Group OSCORE – Secure Group Communication for CoAP

draft-ietf-core-oscore-groupcomm-16

Marco Tiloca, RISE
Göran Selander, Ericsson
Francesca Palombini, Ericsson
John P. Mattsson, Ericsson
Jiye Park, Universitaet Duisburg-Essen

IETF 115 meeting – London – November 7th, 2022
Updates since IETF 114

› Submitted version -16 before the cut-off
  1. Secure handling of multiple, non-notification responses from the same server
  2. One-to-many requests SHOULD be protected in group mode (was MUST)
  3. Improved presentation of security properties; style closer to RFC 8613
  4. Improved presentation of what the pairwise mode shares with the group mode

› All changes captured in PR #98 [1]
  – Early discussion with Christian Amsüss as document Shepherd
    › No need to wait for the Shepherd review and Write-up to make these updates
    › Some points also deserve to be documented from a more general point of view, see [2]
  – The PR received a very careful review from Rikard Höglund – Thanks!

› Post cut-off open point
  – Define and register a new link target attribute “gosc”

Protection of one-to-many requests

› OLD:
The group mode **MUST** be used to protect group requests intended for multiple recipients or for the whole group.

› NEW:
The group mode **SHOULD** be used to protect group requests intended for multiple recipients or for the whole group.

*For an example where this is not fulfilled, see [I-D.amsuess-core-cachable-oscore].*

› As to the cited case in point [3]
  – A deterministic request for cacheable OSCORE might be sent to multiple servers at once
  – Regardless, a deterministic request is protected with Group OSCORE but not in group mode

Handling of multiple responses

From Section 3.1.6 of draft-ietf-core-groupcomm-bis [4]

Since a client sending a group request with a Token T will accept multiple responses with the same Token T, it is possible in particular that the same server sends multiple responses with the same Token T back to the client. ...

… When this happens, the client normally processes at the CoAP layer each of those responses to the same request coming from the same server. If the processing of a response is successful, the client delivers this response to the application as usual.

Then, the application is in a better position to decide what to do, depending on the available context information.

This approach was first proposed at IETF 109 [5]

– The text above was added to -groupcomm-bis-03, before IETF 110 where it was confirmed [6]

[6] https://datatracker.ietf.org/meeting/110/session/core#session_28664
Handling of multiple responses

› Processing of responses in Group OSCORE
   – Single non-notification response from the same server to an (Observe) group request
     › All OK already
   – Multiple notification responses from the same server to an Observe group request
     › All OK already
   – Multiple non-notification responses from the same server to an (Observe) group request
     › This was underspecified before the latest version -16
     › Note: this is irrespective of the request being an Observe request or not

› What was missing on the server side?
   – The Partial IV was not mandatory in the non-notification responses → Reuse of AEAD nonce!

› What was missing on the client side?
   – Replay checks and message ordering were not performed on non-notification responses
Handling of multiple responses

› **How did we address this?**
  – The same rationale used for Observe notifications is *separately* used for non-notification responses

› **New concept: “Non-notification group exchange”**
  – Like for an Observation, it is an “environment” on the client side associated with one group request
  – Used to track non-notification responses, *regardless the request being an Observe request or not*

› **Non-notification responses on the server side**
  – The Partial IV MUST be included in a response, with the possible exception of the first one

› **Non-notification responses on the client side**
  – Use the Partial IV of responses as a “Response Number” (analogous to “Notification Number”)
  – Admit only one such response without Partial IV from the same server, and treat it as the oldest one
  – Use the Response Number to perform replay checks and ordering of such responses
Handling of multiple responses

› Side points were also handled in the same way already used for Observe

› Non-notification group exchanges across a group rekeying
  – The endpoints store the ‘kid context’ of the original group request
  – This is always used when building the external_aad of responses, even after the group rekeying

› Non-notification group exchanges across a change of Client Sender ID
  – The endpoints store the ‘kid’ of the original group request
  – This is always used when building the external_aad of responses

› Editorially-revised presentation of security properties (see especially Section 6)
  – This takes into account also the new handling of non-notification responses
Link target attribute “gosc”?

› **RFC 8613 defines the link target attribute “osc”** [7]

The "osc" attribute is a hint indicating that the destination of that link is only accessible using OSCORE, and unprotected access to it is not supported.

› **Proposal: define and register the link target attribute “gosc”**

The "gosc" attribute is a hint indicating that the destination of that link is only accessible using OSCORE and/or Group OSCORE, and unprotected access to it is not supported.

› **Rules of use**

  – If a link specifies “gosc”, it MUST also specify “osc”
  – If an endpoint consuming the link does not understand “gosc”, it ignores “gosc” anyway
  – If an endpoint consuming the link understands “gosc”, then it ignores “osc” as overshadowed

Summary and next steps

› Changes in version #16, based on PR #98 [1]
  – Secure handling of multiple, non-notification responses from the same server
  – One-to-many requests SHOULD be protected in group mode (was MUST)
  – Improved presentation of security properties; style closer to RFC 8613
  – Improved presentation of what the pairwise mode shares with the group mode

› Next steps
  – Define and register the link target attribute “gosc” → Submit new revision -17
  – Wait for the Shepherd review and Write-up from Christian Amsüss

Thank you!

Comments/questions?

https://github.com/core-wg/oscore-groupcomm