x  | x x  |   | Hop-Limit       | [RFCXXXX] |
+--------+---+---+---+---+-----------------+-----------+
        C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable
```

Table 2: CoAP Option Number

5. Security Considerations

   Security considerations related to CoAP proxying are discussed in
   Section 11.2 of [RFC7252].

   **The diagnostic payload of a TBA1 (Hop Limit Reached) error message
   may leak sensitive information revealing the topology of a domain.
   To prevent that, a CoAP proxy which is located at the boundary of an
   administrative domain MAY be instructed to strip the diagnostic
   payload or part of it before forwarding on the TBA1 response.**

6. **Acknowledgements**

   **This specification was part of [I-D.ietf-dots-signal-channel]. Many
   thanks to those who reviewed DOTS specifications.**

   **Thanks to Klaus Hartke, Carsten Bormann, Peter van der Stok, and Jim
   Schaad for the review.**

7. References

6.1.

7.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained
              Application Protocol (CoAP)", RFC 7252,
              DOI 10.17487/RFC7252, June 2014,
              <https://www.rfc-editor.org/info/rfc7252>.

6.2.

7.2. Informative References

   [I-D.ietf-dots-signal-channel]
              Reddy, T., Boucadair, M., Patil, P., Mortensen, A., and N.
              Teague, "Distributed Denial-of-Service Open Threat
              Signaling (DOTS) Signal Channel Specification", draft-
              ietf-dots-signal-channel-22
              ietf-dots-signal-channel-25 (work in progress), August **September**
              2018.

Authors' Addresses

   Mohamed Boucadair
   Orange
   Rennes 35000
   France

   Email: mohamed.boucadair@orange.com
   Tirumaleswar Reddy
   McAfee, Inc.
   Embassy Golf Link Business Park
   Bangalore, Karnataka 560071
   India

   Email: kondtir@gmail.com

   Jon Shallow
   NCC Group
   United Kingdom

   Email: jon.shallow@nccgroup.trust **jon.shallow@nccgroup.com**
```