CoRE Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 10, 2020

M. Tiloca
RISE AB
E. Dijk
IoTconsultancy.nl

March 09, 2020

Proxy Operations for CoAP Group Communication
draft-tiloca-core-groupcomm-proxy-00

Abstract

This document specifies the operations performed by a forward-proxy, when using the Constrained Application Protocol (CoAP) in group communication scenarios. Proxy operations involve the processing of individual responses from servers, as reply to a single request sent by the client over unicast to the proxy, and then distributed by the proxy over IP multicast to the servers. When receiving the different responses via the proxy, the client is able to distinguish them and their originator servers, by acquiring their addressing information.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Constrained Application Protocol (CoAP) [RFC7252] allows the presence of forward-proxies, as intermediary entities supporting clients to perform requests on their behalf.

CoAP supports also group communication over IP multicast [I-D.dijk-core-groupcomm-bis], where a group request can be addressed to multiple recipient servers, each of which may reply with an individual unicast response. As discussed in Section 2.3.3 of [I-D.dijk-core-groupcomm-bis], this group communication scenario poses a number of issues and limitations to proxy operations.

In particular, the client sends a single unicast request to the proxy, which the proxy forwards to a group of servers over IP multicast. Later on, the proxy delivers back to the client multiple responses to the original unicast request. As defined by [RFC7252] the multiple responses are delivered to the client inside separate CoAP messages, all matching (by Token) to the client's original unicast request. A possible alternative approach of performing aggregation of responses into a single CoAP response would require a

specific aggregation content-format, which is not available yet. Both these approaches have open issues.

This specification considers the former approach of how the proxy forwards the individual responses to a CoAP group request back to the client. The described method addresses all the related issues raised in Section 2.3.3 of [I-D.dijk-core-groupcomm-bis].

To this end, a dedicated signaling protocol is defined, using two new CoAP options. In particular, the client can explicitly confirm its support for receiving multiple responses to a proxied unicast request, i.e. one per originator server, and for how long it is willing to wait for responses. Also, each server originating a response indicates to the client its own addressing information. This enables the client to distinguish the multiple, diffent responses by origin and to possibly contact one or more of the individual servers by a unicast request, optionally bypassing the forward-proxy.

1.1. 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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Readers are expected to be familiar with terms and concepts defined in CoAP [RFC7252], Group Communication for CoAP [I-D.dijk-core-groupcomm-bis], OSCORE [RFC8613] and Group OSCORE [I-D.ietf-core-oscore-groupcomm].

2. The Multicast-Signaling Option

The Multicast-Signaling Option defined in this section has the properties summarized in Figure 1, which extends Table 4 of [RFC7252]. The option is intended only for CoAP requests.

Since the option is not Safe-to-Forward, the column "N" indicates a dash for "not applicable". The Multicast-Signaling Option contains a timeout value in seconds, encoded as a CBOR [RFC7049] unsigned integer.

No.	C	U	N	R	Name	Format	Length	Default
TBD1	X	x	-		Multicast-Signaling	uint	1-5 B	(none)

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable
(*) See below.

Figure 1: The Multicast-Signaling Option.

This document specifically defines how this option is used by a client to indicate to a forward-proxy its support for and interest in receiving multiple responses to a proxied CoAP group request, i.e. one per originator server, and for how long it is willing to wait for receiving responses via that proxy (see Section 5.1 and Section 5.2).

The client, when sending a CoAP group request to a proxy via IP unicast, to be forwarded by the proxy to a targeted group of servers, includes the Multicast-Signaling Option in the request. The option value indicates after what time period in seconds the client will stop accepting responses matching its original unicast request, with the exception of notifications if CoAP Observe is used [RFC7641]. This allows the intermediary proxy to stop forwarding responses back to the client, if received from the servers later than a timeout expiration.

The Multicast-Signaling Option is of class I for OSCORE [RFC8613][I-D.ietf-core-oscore-groupcomm], in order to allow the proxy to access its value as intended consumer.

3. The Response-Forwarding Option

The Response-Forwarding Option defined in this section has the properties summarized in Figure 2, which extends Table 4 of [RFC7252]. The option is intended only for CoAP responses, and builds on the Base-Uri option from Section 3 of [I-D.bormann-coap-misc].

Since the option is not Safe-to-Forward and is intended only for responses, the column "N" indicates a dash.

No.	C	U	N	R	Name	Format	Length	Default
TBD2	X	x	-		Response-Forwarding	(*)	8-20 B	(none)

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable
(*) See below.

Figure 2: The Response-Forwarding Option.

This document specifically defines how this option is used by a server, when it receives a request originated by a client and forwarded by a proxy over IP multicast. The server uses the option to indicate its own addressing information to the originator client, when sending its own response to the proxy (see Section 5).

When replying to a multicast request received via a proxy, the server includes the Response-Forwarding Option in the response sent to the client via that proxy. The option value includes addressing information of the server, that the client can use to identify the response originator and possibly send later unicast requests to directly, or via the proxy as CoAP unicast request.

The value of the option is the unicast IP address of the server, encoded as a CBOR byte string. The byte string is in turn tagged and identified by the CBOR tag 260 "Network Address (IPv4 or IPv6 or MAC Address)".

The Response-Forwarding Option is of class E for OSCORE [RFC8613][I-D.ietf-core-oscore-groupcomm].

4. Requirements and Objectives

This specification assumes that the following requirements are fulfilled.

- REQ1. The CoAP proxy is explicitly configured (white-list) to allow proxied CoAP group requests from specific client(s).

- REQ2. The CoAP proxy MUST identify a client sending a CoAP group request, in order to verify whether that the client is white-listed to do so. This can rely for example on using a (D)TLS channel [RFC6347][I-D.ietf-tls-dtls13] between the client and the proxy, where the client has also been authenticated during the secure channel establishment.

- REQ3. If secure, end-to-end communication is required between the client and the servers in the CoAP group, exchanged messages MUST be protected by using Group OSCORE [I-D.ietf-core-oscore-groupcomm], as discussed in Section 5.2 of [I-D.dijk-core-groupcomm-bis]. This requires the client and the servers to have previously joined the correct OSCORE group, for instance by using the approach described in [I-D.ietf-ace-key-groupcomm-oscore]. The correct OSCORE group to join can be pre-configured or alternatively discovered, for instance by using the approach described in [I-D.tiloca-core-oscore-discovery].

This specification defines how to achieve the following objectives.

- OBJ1. The CoAP proxy gets an indication from the client that it is in fact interested to and capable to receive multiple responses to its unicast request containing a CoAP group URI.

- OBJ2. The CoAP proxy learns how long it should wait for responses to a proxied request, before starting to ignore following responses (except for notifications, if CoAP Observe is used [RFC7641]).

- OBJ3. The CoAP proxy returns individual unicast responses to the client, each of which matches the original unicast request to the proxy.

- OBJ4. The CoAP client is able to distinguish the different responses to the original unicast request, as well as their corresponding originator servers.

- OBJ5. The CoAP client is enabled to optionally contact one or more of the responding servers in the future, either directly or via a CoAP proxy.

5. Protocol Description

5.1. Request Sending

In order to send a request addressed to a group of servers via the proxy, the client proceeds as follows.

1. The client prepares a request addressed to the proxy. The request specifies the group URI as a string in the Proxi-URI option, or by using the Proxy-Scheme option with the group URI constructed from the URI-* options (see Section 2.3.3 of [I-D.dijk-core-groupcomm-bis]).

2. The client MUST retain the Token value used for this original unicast request beyond the reception of a first response matching it. To this end, the client follows the same rules for Token retention defined for multicast requests in Section 2.3.1 of [I-D.dijk-core-groupcomm-bis]. In particular, it picks an amount of time T before freeing up the Token value, such that:

   * T is smaller than the amount of time Tr it may pick for potentially reusing the Token value.

   * T includes the expected worst-case time taken by the request and response processing on the forward-proxy plus the servers in the addressed CoAP group.

   * T includes the expected worst-case round-trip delay between client and proxy, and between proxy and servers.

3. The client includes the Multicast-Signaling Option defined in Section 2, in the unicast request sent to the proxy. The option value specifies an amount of time T' < T. The difference (T - T') should include the expected worst-case round-trip time between the client and the forward-proxy.

4. The rest of the request processing occurs as defined in [I-D.dijk-core-groupcomm-bis], and in [I-D.ietf-core-oscore-groupcomm] when secure group communication is used.

5. The client sends the request to the proxy as a unicast CoAP message.

5.2. Request Processing at the Proxy

Upon receiving the request from the client, the proxy proceeds as follows.

1. The proxy identifies the client and verifies that it is in fact white-listed for proxy requests to CoAP group URIs.

2. The proxy verifies the presence of the Multicast-Signaling Option, as a confirmation that the client is fine to receive multiple responses matching the same original request.

3. The proxy forwards the client's request to the group of servers. In particular, the proxy sends it as a CoAP group request over IP multicast, addressed to the group URI specified by the client.

4. The proxy sets a timeout with the value T' retrieved from the Multicast-Signaling Option of the original unicast request. The proxy will ignore responses to the forwarded group request coming from servers, if received after the timeout expiration, with the exception of Observe notifications (see Section 5.4).

5.3. Request and Response Processing at the Server

Upon receiving the group request from the proxy, a server proceeds as follows.

1. Thanks to the Multicast-Signaling Option, the server understands that the original request originator is in fact a client behind a proxy.

2. The rest of the request processing occurs as defined in [I-D.dijk-core-groupcomm-bis], and in [I-D.ietf-core-oscore-groupcomm] when secure group communication is used.

3. When preparing its response to the proxy, to be forwarded back to the client, the server includes the Response-Forwarding Option defined in Section 3. The server specifies as option value its own addressing information, i.e. its unicast IP address, encoded as defined in Section 3. The server MUST include its IPv6 address if the multicast request was destined to an IPv6 multicast address and MUST include its IPv4 address if the multicast request was destined to an IPv4 address.

4. When using Observe [RFC7641], the server includes the Response-Forwarding Option also in every notification, including non-2.xx notifications resulting in removing the proxy from the list of observers.

5. The rest of the response processing occurs as defined in [I-D.dijk-core-groupcomm-bis], and in [I-D.ietf-core-oscore-groupcomm] when secure group communication is used.

5.4. Response Processing at the Proxy

Upon receiving a response matching the group request before the amount of time T' has elapsed, the proxy forwards the response back to the client.

Upon timeout expiration, i.e. T' seconds after having sent the group request over IP multicast, the proxy frees up its local Token value associated to that request. Thus, following late responses to the

same group request will be discarded and not forwarded back to the client.

When using CoAP Observe [RFC7641], the Token value is freed up only if, after the timeout expiration, no 2.xx (Success) responses matching the group request and also including an Observe option have been received. Then, as long as observations are active with servers in the group for the target resource of the group request, notifications from those servers are forwarded back to the client.

5.5. Response Processing at the Client

Upon receiving from the proxy a response that matches the original unicast request, i.e. before the amount of time T has elapsed, the client is able to identify the originator server, whose addressing information is specified as value of the Response-Forwarding Option.

In particular, the client is able to distinguish different responses as originated by different servers. Optionally the client may contact one or more of those servers individually, directly (bypassing the proxy) or indirectly (via a proxied CoAP unicast request). Note that the client already knows the destination port number to use for sending unicast requests to the server, i.e. the same port number specified in the group URI of the original unicast CoAP group request sent to the proxy (see Section 5.1).

The rest of the response processing occurs as defined in [I-D.dijk-core-groupcomm-bis], and in [I-D.ietf-core-oscore-groupcomm] when secure group communication is used.

Upon the timeout expiration, i.e. T seconds after having sent the original unicast request to the proxy, the client frees up its local Token value associated to that request. Note that, upon this timeout expiration, the Token value is not eligible for possible reuse yet (see Section 5.1). Thus, until the actual amount of time enabling Token reusage expires, following late responses to the same request forwarded by the proxy will be discarded, as not matching any active request Token from the client.

When using CoAP Observe [RFC7641], the Token value is freed up only if, after the timeout expiration, no 2.xx (Success) responses matching the original unicast and also including an Observe option have been received. If at least one such response has been received, then for as long as the observation for the target resource of the original unicast request is active, the client receives those notifications as forwarded by the proxy.

6. Security Considerations

The security considerations from [RFC7252][I-D.dijk-core-groupcomm-bis][RFC8613][I-D.ietf-core-oscore-groupcomm] hold for this document.

The Multicast-Signaling Option is of class I for OSCORE [RFC8613][I-D.ietf-core-oscore-groupcomm]. While this allows the proxy to access the option value and retrieve the timeout value T', the proxy is not able to remove the option altogether without this being noted by the servers. This ensures that the servers include their addressing information as value of the Response-Forwarding Option.

Besides, this prevents further possible intermediaries as well as on-path active adversaries to remove the option or alter its content. However, intermediares as well as on-path passive adversaries are able to access the option content, and thus learn for how long clients are willing to receive responses from the servers in the group via the proxy.

If no secure group communication is enforced end-to-end between the client and the servers (see Section 5.1 of [I-D.dijk-core-groupcomm-bis]), the proxy or any other on-path active intermediary is able to undetectably remove the Multicast-Signaling Option, i.e. to not include it in the group request sent to the servers in the group over multicast. As a consequence, the servers will not include the Response-Forwarding Option in their response, thus preventing the clients to distinguish the different responses and their corresponding originator server. The same result is achievable by removing the Response-Forwarding Option in the individual response of specific servers.

7. IANA Considerations

This document has the following actions for IANA.

7.1. CoAP Option Numbers Registry

IANA is asked to enter the following option numbers to the "CoAP Option Numbers" registry defined in [RFC7252] within the "CoRE Parameters" registry.

Number	Name	Reference
TBD1	Multicast-Signaling	[[this document]]
TBD2	Response-Forwarding	[[this document]]

8. References

8.1. Normative References

[I-D.dijk-core-groupcomm-bis]
Dijk, E., Wang, C., and M. Tiloca, "Group Communication for the Constrained Application Protocol (CoAP)", draft-dijk-core-groupcomm-bis-03 (work in progress), March 2020.

[I-D.ietf-core-oscore-groupcomm]
Tiloca, M., Selander, G., Palombini, F., and J. Park, "Group OSCORE - Secure Group Communication for CoAP", draft-ietf-core-oscore-groupcomm-07 (work in progress), March 2020.

[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>.

[RFC7049] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", RFC 7049, DOI 10.17487/RFC7049, October 2013, <https://www.rfc-editor.org/info/rfc7049>.

[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>.

[RFC7641] Hartke, K., "Observing Resources in the Constrained Application Protocol (CoAP)", RFC 7641, DOI 10.17487/RFC7641, September 2015, <https://www.rfc-editor.org/info/rfc7641>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019, <https://www.rfc-editor.org/info/rfc8613>.

8.2. Informative References

[I-D.bormann-coap-misc]
Bormann, C. and K. Hartke, "Miscellaneous additions to CoAP", draft-bormann-coap-misc-27 (work in progress), November 2014.

[I-D.ietf-ace-key-groupcomm-oscore]
Tiloca, M., Park, J., and F. Palombini, "Key Management for OSCORE Groups in ACE", draft-ietf-ace-key-groupcomm-oscore-05 (work in progress), March 2020.

[I-D.ietf-tls-dtls13]
Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", draft-ietf-tls-dtls13-37 (work in progress), March 2020.

[I-D.tiloca-core-oscore-discovery]
Tiloca, M., Amsuess, C., and P. Stok, "Discovery of OSCORE Groups with the CoRE Resource Directory", draft-tiloca-core-oscore-discovery-05 (work in progress), March 2020.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <https://www.rfc-editor.org/info/rfc6347>.

Acknowledgments

The work on this document has been partly supported by VINNOVA and the Celtic-Next project CRITISEC.

Authors' Addresses

Marco Tiloca
RISE AB
Isafjordsgatan 22
Kista SE-16440 Stockholm
Sweden

Email: marco.tiloca@ri.se

Esko Dijk
IoTconsultancy.nl
________________\
Utrecht
The Netherlands

Email: esko.dijk@iotconsultancy.nl