Admin Interface for the OSCORE Group Manager

draft-ietf-ace-oscore-gm-admin-04

Marco Tiloca, RISE
Rikard Höglund, RISE
Peter van der Stok
Francesca Palombini, Ericsson

IETF 112, ACE WG, November 9th, 2021
Recap

› **RESTful admin interface at the OSCORE Group Manager**
  – Create, (re-)configure and delete OSCORE groups
  – Support for both: i) Link Format and CBOR ; ii) CoRAL

› **Two new types of resources at the Group Manager**
  – A *single* group-collection resource, at /manage
  – One *group-configuration* resource per group, at /manage/GROUPNAME

› **Using ACE for authentication and authorization**
  – The Administrator is the Client
  – The Group Manager is the Resource Server
  – For secure communication, use transport profiles of ACE
Overview

**Group-collection resource**
- Create a new OSCORE group (POST)
  - A group-configuration resource is created
  - A group-membership for joining nodes is also created, see `ace-key-groupcomm-oscore`
- Retrieve the list of OSCORE groups
  - All groups (GET)
  - Group selected by filters (FETCH)

**Group-configuration resource**
- Retrieve the group configuration (GET)
- Retrieve part of the group configuration (FETCH)
- Overwrite the group configuration (PUT)
- Update the group configuration (PATCH)
- Delete the group (DELETE)
Main updates from -02, -03 and -04

› Improved error handling, using also the new error types
  – Following the format of error messages in `draft-ietf-ace-key-groupcomm`
  – Additional error situations are handled and replied to, e.g. deleting a still active group

› Admit multiple Administrators
  – All Administrators can access the group-configuration resource
  – A set of Administrators can access a group-configuration resource
    › Possibly, only certain operations to certain Administrators
    › Possibly, act on groups created by another Administrator
  – “Classes” of Administrators, to be enforced through ‘scope’
    › Section 2.1.1 as placeholder, with a sketched technical direction
Main updates from -02, -03 and -04

Added PATCH/iPATCH handler for group-configuration resources

› Selective updates of an existing group configuration
  – Specify only parameters to update
  – Other parameters keep their current value (they don’t default like when PUT is used)

› Plain “replacement” update for most parameters, as new pair (“label”, value)

› Special handling for the ‘app_groups’ parameter – List of names of application groups
  – Option 1: whole new list as hard replacement
  – Option 2: set of names to remove and set of new names to add

} Not both in the same PATCH request!

› PATCH can’t be used to create a new group!
Main updates from -02, -03 and -04

- Improved handling of default values for configuration/status parameters
  - If the request to the Group Manager does not include a parameter ...
  - ... and the Group Manager uses a different default value than what recommended ...
  - ... then the Group Manager must include the chosen value in the response

- New optional configuration parameters for OSCORE deterministic requests
  - Main approach defined in draft-amsuess-core-cachable-oscore
  - ‘det_req’ : simple value true/false  // Use of deterministic requests in the group
  - ‘det_hash_alg’ : tstr/int      // Hash algorithm to use with deterministic requests

- Alignment with recent updates in other documents, especially
  - draft-ietf-ace-key-groupcomm, draft-ietf-ace-key-groupcomm-oscore
  - draft-ietf-core-oscore-groupcomm
Summary and next steps

Latest updates

- Improved error handling
- Admitting multiple Administrators (to be practically enforced with ‘scope’)
- PATCH/iPATCH, to selectively update a group configuration
- New/update parameters, with handling of default values
- Revised examples and side effects due to parameter changes

Next steps

- Define the format of ‘scope’, using AIF and patterns for group names
  - Allow some actions to Administrators that did not create the group [1]
  - Direction sketched in Section 2.1.1 – Discuss at an ACE interim meeting?
- Keep aligned with other related documents as they get updated

[1] https://mailarchive.ietf.org/arch/msg/ace/gLr5NgAURoi5P9f6RcgHkL2jFr8/
Thank you!

Comments/questions?

https://github.com/ace-wg/ace-oscore-gm-admin
Backup
Group Configuration Parameters

Configuration properties
- hkdf
- pub_key_enc
- group_mode
- sign_enc_alg
- sign_alg
- sign_params
- pairwise_mode
- alg
- ecdh_alg
- ecdh_params
- det_req
- det_hash_alg

Status properties
- rt = “core.osc.gconf”
- active
- group_name // Plain immutable identifier
- group_title // Descriptive string
- ace_groupcomm_profile
- exp
- app_groups // Names of application groups
- joining_uri
- ? group_policies
- ? max_stale_sets
- ? as_uri // Link to the AS

- When using PATCH, easy “replacement” update for most parameters
  - Specify the pair (“label”, new_value), like when creating the group
- ‘app_groups’ is a list of names and requires special handling
Two ways to update ‘app_groups’
– List of associated applications groups

Overwrite – New array of names as hard replacement
– app_groups : [“room1”, “room8”]  Custom CBOR
– app_group “room1”
  app_group “room8”  CoRAL

Addition/deletion – [ [*name_to_remove], [*name_to_add] ]
– app_groups_diff : [ [“room1”], [“room5”] ]  Custom CBOR
– app_group_del “room1”
  app_group_add “room8”  CoRAL

Overwrite and addition/deletion not together in the same PATCH payload
Configuration update with PATCH

- **4.00 (Bad request)**
  - Any malformed or invalid payload
  - iPATCH is used as request method, but:
    - ‘app_groups_diff’ is included (Custom CBOR)
    - ‘app_group_del’ and/or ‘app_group_add’ are included (CoRAL)

- **4.09 (Conflict)**
  - New parameter values would yield an inconsistent group configuration

- **4.22 (Unprocessable entity)** might be returned just as per RFC 8132
  - The server is unable to or is incapable of processing the request
Sketched format of ‘scope’

Section 2.1.1 as placeholder, with a list of design considerations

New AIF Data Model – Similar to draft-ace-key-groupcomm-oscore

AIF-Generic<Toid, Tperm> = [ *[Toid, Tperm] ]

- Toid: Text string, specifying a name pattern for security groups
- Tperm: Unsigned integer, indicating permissions as flag bits
  - Retrieving the list of security groups is always possible, for any Tperm

Possible granted permissions – Over group names matching the pattern!

- Create the groups, and later do anything on those
- Read the configuration of groups created by others
- Overwrite/update the configuration of groups created by others
- Delete groups created by others

Useful to enable additional Administrators after a group creation