

Workgroup: ACE Working Group

Internet-Draft:

draft-ietf-ace-oscore-gm-admin-05

Published: 7 March 2022

Intended Status: Standards Track

Expires: 8 September 2022

Authors: M. Tiloca R. Höglund P. van der Stok
 RISE AB RISE AB Consultant
 F. Palombini
 Ericsson AB

Admin Interface for the OSCORE Group Manager

Abstract

Group communication for CoAP can be secured using Group Object Security for Constrained RESTful Environments (Group OSCORE). A Group Manager is responsible to handle the joining of new group members, as well as to manage and distribute the group keying material. This document defines a RESTful admin interface at the Group Manager, that allows an Administrator entity to create and delete OSCORE groups, as well as to retrieve and update their configuration. The ACE framework for Authentication and Authorization is used to enforce authentication and authorization of the Administrator at the Group Manager. Protocol-specific transport profiles of ACE are used to achieve communication security, proof-of-possession and server authentication.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Authentication and Authorization for Constrained Environments Working Group mailing list (ace@ietf.org), which is archived at <https://mailarchive.ietf.org/arch/browse/ace/>.

Source for this draft and an issue tracker can be found at <https://github.com/ace-wg/ace-oscore-gm-admin>.

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 8 September 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. [Introduction](#)
 - 1.1. [Terminology](#)
2. [Group Administration](#)
 - 2.1. [Managing OSCORE Groups](#)
 - 2.2. [Collection Representation](#)
 - 2.3. [Discovery](#)
3. [Format of Scope](#)
4. [Getting Access to the Group Manager](#)
5. [Group Configurations](#)
 - 5.1. [Group Configuration Representation](#)
 - 5.1.1. [Configuration Properties](#)
 - 5.1.2. [Status Properties](#)
 - 5.2. [Default Values](#)
 - 5.2.1. [Configuration Parameters](#)
 - 5.2.2. [Status Parameters](#)
6. [Interactions with the Group Manager](#)
 - 6.1. [Retrieve the Full List of Group Configurations](#)
 - 6.2. [Retrieve a List of Group Configurations by Filters](#)
 - 6.3. [Create a New Group Configuration](#)
 - 6.4. [Retrieve a Group Configuration](#)
 - 6.5. [Retrieve Part of a Group Configuration by Filters](#)
 - 6.6. [Overwrite a Group Configuration](#)
 - 6.6.1. [Effects on Joining Nodes](#)
 - 6.6.2. [Effects on the Group Members](#)

- [6.7. Selective Update of a Group Configuration](#)
 - [6.7.1. Effects on Joining Nodes](#)
 - [6.7.2. Effects on the Group Members](#)
- [6.8. Delete a Group Configuration](#)
 - [6.8.1. Effects on the Group Members](#)
- [7. ACE Groupcomm Error Identifiers](#)
- [8. Security Considerations](#)
- [9. IANA Considerations](#)
 - [9.1. ACE Groupcomm Parameters](#)
 - [9.2. ACE Groupcomm Errors](#)
 - [9.3. Resource Types](#)
 - [9.4. Group OSCORE Admin Permissions](#)
 - [9.5. AIF](#)
 - [9.6. CoAP Content-Format](#)
 - [9.7. ACE Scope Semantics](#)
 - [9.8. Expert Review Instructions](#)
- [10. References](#)
 - [10.1. Normative References](#)
 - [10.2. Informative References](#)
- [Appendix A. Document Updates](#)
 - [A.1. Version -04 to -05](#)
 - [A.2. Version -03 to -04](#)
 - [A.3. Version -02 to -03](#)
 - [A.4. Version -01 to -02](#)
 - [A.5. Version -00 to -01](#)
- [Acknowledgments](#)
- [Authors' Addresses](#)

1. Introduction

The Constrained Application Protocol (CoAP) [[RFC7252](#)] can be used in group communication environments where messages are also exchanged over IP multicast [[I-D.ietf-core-groupcomm-bis](#)]. Applications relying on CoAP can achieve end-to-end security at the application layer by using Object Security for Constrained RESTful Environments (OSCORE) [[RFC8613](#)], and especially Group OSCORE [[I-D.ietf-core-oscore-groupcomm](#)] in group communication scenarios.

When group communication for CoAP is protected with Group OSCORE, nodes are required to explicitly join the correct OSCORE group. To this end, a joining node interacts with a Group Manager (GM) entity responsible for that group, and retrieves the required keying material to securely communicate with other group members using Group OSCORE.

The method in [[I-D.ietf-ace-key-groupcomm-oscore](#)] specifies how nodes can join an OSCORE group through the respective Group Manager. Such a method builds on the ACE framework for Authentication and Authorization [[I-D.ietf-ace-oauth-authz](#)], so ensuring a secure

joining process as well as authentication and authorization of joining nodes (clients) at the Group Manager (resource server).

In some deployments, the application running on the Group Manager may know when a new OSCORE group has to be created, as well as how it should be configured and later on updated or deleted, e.g., based on the current application state or on pre-installed policies. In this case, the Group Manager application can create and configure OSCORE groups when needed, by using a local application interface. However, this requires the Group Manager to be application-specific, which in turn leads to error prone deployments and is poorly flexible.

In other deployments, a separate Administrator entity, such as a Commissioning Tool, is directly responsible for creating and configuring the OSCORE groups at a Group Manager, as well as for maintaining them during their whole lifetime until their deletion. This allows the Group Manager to be agnostic of the specific applications using secure group communication.

This document specifies a RESTful admin interface at the Group Manager, intended for an Administrator as a separate entity external to the Group Manager and its application. The interface allows the Administrator to create and delete OSCORE groups, as well as to configure and update their configuration.

Interaction examples are provided, in Link Format [[RFC6690](#)] and CBOR [[RFC8949](#)], as well as in CoRAL [[I-D.ietf-core-coral](#)]. While all the CoRAL examples show the CoRAL textual serialization format, its binary serialization format is used on the wire.

[NOTE:

The reported CoRAL examples are based on the textual representation used until version -03 of [[I-D.ietf-core-coral](#)]. These will be revised to use the CBOR diagnostic notation instead.

]

The ACE framework is used to ensure authentication and authorization of the Administrator (client) at the Group Manager (resource server). In order to achieve communication security, proof-of-possession and server authentication, the Administrator and the Group Manager leverage protocol-specific transport profiles of ACE, such as [[I-D.ietf-ace-oscore-profile](#)][[I-D.ietf-ace-dtls-authorize](#)]. These include also possible forthcoming transport profiles that comply with the requirements in Appendix C of [[I-D.ietf-ace-oauth-authz](#)].

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 the terms and concepts from the following specifications:

*CBOR [[RFC8949](#)] and COSE [[I-D.ietf-cose-rfc8152bis-struct](#)][[I-D.ietf-cose-rfc8152bis-algs](#)].

*The CoAP protocol [[RFC7252](#)], also in group communication scenarios [[I-D.ietf-core-groupcomm-bis](#)]. These include the concepts of:

- "application group", as a set of CoAP nodes that share a common set of resources; and of
- "security group", as a set of CoAP nodes that share the same security material, and use it to protect and verify exchanged messages.

*The OSCORE [[RFC8613](#)] and Group OSCORE [[I-D.ietf-core-oscore-groupcomm](#)] security protocols. These especially include the concepts of:

- Group Manager, as the entity responsible for a set of OSCORE groups where communications among members are secured using Group OSCORE. An OSCORE group is used as security group for one or many application groups.
- Authentication credential, as the set of information associated with an entity, including that entity's public key and parameters associated with the public key. Examples of authentication credentials are CBOR Web Tokens (CWTs) and CWT Claims Sets (CCSs) [[RFC8392](#)], X.509 certificates [[RFC7925](#)] and C509 certificates [[I-D.ietf-cose-cbor-encoded-cert](#)].

*The ACE framework for authentication and authorization [[I-D.ietf-ace-oauth-authz](#)]. The terminology for entities in the considered architecture is defined in OAuth 2.0 [[RFC6749](#)]. In particular, this includes Client (C), Resource Server (RS), and Authorization Server (AS).

*The management of keying material for groups in ACE [[I-D.ietf-ace-key-groupcomm](#)] and specifically for OSCORE groups [[I-D.ietf-ace-key-groupcomm-oscore](#)]. These include the concept of group-

membership resource hosted by the Group Manager, that new members access to join the OSCORE group, while current members can access to retrieve updated keying material.

Note that, unless otherwise indicated, the term "endpoint" is used here following its OAuth definition, aimed at denoting resources such as /token and /introspect at the AS, and /authz-info at the RS. This document does not use the CoAP definition of "endpoint", which is "An entity participating in the CoAP protocol".

This document also refers to the following terminology.

*Administrator: entity responsible to create, configure and delete OSCORE groups at a Group Manager.

*Group name: stable and invariant name of an OSCORE group. The group name MUST be unique under the same Group Manager, and MUST include only characters that are valid for a URI path segment.

*Group-collection resource: a single-instance resource hosted by the Group Manager. An Administrator accesses a group-collection resource to retrieve the list of existing OSCORE groups, or to create a new OSCORE group, under that Group Manager.

As an example, this document uses /manage as the url-path of the group-collection resource; implementations are not required to use this name, and can define their own instead.

*Group-configuration resource: a resource hosted by the Group Manager, associated with an OSCORE group under that Group Manager. A group-configuration resource is identifiable with the invariant group name of the respective OSCORE group. An Administrator accesses a group-configuration resource to retrieve or change the configuration of the respective OSCORE group, or to delete that group.

The url-path to a group-configuration resource has GROUPNAME as last segment, with GROUPNAME the invariant group name assigned upon its creation. Building on the considered url-path of the group-collection resource, this document uses /manage/GROUPNAME as the url-path of a group-configuration resource; implementations are not required to use this name, and can define their own instead.

*Admin endpoint: an endpoint at the Group Manager associated with the group-collection resource or to a group-configuration resource hosted by that Group Manager.

2. Group Administration

With reference to the ACE framework and the terminology defined in OAuth 2.0 [[RFC6749](#)]:

- *The Group Manager acts as Resource Server (RS). It provides one single group-collection resource, and one group-configuration resource per existing OSCORE group. Each of those is exported by a distinct admin endpoint.
- *The Administrator acts as Client (C), and requests to access the group-collection resource and group-configuration resources, by accessing the respective admin endpoint at the Group Manager.
- *The Authorization Server (AS) authorizes the Administrator to access the group-collection resource and group-configuration resources at a Group Manager. Multiple Group Managers can be associated with the same AS.

The authorized access for an Administrator can be limited to performing only a subset of operations, according to what is allowed by the authorization information in the Access Token issued to that Administrator (see [Section 3](#) and [Section 4](#)). The AS can authorize multiple Administrators to access the group-collection resource and the (same) group-configuration resources at the Group Manager.

The AS MAY release Access Tokens to the Administrator for other purposes than accessing admin endpoints of registered Group Managers.

2.1. Managing OSCORE Groups

[Figure 1](#) shows the resources of a Group Manager available to an Administrator.

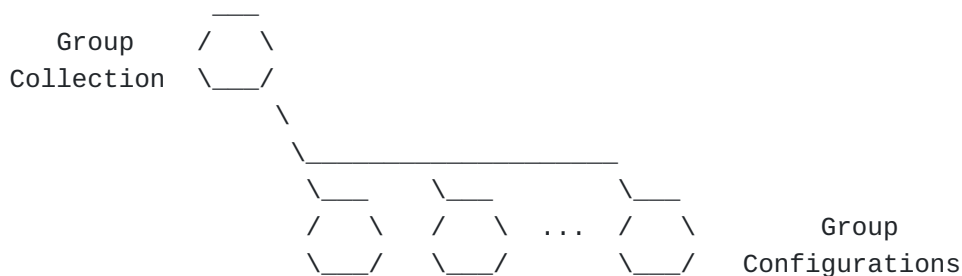


Figure 1: Resources of a Group Manager

The Group Manager exports a single group-collection resource, with resource type "core.osc.gcoll" defined in [Section 9.3](#) of this

document. The interface for the group-collection resource defined in [Section 6](#) allows the Administrator to:

- *Retrieve the list of existing OSCORE groups.
- *Retrieve the list of existing OSCORE groups matching with specified filter criteria.
- *Create a new OSCORE group, specifying its invariant group name and, optionally, its configuration.

The Group Manager exports one group-configuration resource for each of its OSCORE groups. Each group-configuration resource has resource type "core.osc.gconf" defined in [Section 9.3](#) of this document, and is identified by the group name specified upon creating the OSCORE group. The interface for a group-configuration resource defined in [Section 6](#) allows the Administrator to:

- *Retrieve the complete current configuration of the OSCORE group.
- *Retrieve part of the current configuration of the OSCORE group, by applying filter criteria.
- *Overwrite the current configuration of the OSCORE group.
- *Selectively update only part of the current configuration of the OSCORE group.
- *Delete the OSCORE group.

2.2. Collection Representation

A list of group configurations is represented as a document containing the corresponding group-configuration resources in the list. Each group-configuration is represented as a link, where the link target is the URI of the group-configuration resource.

The list can be represented as a Link Format document [[RFC6690](#)] or a CoRAL document [[I-D.ietf-core-coral](#)].

In the former case, the link to each group-configuration resource specifies the link target attribute 'rt' (Resource Type), with value "core.osc.gconf" defined in [Section 9.3](#) of this document.

In the latter case, the CoRAL document specifies the group-configuration resources in the list as top-level elements. In particular, the link to each group-configuration resource has `http://coreapps.org/core.osc.gcoll#item` as relation type.

2.3. Discovery

The Administrator can discover the group-collection resource from a Resource Directory, for instance [[I-D.ietf-core-resource-directory](#)] and [[I-D.hartke-t2trg-coral-reef](#)], or from .well-known/core, by using the resource type "core.osc.gcoll" defined in [Section 9.3](#) of this document.

The Administrator can discover group-configuration resources for the group-collection resource as specified in [Section 6.1](#) and [Section 6.2](#).

3. Format of Scope

This section defines the exact format and encoding of scope to use, in order to express authorization information for the Administrator (see [Section 4](#)).

To this end, this document uses the Authorization Information Format (AIF) [[I-D.ietf-ace-aif](#)], and defines the following AIF specific data model AIF-OSCORE-GROUPCOMM-ADMIN.

With reference to the generic AIF model

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

the value of the CBOR byte string used as scope encodes the CBOR array [* [Toid, Tperm]], where each [Toid, Tperm] element corresponds to one scope entry.

Then, for each scope entry, the following applies.

- *The object identifier ("Toid") is specialized as a CBOR text string, specifying a wildcard pattern P for the scope entry. The pattern P is intended as a template for group names.

- *The permission set ("Tperm") is specialized as a CBOR unsigned integer with value Q. This specifies the permissions that the Administrator has to perform operations on the admin endpoints at the Group Manager, as pertaining to any OSCORE group whose name matches with the wildcard pattern P. The value Q is computed as follows.

- Each permission in the permission set is converted into the corresponding numeric identifier X from the "Value" column of the "Group OSCORE Admin Permissions" registry, for which this document defines the entries in [Figure 2](#).
- The set of N numbers is converted into the single value Q, by taking each numeric identifier X₁, X₂, ..., X_N to the power

of two, and then computing the inclusive OR of the binary representations of all the power values.

In general, a single permission can be associated with multiple different operations that are possible to be performed when interacting with the Group Manager. For example, the "List" permission allows the Administrator to retrieve a list of group configurations (see [Section 6.1](#)) or only a subset of that according to specified filter criteria (see [Section 6.2](#)), by issuing a GET or FETCH request to the group-collection resource, respectively.

Name	Value	Description
List	0	Retrieve list of group configurations
Create	1	Create new group configurations
Read	2	Retrieve group configurations
Write	3	Change group configurations
Delete	4	Delete group configurations

Figure 2: Numeric identifier of permissions on the admin endpoints at a Group Manager

The CDDL [[RFC8610](#)] definition of the AIF-OSCORE-GROUPCOMM-ADMIN data model and the format of scope using such a data model is as follows:

AIF-OSCORE-GROUPCOMM-ADMIN = AIF-Generic<pattern, permissions>

pattern = tstr ; wilcard pattern of group names

permission_set = uint . bits permissions

permissions = &(

List: 0,
Create: 1,
Read: 2,
Write: 3,
Delete: 4

)

scope_entry = AIF-OSCORE-GROUPCOMM-ADMIN

scope = << [+ scope_entry] >>

By relying on the scope format defined above and given an OSCORE group G1 created by a "main" Administrator, then a second "assistant" Administrator can be effectively authorized to perform some operations on G1, in spite of not being the group creator.

Furthermore, having the object identifier ("Toid") specialized as a wildcard pattern displays a number of advantages.

- *The encoded scope can be compact in size, while allowing the Administrator to operate on large pools of group names.

- *The Administrator and the AS do not need to know exact group names when requesting and issuing an Access Token, respectively (see [Section 4](#)). In turn, the Group Manager can effectively take the final decision about the name to assign to an OSCORE group, upon its creation (see [Section 6.3](#)).

- *The Administrator may have established a secure communication association with the Group Manager based on a first Access Token T1, and then created an OSCORE group G. Following the invalidation of T1 (e.g., due to expiration) and the establishment of a new secure communication association with the Group Manager based on a new Access Token T2, the Administrator can seamlessly perform authorized operations on the previously created group G.

When using the scope format defined in this section, the permission set ("Tperm") of each scope entry MUST include the "List" permission in order for the scope to be considered valid. That is, for each scope entry, the unsigned integer Q MUST be odd. Therefore, an Administrator is always allowed to retrieve a list of existing group configurations. The exact elements included in the returned list are determined by the Group Manager, based on the group name patterns specified in the scope entries of the Administrator's Access Token, as well as on possible filter criteria specified in the request from the Administrator.

[NOTE:

There is a potential follow-up building on this.

An ACE Client might want to interact with the same Group Manager to be both Administrator for some groups and member for some other groups.

In order to keep a single Access Token per Client, the scope would have to generally include some "admin" scope entries as per the AIF data model defined in this document, together with some "user" scope entries as per the AIF data model defined in [[I-D.ietf-ace-key-groupcomm-oscore](#)].

In the scope entries of the former type, the least significant bit of the Tperm integer and denoting the "List" admin permission is always set to 1 (see above). In the scope entries of the latter type, the least significant bit of the Tperm integer is reserved and always 0 (see [[I-D.ietf-ace-key-groupcomm-oscore](#)]).

Therefore, "admin" and "user" scope entries can unambiguously coexist in the same 'scope' claim and Authorization Request/Response parameter, and can be easily distinguished by checking the least significant bit of the Tperm integer.

In turn, this would require to accordingly revise the scope format and the ACE scope semantics integer defined in this document, in order to denote the certain presence of "admin" scope entries and the optional additional presence of "user" scope entries, within a same scope claim/parameter.

]

Future specifications that define new permissions on the admin endpoints at the Group Manager MUST register a corresponding numeric identifier in the "Group OSCORE Admin Permissions" registry defined in [Section 9.4](#) of this document.

4. Getting Access to the Group Manager

All communications between the involved entities rely on the CoAP protocol and MUST be secured.

In particular, communications between the Administrator and the Group Manager leverage protocol-specific transport profiles of ACE to achieve communication security, proof-of-possession and server authentication. To this end, the AS may explicitly signal the specific transport profile to use, consistently with requirements and assumptions defined in the ACE framework [[I-D.ietf-ace-oauth-authz](#)].

With reference to the AS, communications between the Administrator and the AS (/token endpoint) as well as between the Group Manager and the AS (/introspect endpoint) can be secured by different means, for instance using DTLS [[RFC6347](#)][[I-D.ietf-tls-dtls13](#)] or OSCORE [[RFC8613](#)]. Further details on how the AS secures communications (with the Administrator and the Group Manager) depend on the specifically used transport profile of ACE, and are out of the scope of this document.

The format and encoding of scope defined in [Section 3](#) of this document MUST be used, for both the 'scope' claim in the Access Token, as well as for the 'scope' parameter in the Authorization

Request and Authorization Response exchanged with the AS (see Sections [5.8.1](#) and [5.8.2](#) of [[I-D.ietf-ace-oauth-authz](#)]).

Furthermore, the AS MAY use the extended format of scope defined in [Section 7](#) of [[I-D.ietf-ace-key-groupcomm](#)] for the 'scope' claim of the Access Token. In such a case, the first element of the CBOR sequence [[RFC8742](#)] MUST be the CBOR integer with value SEM_ID_TBD, defined in [Section 9.7](#) of this document. This indicates that the second element of the CBOR sequence, as conveying the actual access control information, follows the scope semantics defined in [Section 3](#) of this document.

In order to get access to the Group Manager for managing OSCORE groups, an Administrator performs the following steps.

1. The Administrator requests an Access Token from the AS, in order to access the group-collection and group-configuration resources on the Group Manager. To this end, it sends to the AS an Authorization Request as defined in [Section 5.8.1](#) of [[I-D.ietf-ace-oauth-authz](#)]. The Administrator will start or continue using secure communications with the Group Manager, according to the response from the AS.
2. The AS processes the Authorization Request as defined in [Section 5.8.2](#) of [[I-D.ietf-ace-oauth-authz](#)], especially verifying that the Administrator is authorized to obtain the requested permissions, or possibly a subset of those.

With reference to the scope format specified in [Section 3](#), the AS builds the value of the 'scope' claim to include in the Access Token as follows.

1. The AS initializes three empty sets of scope entries, namely S1, S2 and S3.
2. For each scope entry E in the 'scope' parameter of the Authorization Request, the AS performs the following actions.

In its access policies related to administrative operations at the Group Manager for the Administrator, the AS determines every group name superpattern P, such that every group name matching with the wildcard pattern P of the scope entry E matches also with P*.

If no superpatterns are found, the AS proceeds with the next scope entry, if any. Otherwise, the AS computes Tperm as the union of the permission sets associated with the superpatterns found at the previous step. That is, Tperm* is the inclusive OR of the binary

representations of the Tperm values associated with the found superpatterns and encoding the corresponding permission sets as per [Section 3](#).

The AS adds to the set S1 a scope entry, such that its Toid is the same as in the scope entry E, while its Tperm is the AND of Tperm with the Tperm in the scope entry E.

3. For each scope entry E in the 'scope' parameter of the Authorization Request, the AS performs the following actions.

In its access policies related to administrative operations at the Group Manager for the Administrator, the AS determines every group name subpattern P, such that: i) the wildcard pattern P of the scope entry E is different from P*; and ii) every group name matching with P* also matches with P.

If no subpatterns are found, the AS proceeds with the next scope entry, if any. Otherwise, for each found subpattern P, the AS adds to the set S2 a scope entry, such that its Toid is the same as in the subpattern P*, while its Tperm is the AND of the Tperm from the subpattern P* with the Tperm in the scope entry E.

4. For each scope entry E in the 'scope' parameter of the Authorization Request, the AS performs the following actions.

For each group name pattern P in its access policies related to administrative operations at the Group Manager for the Administrator, the AS performs the following actions.

-The AS attempts to determine a crosspattern P** such that: i) in the previous step, P** was not identified as a superpattern or subpattern for the pattern P of the scope entry E; ii) every group name matching with P** also matches with both P and P*.

-If no crosspattern is built, the AS proceeds with the next pattern in its access policies related to administrative operations at the Group Manager for the Administrator, if any. Otherwise, the AS adds to the set S3 a scope entry, such that its Toid is the same as in the crosspattern P**, while its Tperm is the AND of the Tperm from the pattern P* and the Tperm in the scope entry E.

5. If the sets S1, S2 and S3 are all empty, the Authorization Request has not been successfully verified, and the AS returns an error response as per [Section 5.8.3](#) of [[I-D.ietf-ace-oauth-authz](#)]. Otherwise, the AS uses the scope entries in the sets S1, S2 and S3 as the scope entries for the 'scope' claim to include in the Access Token, as per the format defined in [Section 3](#).

The AS MUST include the 'scope' parameter in the Authorization Response defined in [Section 5.8.2](#) of [[I-D.ietf-ace-oauth-authz](#)], when the value included in the Access Token differs from the one specified by the Administrator in the Authorization Response. In such a case, the second element of each scope entry specifies a set of permissions that the Administrator actually has to perform operations at the Group Manager, encoded as specified in [Section 3](#).

3. The Administrator transfers authentication and authorization information to the Group Manager by posting the obtained Access Token, according to the used profile of ACE, such as [[I-D.ietf-ace-dtls-authorize](#)] and [[I-D.ietf-ace-oscore-profile](#)]. After that, the Administrator must have a secure communication association established with the Group Manager, before performing any administrative operation on that Group Manager. Possible ways to provide secure communication are DTLS [[RFC6347](#)][[I-D.ietf-tls-dtls13](#)] and OSCORE [[RFC8613](#)]. The Administrator and the Group Manager maintain the secure association, to support possible future communications.
4. Consistently with what is allowed by the authorization information in the Access Token, the Administrator performs administrative operations at the Group Manager, as described in [Section 6](#). These include retrieving a list of existing OSCORE groups, creating new OSCORE groups, retrieving and changing OSCORE group configurations, and removing OSCORE groups. Messages exchanged among the Administrator and the Group Manager are specified in [Section 6](#).

Upon receiving a request from the Administrator targeting the group-configuration resource or a group-collection resource, the Group Manager MUST check that it is storing a valid Access Token for that Administrator. If this is not the case, the Group Manager MUST reply with a 4.01 (Unauthorized) error response.

If the request targets the group-configuration resource associated to a group with name GROUPNAME, the Group Manager MUST check that it is storing a valid Access Token from that Administrator, such that the 'scope' claim specified in the

Access Token has the format defined in [Section 3](#) and includes a scope entry where:

- *The group name GROUPNAME matches with the wildcard pattern specified in the scope entry; and

- *The permission set specified in the scope entry allows the Administrator to perform the requested operation on the targeted group-configuration resource.

Further details are defined separately for each operation specified in [Section 6](#).

In case the Group Manager stores a valid Access Token but the verifications above fail, the Group Manager MUST reply with a 4.03 (Forbidden) error response. This response MAY be an AS Request Creation Hints, as defined in [Section 5.3](#) of [[I-D.ietf-ace-oauth-authz](#)], in which case the Content-Format MUST be set to application/ace+cbor.

If the request is not formatted correctly (e.g., required fields are not present or are not encoded as expected), the Group Manager MUST reply with a 4.00 (Bad Request) error response.

5. Group Configurations

A group configuration consists of a set of parameters.

5.1. Group Configuration Representation

The group configuration representation is a CBOR map which MUST include configuration properties and status properties.

5.1.1. Configuration Properties

The CBOR map MUST include the following configuration parameters, whose CBOR abbreviations are defined in [Section 9.1](#) of this document.

- *'hkdf', which specifies the HKDF Algorithm used in the OSCORE group, encoded as a CBOR text string or a CBOR integer. Possible values are the same ones admitted for the 'hkdf' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

- *'cred_fmt', which specifies the format of authentication credentials used in the OSCORE group, encoded as a CBOR integer. Possible values are the same ones admitted for the 'cred_fmt'

parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'group_mode', encoded as a CBOR simple value. Its value is "true" (0xf5) if the OSCORE group uses the group mode of Group OSCORE [[I-D.ietf-core-oscore-groupcomm](#)], or "false" (0xf4) otherwise.

*'sign_enc_alg', which is formatted as follows. If the configuration parameter 'group_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the Signature Encryption Algorithm used in the OSCORE group to encrypt messages protected with the group mode, encoded as a CBOR text string or a CBOR integer. Possible values are the same ones admitted for the 'sign_enc_alg' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'sign_alg', which is formatted as follows. If the configuration parameter 'group_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the Signature Algorithm used in the OSCORE group, encoded as a CBOR text string or a CBOR integer. Possible values are the same ones admitted for the 'sign_alg' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'sign_params', which is formatted as follows. If the configuration parameter 'group_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the additional parameters for the Signature Algorithm used in the OSCORE group, encoded as a CBOR array. Possible formats and values are the same ones admitted for the 'sign_params' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'pairwise_mode', encoded as a CBOR simple value. Its value is "true" (0xf5) if the OSCORE group uses the pairwise mode of Group OSCORE [[I-D.ietf-core-oscore-groupcomm](#)], or "false" (0xf4) otherwise.

*'alg', which is formatted as follows. If the configuration parameter 'pairwise_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the AEAD Algorithm used in the OSCORE group to encrypt messages protected with the pairwise mode, encoded as a CBOR text string or a CBOR integer. Possible values are the same ones admitted for the 'alg' parameter of the

Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'ecdh_alg', which is formatted as follows. If the configuration parameter 'pairwise_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the Pairwise Key Agreement Algorithm used in the OSCORE group, encoded as a CBOR text string or a CBOR integer. Possible values are the same ones admitted for the 'ecdh_alg' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

*'ecdh_params', which is formatted as follows. If the configuration parameter 'pairwise_mode' has value "false" (0xf4), this parameter has as value the CBOR simple value "null" (0xf6). Otherwise, this parameter specifies the parameters for the Pairwise Key Agreement Algorithm used in the OSCORE group, encoded as a CBOR array. Possible formats and values are the same ones admitted for the 'ecdh_params' parameter of the Group_OSCORE_Input_Material object, defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

The CBOR map MAY include the following configuration parameters, whose CBOR abbreviations are defined in [Section 9.1](#) of this document.

*'det_req', encoded as a CBOR simple value. Its value is "true" (0xf5) if the OSCORE group uses deterministic requests as defined in [[I-D.amsuess-core-cachable-oscore](#)], or "false" (0xf4) otherwise. This parameter MUST NOT be present if the configuration parameter 'group_mode' has value "false" (0xf4).

*'det_hash_alg', encoded as a CBOR integer or text string. If present, this parameter specifies the Hash Algorithm used in the OSCORE group when producing deterministic requests, as defined in [[I-D.amsuess-core-cachable-oscore](#)]. This parameter takes values from the "Value" column of the "COSE Algorithms" Registry [[COSE.Algorithms](#)].

This parameter MUST NOT be present if the configuration parameter 'det_req' is not present or if it is present with value "false" (0xf4). If the configuration parameter 'det_req' is present with value "true" (0xf5) and 'det_hash_alg' is not present, the choice of the Hash Algorithm to use when producing deterministic requests is left to the Group Manager.

5.1.2. Status Properties

The CBOR map MUST include the following status parameters:

- *'rt', with value the resource type "core.osc.gconf" associated with group-configuration resources, encoded as a CBOR text string.
- *'active', encoding the CBOR simple value "true" (0xf5) if the OSCORE group is currently active, or the CBOR simple value "false" (0xf4) otherwise. This parameter is defined in [Section 9.1](#) of this document.
- *'group_name', with value the group name of the OSCORE group encoded as a CBOR text string. This parameter is defined in [Section 9.1](#) of this document.
- *'group_title', with value either a human-readable description of the OSCORE group encoded as a CBOR text string, or the CBOR simple value "null" (0xf6) if no description is specified. This parameter is defined in [Section 9.1](#) of this document.
- *'ace-groupcomm-profile', defined in Section 4.3.1 of [[I-D.ietf-ace-key-groupcomm](#)], with value "coap_group_oscore_app" defined in [Section 25.5](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)] encoded as a CBOR integer.
- *'exp', defined in [Section 4.3.1](#) of [[I-D.ietf-ace-key-groupcomm](#)].
- *'app_groups', with value a list of names of application groups, encoded as a CBOR array. Each element of the array is a CBOR text string, specifying the name of an application group using the OSCORE group as security group (see [Section 2.1](#) of [[I-D.ietf-core-groupcomm-bis](#)]).
- *'joining_uri', with value the URI of the group-membership resource for joining the newly created OSCORE group as per [Section 6.2](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)], encoded as a CBOR text string. This parameter is defined in [Section 9.1](#) of this document.

The CBOR map MAY include the following status parameters:

- *'group_policies', defined in [Section 4.3.1](#) of [[I-D.ietf-ace-key-groupcomm](#)], and consistent with the format and content defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].
- *'max_stale_sets', defined in [Section 9.1](#) of this document and encoded as a CBOR unsigned integer, with value strictly greater than 1. With reference to [Section 2.2.1](#) of [[I-D.ietf-ace-key-](#)

[groupcomm-oscure](#)], this parameter specifies N, i.e., the maximum number of sets of stale OSCORE Sender IDs that the Group Manager stores in the collection associated with the group.

*'as_uri', defined in [Section 9.1](#) of this document, specifies the URI of the Authorization Server associated with the Group Manager for the OSCORE group, encoded as a CBOR text string. Candidate group members will have to obtain an Access Token from that Authorization Server, before starting the joining process with the Group Manager to join the OSCORE group (see [Sections 4](#) and [6](#) of [\[I-D.ietf-ace-key-groupcomm-oscure\]](#)).

5.2. Default Values

This section defines the default values that the Group Manager assumes for configuration and status parameters.

5.2.1. Configuration Parameters

For each configuration parameter, the Group Manager MUST use a pre-configured default value, if none is specified by the Administrator. In particular:

*For 'group_mode', the Group Manager SHOULD use the CBOR simple value "true" (0xf5).

*If 'group_mode' has value "true" (0xf5), the Group Manager SHOULD use the same default values defined in [Section 23.2](#) of [\[I-D.ietf-ace-key-groupcomm-oscure\]](#) for the parameters 'sign_enc_alg', 'sign_alg' and 'sign_params'.

*If 'group_mode' has value "true" (0xf5), the Group Manager SHOULD use the CBOR simple value "false" (0xf4) for the parameter 'det_req'.

*If 'det_req' has value "true" (0xf5), the Group Manager SHOULD use SHA-256 (COSE algorithm encoding: -16) as default value for the parameter 'det_hash_alg'.

*For 'pairwise_mode', the Group Manager SHOULD use the CBOR simple value "false" (0xf4).

*If 'pairwise_mode' has value "true" (0xf5), the Group Manager SHOULD use the same default values defined in [Section 23.3](#) of [\[I-D.ietf-ace-key-groupcomm-oscure\]](#) for the parameters 'alg', 'ecdh_alg' and 'ecdh_params'.

*For any other configuration parameter, the Group Manager SHOULD use the same default values defined in [Section 23.1](#) of [\[I-D.ietf-ace-key-groupcomm-oscure\]](#).

5.2.2. Status Parameters

For the following status parameters, the Group Manager MUST use a pre-configured default value, if none is specified by the Administrator. In particular:

*For 'active', the Group Manager SHOULD use the CBOR simple value "false" (0xf4).

*For 'group_title', the Group Manager SHOULD use the CBOR simple value "null" (0xf6).

*For 'app_groups', the Group Manager SHOULD use the empty CBOR array.

*For 'group_policies', the Group Manager SHOULD use the default values defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

6. Interactions with the Group Manager

This section describes the operations available on the group-collection resource and the group-configuration resources.

When custom CBOR is used, the Content-Format in messages containing a payload is set to application/ace-groupcomm+cbor, defined in [Section 11.2](#) of [[I-D.ietf-ace-key-groupcomm](#)]. Furthermore, the entry labels defined in [Section 9.1](#) of this document MUST be used, when specifying the corresponding configuration and status parameters.

6.1. Retrieve the Full List of Group Configurations

The Administrator can send a GET request to the group-collection resource, in order to retrieve a list of the existing OSCORE groups at the Group Manager. This is returned as a list of links to the corresponding group-configuration resources.

The Group Manager MUST prepare the list L to include in the response as follows. For each group-configuration resource R:

1. The Group Manager considers the group name GROUPNAME of the OSCORE group associated to R.
2. The Group Manager retrieves the stored Access Token for the Administrator. Then, it checks whether GROUPNAME matches with the group name pattern specified in any scope entry of the 'scope' claim in the Access Token.
3. The link to the group-configuration resource R is added to the list L only in case of a positive match.

Example in Link Format:

```
=> 0.01 GET
    Uri-Path: manage

<= 2.05 Content
    Content-Format: 40 (application/link-format)

    <coap://[2001:db8::ab]/manage/gp1>;rt="core.osc.gconf",
    <coap://[2001:db8::ab]/manage/gp2>;rt="core.osc.gconf",
    <coap://[2001:db8::ab]/manage/gp3>;rt="core.osc.gconf"
```

Example in CoRAL:

```
=> 0.01 GET
    Uri-Path: manage

<= 2.05 Content
    Content-Format: TBD1 (application/coral+cbor)

    #using <http://coreapps.org/core.osc.gcoll#>
    #base </manage/>
    item <gp1>
    item <gp2>
    item <gp3>
```

6.2. Retrieve a List of Group Configurations by Filters

The Administrator can send a FETCH request to the group-collection resource, in order to retrieve a list of the existing OSCORE groups that fully match a set of specified filter criteria. This is returned as a list of links to the corresponding group-configuration resources.

When custom CBOR is used, the set of filter criteria is specified in the request payload as a CBOR map, whose possible entries are specified in [Section 5.1](#) and use the same abbreviations defined in [Section 9.1](#). Entry values are the ones admitted for the corresponding labels in the POST request for creating a group configuration (see [Section 6.3](#)). A valid request MUST NOT include the same entry multiple times.

When CoRAL is used, the filter criteria are specified in the request payload with top-level elements, each of which corresponds to an entry specified in [Section 5.1](#), with the exception of the 'app_groups' status parameter. If names of application groups are used as filter criteria, each element of the 'app_groups' array from the status properties is included as a separate element with name 'app_group'. With the exception of the 'app_group' element, a valid request MUST NOT include the same element multiple times. Element

values are the ones admitted for the corresponding labels in the POST request for creating a group configuration (see [Section 6.3](#)).

The Group Manager MUST prepare the list L to include in the response as follows.

1. The Group Manager prepares a preliminary version of the list L, as specified in [Section 6.1](#) for the processing of a GET request to the group-collection resource.
2. The Group Manager applies the filter criteria specified in the FETCH request to the list L from the previous step. The result is the list L to include in the response.

Example in custom CBOR and Link Format:

=> 0.05 FETCH

Uri-Path: manage

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "group_mode" : true,
  "sign_enc_alg" : 10,
  "hkdf" : 5
}
```

<= 2.05 Content

Content-Format: 40 (application/link-format)

```
<coap://[2001:db8::ab]/manage/gp1>;rt="core.osc.gconf",
<coap://[2001:db8::ab]/manage/gp2>;rt="core.osc.gconf",
<coap://[2001:db8::ab]/manage/gp3>;rt="core.osc.gconf"
```

Example in CoRAL:

```
=> 0.05 FETCH
    Uri-Path: manage
    Content-Format: TBD1 (application/coral+cbor)

    group_mode true
    sign_enc_alg 10
    hkdf 5

<= 2.05 Content
    Content-Format: TBD1 (application/coral+cbor)

    #using <http://coreapps.org/core.osc.gcoll#>
    #base </manage/>
    item <gp1>
    item <gp2>
    item <gp3>
```

6.3. Create a New Group Configuration

The Administrator can send a POST request to the group-collection resource, in order to create a new OSCORE group at the Group Manager. The request MUST specify the intended group name GROUPNAME, and MAY specify the intended group title together with pieces of information concerning the group configuration.

When custom CBOR is used, the request payload is a CBOR map, whose possible entries are specified in [Section 5.1](#) and use the same abbreviations defined in [Section 9.1](#).

When CoRAL is used, each element of the request payload corresponds to an entry specified in [Section 5.1](#), with the exception of the 'app_groups' status parameter (see below).

In particular:

- *The payload MAY include any of the configuration parameter defined in [Section 5.1.1](#).
- *The payload MUST include the status parameter 'group_name' defined in [Section 5.1.2](#) and specifying the intended group name.
- *The payload MAY include any of the status parameter 'group_title', 'max_stale_sets', 'exp', 'app_groups', 'group_policies', 'as_uri' and 'active' defined in [Section 5.1.2](#).

When CoRAL is used, each element of the 'app_groups' array from the status properties is included as a separate element with name 'app_group'.

*The payload MUST NOT include any of the status parameter 'rt', 'ace-groupcomm-profile' and 'joining_uri' defined in [Section 5.1.2](#).

Consistently with what is defined at step 4 of [Section 4](#), the Group Manager MUST check whether the group name specified in the 'group_name' parameter matches with the group name pattern specified in any scope entry of the 'scope' claim in the stored Access Token for the Administrator. In case of a positive match, the Group Manager MUST check whether the permission set in the found scope entry specifies the permission "Create".

If the verification above fails (i.e., there are no matching scope entries specifying the "Create" permission), the Group Manager MUST reply with a 4.03 (Forbidden) error response. The response MUST have Content-Format set to application/ace-groupcomm+cbor and is formatted as defined in [Section 4.1.2](#) of [[I-D.ietf-ace-key-groupcomm](#)].

Otherwise, if any of the following occurs, the Group Manager MUST respond with a 4.00 (Bad Request) response.

*Any of the received parameters is specified multiple times, with the exception of the 'app_group' element when using CoRAL.

*Any of the received parameters is not recognized, or not valid, or not consistent with respect to other related parameters.

*The Group Manager does not trust the Authorization Server with URI specified in the 'as_uri' parameter, and has no alternative Authorization Server to consider for the OSCORE group to create.

After a successful processing of the POST request, the Group Manager performs the following actions.

If the 'group_name' parameter specifies the group name of an already existing OSCORE group, the Group Manager MUST find an alternative name for the new OSCORE group to create. Note that the final decision about the name assigned to the new OSCORE group is always of the Group Manager, which may have more constraints than the Administrator can be aware of, possibly beyond the availability of suggested names.

If the Group Manager has selected a name GROUPNAME different from the name GROUPNAME* indicated in the parameter 'group_name' of the request, then the following conditions MUST hold.

*The chosen name GROUPNAME is available to assign; and

If GROUPNAME matches with the group name pattern of certain scope entries from the 'scope' claim in the stored Access Token for the Administrator, then the chosen group name GROUPNAME also matches with each of those group name patterns.

If the Group Manager does not find any group name for which both the above conditions hold, the Group Manager MUST respond with a 5.03 (Service Unavailable) response.

Otherwise, the Group Manager creates a new group-configuration resource, accessible to the Administrator at /manage/GROUPNAME, where GROUPNAME is the name of the OSCORE group as either indicated in the parameter 'group_name' of the request or uniquely assigned by the Group Manager.

The value of the status parameter 'rt' is set to "core.osc.gconf". The values of other parameters specified in the request are used as group configuration information for the newly created OSCORE group. For each parameter not specified in the request, the Group Manager MUST use default values as specified in [Section 5.2](#).

After that, the Group Manager creates a new group-membership resource accessible at ace-group/GROUPNAME to nodes that want to join the OSCORE group, as specified in [Section 6.2](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]. Note that such group membership-resource comprises a number of sub-resources intended to current group members, as defined in [Section 4.1](#) of [[I-D.ietf-ace-key-groupcomm](#)] and [Section 5](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

From then on, the Group Manager will rely on the current group configuration to build the Joining Response message defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)], when handling the joining of a new group member. Furthermore, the Group Manager generates the following pieces of information, and assigns them to the newly created OSCORE group.

*The OSCORE Master Secret.

*The OSCORE Master Salt (optionally).

*The Group ID, used as OSCORE ID Context, which MUST be unique within the set of OSCORE groups under the Group Manager.

Finally, the Group Manager replies to the Administrator with a 2.01 (Created) response. The Location-Path option MUST be included in the response, indicating the location of the just created group-configuration resource. The response MUST NOT include a Location-Query option.

The response payload specifies the parameters 'group_name', 'joining_uri' and 'as_uri', from the status properties of the newly created OSCORE group (see [Section 5.1](#)), as detailed below.

When custom CBOR is used, the response payload is a CBOR map, where entries use the same abbreviations defined in [Section 9.1](#). When CoRAL is used, the response payload includes one element for each specified parameter.

- *'group_name', with value the group name of the OSCORE group. This value can be different from the group name possibly specified by the Administrator in the POST request, and reflects the final choice of the Group Manager as 'group_name' status property for the OSCORE group. This parameter MUST be included.

- *'joining_uri', with value the URI of the group-membership resource for joining the newly created OSCORE group. This parameter MUST be included.

- *'as_uri', with value the URI of the Authorization Server associated with the Group Manager for the newly created OSCORE group. This parameter MUST be included if specified in the status properties of the group. This value can be different from the URI possibly specified by the Administrator in the POST request, and reflects the final choice of the Group Manager as 'as_uri' status property for the OSCORE group.

If the POST request did not specify certain parameters and the Group Manager used default values different from the ones recommended in [Section 5.2](#), then the response payload MUST include also those parameters, specifying the values chosen by the Group Manager for the current group configuration.

The Group Manager can register the link to the group-membership resource with URI specified in 'joining_uri' to a Resource Directory [[I-D.ietf-core-resource-directory](#)][[I-D.hartke-t2trg-coral-reef](#)], as defined in [Section 2](#) of [[I-D.tiloca-core-oscore-discovery](#)]. The Group Manager considers the current group configuration when specifying additional information for the link to register.

Alternatively, the Administrator can perform the registration in the Resource Directory on behalf of the Group Manager, acting as Commissioning Tool. The Administrator considers the following when specifying additional information for the link to register.

- *The name of the OSCORE group MUST take the value specified in 'group_name' from the 2.01 (Created) response.

- *The names of the application groups using the OSCORE group MUST take the values possibly specified by the elements of the

'app_groups' parameter (when custom CBOR is used) or by the different 'app_group' elements (when CoRAL is used) in the POST request.

*If also registering a related link to the Authorization Server associated with the OSCORE group, the related link MUST have as link target the URI in 'as_uri' from the 2.01 (Created) response, if the 'as_uri' parameter was included in the response.

*Every other information element describing the current group configuration MUST take the value that the Administrator specified in the POST request. If a certain parameter was not specified in the POST request, the Administrator MUST use either the value specified in the the 2.01 (Created) response, if the Group Manager specified one, or the corresponding default value recommended in [Section 5.2.1](#) otherwise.

Note that, compared to the Group Manager, the Administrator is less likely to remain closely aligned with possible changes and updates that would require a prompt update to the registration in the Resource Directory. This applies especially to the address of the Group Manager, as well as the URI of the group-membership resource or of the Authorization Server associated with the Group Manager.

Therefore, it is RECOMMENDED that registrations of links to group-membership resources in the Resource Directory are made (and possibly updated) directly by the Group Manager, rather than by the Administrator.

Example in custom CBOR:

=> 0.02 POST

Uri-Path: manage

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "sign_enc_alg" : 10,
  "hkdf" : 5,
  "pairwise_mode" : true,
  "active" : true,
  "group_name" : "gp4",
  "group_title" : "rooms 1 and 2",
  "app_groups": : ["room1", "room2"],
  "as_uri" : "coap://as.example.com/token"
}
```

<= 2.01 Created

Location-Path: manage

Location-Path: gp4

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "group_name" : "gp4",
  "joining_uri" : "coap://[2001:db8::ab]/ace-group/gp4/",
  "as_uri" : "coap://as.example.com/token"
}
```

Example in CoRAL:

```

=> 0.02 POST
  Uri-Path: manage
  Content-Format: TBD1 (application/coral+cbor)

  #using <http://coreapps.org/core.osc.gconf#>
  sign_enc_alg 10
  hkdf 5
  pairwise_mode true
  active true
  group_name "gp4"
  group_title "rooms 1 and 2"
  app_group "room1"
  app_group "room2"
  as_uri <coap://as.example.com/token>

<= 2.01 Created
  Location-Path: manage
  Location-Path: gp4
  Content-Format: TBD1 (application/coral+cbor)

  #using <http://coreapps.org/core.osc.gconf#>
  group_name "gp4"
  joining_uri <coap://[2001:db8::ab]/ace-group/gp4/>
  as_uri <coap://as.example.com/token>

```

6.4. Retrieve a Group Configuration

The Administrator can send a GET request to the group-configuration resource `manage/GROUPNAME` associated with an OSCORE group with group name `GROUPNAME`, in order to retrieve the complete current configuration of that group.

Consistently with what is defined at step 4 of [Section 4](#), the Group Manager MUST check whether `GROUPNAME` matches with the group name pattern specified in any scope entry of the 'scope' claim in the stored Access Token for the Administrator. In case of a positive match, the Group Manager MUST check whether the permission set in the found scope entry specifies the permission "Read".

If the verification above fails (i.e., there are no matching scope entries specifying the "Read" permission), the Group Manager MUST reply with a 4.03 (Forbidden) error response. The response MUST have Content-Format set to `application/ace-groupcomm+cbor` and is formatted as defined in [Section 4.1.2](#) of [[I-D.ietf-ace-key-groupcomm](#)].

Otherwise, after a successful processing of the GET request, the Group Manager replies to the Administrator with a 2.05 (Content) response. The response has as payload the representation of the group configuration as specified in [Section 5.1](#). The exact content

of the payload reflects the current configuration of the OSCORE group. This includes both configuration properties and status properties.

When custom CBOR is used, the response payload is a CBOR map, whose possible entries are specified in [Section 5.1](#) and use the same abbreviations defined in [Section 9.1](#).

When CoRAL is used, the response payload includes one element for each entry specified in [Section 5.1](#), with the exception of the 'app_groups' status parameter. That is, each element of the 'app_groups' array from the status properties is included as a separate element with name 'app_group'.

Example in custom CBOR:

```
=> 0.01 GET
  Uri-Path: manage
  Uri-Path: gp4

<= 2.05 Content
  Content-Format: TBD2 (application/ace-groupcomm+cbor)

{
  "hkdf" : 5,
  "cred_fmt" : 33,
  "group_mode" : true,
  "sign_enc_alg" : 10,
  "sign_alg" : -8,
  "sign_params" : [[1], [1, 6]],
  "pairwise_mode" : true,
  "alg" : 10,
  "ecdh_alg" : -27,
  "ecdh_params" : [[1], [1, 6]],
  "rt" : "core.osc.gconf",
  "active" : true,
  "group_name" : "gp4",
  "group_title" : "rooms 1 and 2",
  "ace-groupcomm-profile" : "coap_group_oscore_app",
  "max_stale_sets" : 3,
  "exp" : 1360289224,
  "app_groups": : ["room1", "room2"],
  "joining_uri" : "coap://[2001:db8::ab]/ace-group/gp4/",
  "as_uri" : "coap://as.example.com/token"
}
```

Example in CoRAL:

```

=> 0.01 GET
    Uri-Path: manage
    Uri-Path: gp4

<= 2.05 Content
    Content-Format: TBD1 (application/coral+cbor)

    #using <http://coreapps.org/core.osc.gconf#>
    hkdf 5
    cred_fmt 33
    group_mode true
    sign_enc_alg 10
    sign_alg -8
    sign_params.alg_capab.key_type 1
    sign_params.key_type_capab.key_type 1
    sign_params.key_type_capab.curve 6
    pairwise_mode true
    alg 10
    ecdh_alg -27
    ecdh_params.alg_capab.key_type 1
    ecdh_params.key_type_capab.key_type 1
    ecdh_params.key_type_capab.curve 6
    rt "core.osc.gconf",
    active true
    group_name "gp4"
    group_title "rooms 1 and 2"
    ace-groupcomm-profile "coap_group_oscure_app"
    max_stale_sets 3
    exp 1360289224
    app_group "room1"
    app_group "room2"
    joining_uri <coap://[2001:db8::ab]/ace-group/gp4/>
    as_uri <coap://as.example.com/token>

```

6.5. Retrieve Part of a Group Configuration by Filters

The Administrator can send a FETCH request to the group-configuration resource `manage/GROUPNAME` associated with an OSCORE group with group name `GROUPNAME`, in order to retrieve part of the current configuration of that group.

When custom CBOR is used, the request payload is a CBOR map, which contains the following fields:

- *'conf_filter', defined in [Section 9.1](#) of this document and encoded as a CBOR array. Each element of the array specifies one requested configuration parameter or status parameter of the current group configuration (see [Section 5.1](#)), using the corresponding abbreviation defined in [Section 9.1](#).

When CoRAL is used, the request payload includes one element for each requested configuration parameter or status parameter of the current group configuration (see [Section 5.1](#)). All the specified elements have no value.

The Group Manager MUST perform the same authorization checks defined for the processing of a GET request to a group-configuration resource in [Section 6.4](#). That is, the Group Manager MUST verify that the Administrator has been granted a "Read" permission applicable to the targeted group-configuration resource.

After a successful processing of the FETCH request, the Group Manager replies to the Administrator with a 2.05 (Content) response. The response has as payload a partial representation of the group configuration (see [Section 5.1](#)). The exact content of the payload reflects the current configuration of the OSCORE group, and is limited to the configuration properties and status properties requested by the Administrator in the FETCH request.

The response payload includes the requested configuration parameters and status parameters, and is formatted as in the response payload of a GET request to a group-configuration resource (see [Section 6.4](#)).

Example in custom CBOR:

=> 0.05 FETCH

Uri-Path: manage

Uri-Path: gp4

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "conf_filter" : ["sign_enc_alg",
                  "hkdf",
                  "pairwise_mode",
                  "active",
                  "group_title",
                  "app_groups"]
}
```

<= 2.05 Content

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "sign_enc_alg" : 10,
  "hkdf" : 5,
  "pairwise_mode" : true,
  "active" : true,
  "group_title" : "rooms 1 and 2",
  "app_groups": : ["room1", "room2"]
}
```

Example in CoRAL:

```
=> 0.05 FETCH
Uri-Path: manage
Uri-Path: gp4
Content-Format: TBD1 (application/coral+cbor)
```

```
#using <http://coreapps.org/core.osc.gconf#>
sign_enc_alg
hkdf
pairwise_mode
active
group_title
app_groups
```

```
<= 2.05 Content
Content-Format: TBD1 (application/coral+cbor)
```

```
#using <http://coreapps.org/core.osc.gconf#>
sign_enc_alg 10
hkdf 5
pairwise_mode true
active true
group_title "rooms 1 and 2"
app_group "room1"
app_group "room2"
```

6.6. Overwrite a Group Configuration

The Administrator can send a PUT request to the group-configuration resource associated with an OSCORE group, in order to overwrite the current configuration of that group with a new one. The payload of the request has the same format of the POST request defined in [Section 6.3](#), with the exception that the configuration parameters 'group_mode' and 'pairwise_mode' as well as the status parameter 'group_name' MUST NOT be included.

The error handling for the PUT request is the same as for the POST request defined in [Section 6.3](#), with the following difference in terms of authorization checks.

Consistently with what is defined at step 4 of [Section 4](#), the Group Manager MUST check whether GROUPNAME matches with the group name pattern specified in any scope entry of the 'scope' claim in the stored Access Token for the Administrator. In case of a positive match, the Group Manager MUST check whether the permission set in the found scope entry specifies the permission "Write".

If the verification above fails (i.e., there are no matching scope entries specifying the "Write" permission), the Group Manager MUST reply with a 4.03 (Forbidden) error response. The response MUST have Content-Format set to application/ace-groupcomm+cbor and is

formatted as defined in [Section 4.1.2](#) of [[I-D.ietf-ace-key-groupcomm](#)].

If no error occurs and the PUT request is successfully processed, the Group Manager performs the following actions.

First, the Group Manager updates the group-configuration resource, consistently with the values indicated in the PUT request from the Administrator. For each parameter not specified in the PUT request, the Group Manager MUST use default values as specified in [Section 5.2](#).

If a new value N' is specified for the 'max_stale_sets' status parameter and N' is smaller than the current value N, the Group Manager preserves the (up to) N' most recent sets in the collection of sets of stale OSCORE Sender IDs associated with the group, and deletes any possible older set from the collection (see [Section 2.2.1](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]).

From then on, the Group Manager relies on the latest updated configuration to build the Joining Response message defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)], when handling the joining of a new group member. Similarly, the Group Manager relies on the new group configuration when building responses specifying (part of) the group configuration to a current group member. For instance, this applies when a group member retrieves from the Group Manager the updated group keying material (see [Section 8](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]) or the current group status (see [Section 16](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]).

Then, the Group Manager replies to the Administrator with a 2.04 (Changed) response. The payload of the response has the same format of the 2.01 (Created) response defined in [Section 6.3](#).

If the PUT request did not specify certain parameters and the Group Manager used default values different from the ones recommended in [Section 5.2](#), then the response payload MUST include also those parameters, specifying the values chosen by the Group Manager for the current group configuration.

If the link to the group-membership resource was registered in the Resource Directory [[I-D.ietf-core-resource-directory](#)], the GM is responsible to refresh the registration, as defined in [Section 3](#) of [[I-D.tiloca-core-oscore-discovery](#)].

Alternatively, the Administrator can update the registration in the Resource Directory on behalf of the Group Manager, acting as

Commissioning Tool. The Administrator considers the following when specifying additional information for the link to update.

- *The name of the OSCORE group MUST take the value specified in 'group_name' from the 2.04 (Changed) response.
- *The names of the application groups using the OSCORE group MUST take the values possibly specified by the elements of the 'app_groups' parameter (when custom CBOR is used) or by the different 'app_group' elements (when CoRAL is used) in the PUT request.
- *If also registering a related link to the Authorization Server associated with the OSCORE group, the related link MUST have as link target the URI in 'as_uri' from the 2.04 (Changed) response, if the 'as_uri' parameter was included in the response.
- *Every other information element describing the current group configuration MUST take the value that the Administrator specified in the PUT request. If a certain parameter was not specified in the PUT request, the Administrator MUST use either the value specified in the 2.04 (Changed) response, if the Group Manager specified one, or the corresponding default value recommended in [Section 5.2.1](#) otherwise.

As discussed in [Section 6.3](#), it is RECOMMENDED that registrations of links to group-membership resources in the Resource Directory are made (and possibly updated) directly by the Group Manager, rather than by the Administrator.

Example in custom CBOR:

=> 0.03 PUT

Uri-Path: manage

Uri-Path: gp4

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "sign_enc_alg" : 11,
  "hkdf" : 5
}
```

<= 2.04 Changed

Content-Format: TBD2 (application/ace-groupcomm+cbor)

```
{
  "group_name" : "gp4",
  "joining_uri" : "coap://[2001:db8::ab]/ace-group/gp4/",
  "as_uri" : "coap://as.example.com/token"
}
```

Example in CoRAL:

```
=> 0.03 PUT
  Uri-Path: manage
  Uri-Path: gp4
  Content-Format: TBD1 (application/coral+cbor)

  #using <http://coreapps.org/core.osc.gconf#>
  sign_enc_alg 11
  hkdf 5

<= 2.04 Changed
  Content-Format: TBD1 (application/coral+cbor)

  #using <http://coreapps.org/core.osc.gconf#>
  group_name "gp4"
  joining_uri <coap://[2001:db8::ab]/ace-group/gp4/>
  as_uri <coap://as.example.com/token>
```

6.6.1. Effects on Joining Nodes

After having overwritten a group configuration, if the value of the status parameter 'active' is changed from "true" (0xf5) to "false" (0xf4), the Group Manager MUST stop admitting new members in the OSCORE group. In particular, until the status parameter 'active' is changed back to "true" (0xf5), the Group Manager MUST respond to a Joining Request with a 5.03 (Service Unavailable) response, as defined in [Section 6.3](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#).

If the value of the status parameter 'active' is changed from "false" (0xf4) to "true" (0xf5), the Group Manager resumes admitting new members in the OSCORE group, by processing their Joining Requests (see [Section 6.3](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)).

6.6.2. Effects on the Group Members

After having overwritten a group configuration, the Group Manager informs the members of the OSCORE group, over the pairwise secure communication channels established when joining the group (see [Section 6](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)).

To this end, the Group Manager can individually target the 'control_uri' URI of each group member (see [Section 4.3.1](#) of [\[I-D.ietf-ace-key-groupcomm\]](#)), if provided by the intended recipient upon joining the OSCORE group (see [Section 6.2](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)). To this end, messages sent by the Group Manager to each group member MUST have Content-Format set to application/ace-groupcomm+cbor, and MUST be formatted as the Joining Response

defined in [Section 6.4](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)], with the following differences.

- *Only the parameters 'gkty', 'key', 'num', 'exp' and 'ace-groupcomm-profile' are present.

- *The 'key' parameter includes only the parameters 'hkdf', 'cred_fmt', 'sign_enc_alg', 'sign_alg', 'sign_params', 'alg', 'ecdh_alg' and 'ecdh_params', with values reflecting the new configuration of the OSCORE group.

Alternatively, group members can subscribe for updates to the group-membership resource of the OSCORE group, e.g., by using CoAP Observe [[RFC7641](#)].

If the value of the status parameter 'active' is changed from "true" (0xf5) to "false" (0xf4):

- *The Group Manager MUST stop accepting requests for new individual keying material from current group members (see [Section 9](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]). In particular, until the status parameter 'active' is changed back to "true" (0xf5), the Group Manager MUST respond to a Key Renewal Request with a 5.03 (Service Unavailable) response, as defined in [Section 9](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

- *The Group Manager MUST stop accepting updated authentication credentials uploaded by current group members (see [Section 11](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]). In particular, until the status parameter 'active' is changed back to "true" (0xf5), the Group Manager MUST respond to a Public Key Update Request with a 5.03 (Service Unavailable) response, as defined in [Section 11](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)].

Every group member, upon learning that the OSCORE group has been deactivated (i.e., 'active' has value "false" (0xf4)), SHOULD stop communicating in the group.

Every group member, upon learning that the OSCORE group has been reactivated (i.e., 'active' has value "true" (0xf5) again), can resume communicating in the group.

Every group member, upon receiving updated values for 'hkdf', 'sign_enc_alg' and 'alg', MUST either:

- *Leave the OSCORE group (see [Section 18](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]), e.g., if not supporting the indicated new algorithms; or

*Use the new parameter values, and accordingly re-derive the OSCORE Security Context for the OSCORE group (see [Section 2](#) of [\[I-D.ietf-core-oscore-groupcomm\]](#)).

Every group member, upon receiving updated values for 'cred_fmt', 'sign_alg', 'sign_params', 'ecdh_alg' and 'ecdh_params' MUST either:

*Leave the OSCORE group, e.g., if not supporting the indicated new format, algorithms, parameters and encoding; or

*Leave the OSCORE group and rejoin it (see [Section 6](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)). When rejoining the group, a new authentication credential in the indicated format used in the OSCORE group MUST be provided to the Group Manager. The authentication credential as well as the included public key MUST be compatible with the indicated algorithms and parameters.

*Use the new parameter values, and, if required, perform the following actions.

- Provide the Group Manager with a new authentication credential to use in the OSCORE group (see [Section 11](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)). The new authentication credential MUST be in the indicated format used in the OSCORE group. The new authentication credential as well as the included public key MUST be compatible with the indicated algorithms and parameters.

- Retrieve from the Group Manager the new Group Manager's authentication credential (see [Section 12](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)). The new Group Manager's authentication credential is in the indicated format used in the OSCORE group. The new authentication credential as well as the included public key are compatible with the indicated algorithms and parameters.

6.7. Selective Update of a Group Configuration

The Administrator can send a PATCH/iPATCH request [[RFC8132](#)] to the group-configuration resource associated with an OSCORE group, in order to update the value of only part of the group configuration.

The request payload has the same format of the PUT request defined in [Section 6.6](#), with the difference that it MAY also specify names of application groups to be removed from or added to the 'app_groups' status parameter. The names of such application groups are provided as defined below.

*When custom CBOR is used, the CBOR map in the request payload includes the field 'app_groups_diff'. This field MUST NOT be

present multiple times, and it is encoded as a CBOR array including the following two elements.

- The first element is a CBOR array, namely 'app_groups_del'. Each of its elements is a CBOR text string, with value the name of an application group to remove from the 'app_groups' status parameter.
- The second element is a CBOR array, namely 'app_groups_add'. Each of its elements is a CBOR text string, with value the name of an application group to add to the 'app_groups' status parameter.

The CDDL definition [[RFC8610](#)] of the CBOR array 'app_groups_diff' formatted as in the response from the Group Manager is provided below.

```
app-group-name = tstr
name-patch = [* app-group-name]
app_groups_diff = [app_groups_del: name-patch,
                   app_groups_add: name-patch]
```

Figure 3: CDDL definition of the 'app_groups_diff' field

The Group Manager MUST respond with a 4.00 (Bad Request) response, in case both the inner CBOR arrays 'app_groups_del' and 'app_groups_add' are empty, or in case the 'app_groups_diff' field occurs more than once.

The Group Manager MUST respond with a 4.00 (Bad Request) response, in case the CBOR map in the request payload includes both the 'app_groups' field and the 'app_groups_diff' field.

*When CoRAL is used, the request payload includes the following top-level elements.

- 'app_group_del', with value a text string specifying the name of an application group to remove from the 'app_groups' status parameter. This element can be included multiple times.
- 'app_group_add', with value a text string specifying the name of an application group to add to the 'app_groups' status parameter. This element can be included multiple times.

The Group Manager MUST respond with a 4.00 (Bad Request) response, in case the request payload includes both any 'app_group' element as well as any 'app_group_del' and/or 'app_group_add' element.

The error handling for the PATCH/iPATCH request is the same as for the PUT request defined in [Section 6.6](#), with the following additions.

- *The set of group configuration parameters to update MUST NOT be empty. That is, the Group Manager MUST respond with a 4.00 (Bad Request) response, if the request payload includes an empty CBOR map (when custom CBOR is used) or no elements (when CoRAL is used).

- *If the Request-URI does not point to an existing group-configuration resource, the Group Manager MUST NOT create a new resource, and MUST respond with a 4.04 (Not Found) response.

- *When applying the specified updated values would yield an inconsistent group configuration, the Group Manager MUST respond with a 4.09 (Conflict) response.

The response, MAY include the current representation of the group configuration resource, like when responding to a GET request as defined in [Section 6.4](#). Otherwise, the response SHOULD include a diagnostic payload with additional information for the Administrator to recognize the source of the conflict.

- *When the request uses specifically the iPATCH method, the Group Manager MUST respond with a 4.00 (Bad Request) response, in case:

- When custom CBOR is used, the CBOR map includes the parameter 'app_groups_diff'; or

- When CoRAL is used, any element 'app_group_del' and/or 'app_group_add' is included.

Furthermore, the Group Manager MUST perform the same authorization checks defined for the processing of a PUT request to a group-configuration resource in [Section 6.6](#). That is, the Group Manager MUST verify that the Administrator has been granted a "Write" permission applicable to the targeted group-configuration resource.

If no error occurs and the PATCH/iPATCH request is successfully processed, the Group Manager performs the following actions.

First, the Group Manager updates the group-configuration resource, consistently with the values indicated in the PATCH/iPATCH request from the Administrator.

Unlike for the PUT request defined in [Section 6.6](#), the Group Manager does not alter the value of configuration parameters and status parameters for which updated values are not specified in the request

payload. In particular, the Group Manager does not assign possible default values to those parameters.

Special processing occurs when updating the 'app_groups' status parameter by difference, as defined below. The Administrator should not expect the Group Manager to add or delete names of application group names according to any particular order.

- *If the name of an application group to add (delete) is specified multiple times, the Group Manager considers it only once for addition to (deletion from) the 'app_groups' status parameter.

- *If the name of an application group to delete is not present in the 'app_groups' status parameter before any change is applied, the Group Manager ignores that name.

- *If the name of an application group to add is already present in the 'app_groups' status parameter before any change is applied, the Group Manager ignores that name.

- *When custom CBOR is used, the Group Manager:

- Deletes from the 'app_groups' status parameter the names of the application groups specified in the inner 'app_groups_del' CBOR array of the 'app_groups_diff' field.

- Adds to the 'app_groups' status parameter the names of the application groups specified in the inner 'app_groups_add' CBOR array of the 'app_groups_diff' field.

- *When CoRAL is used, the Group Manager:

- Deletes from the 'app_groups' status parameter the names of the application groups specified in the different 'app_group_del' elements.

- Adds to the 'app_groups' status parameter the names of the application groups specified in the different 'app_group_add' elements.

After having updated the group-configuration resource, from then on the Group Manager relies on the new group configuration to build the Joining Response message defined in [Section 6.4](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#), when handling the joining of a new group member. Similarly, the Group Manager relies on the new group configuration when building responses specifying (part of) the group configuration to a current group member. For instance, this applies when a group member retrieves from the Group Manager the updated group keying material (see [Section 8](#) of [\[I-D.ietf-ace-key-groupcomm-](#)

[oscore](#)]) or the current group status (see [Section 16](#) of [[I-D.ietf-ace-key-groupcomm-oscore](#)]).

Finally, the Group Manager replies to the Administrator with a 2.04 (Changed) response. The payload of the response has the same format of the 2.01 (Created) response defined in [Section 6.3](#).

The same considerations as for the PUT request defined in [Section 6.6](#) hold also in this case, with respect to refreshing a possible registration of the link to the group-membership resource in the Resource Directory [[I-D.ietf-core-resource-directory](#)].

Example in custom CBOR:

```
=> 0.06 PATCH
Uri-Path: manage
Uri-Path: gp4
Content-Format: TBD2 (application/ace-groupcomm+cbor)

{
  "sign_enc_alg" : 10,
  "app_groups_diff" : [["room1"],
                       ["room3", "room4"]]
}

<= 2.04 Changed
Content-Format: TBD2 (application/ace-groupcomm+cbor)

{
  "group_name" : "gp4",
  "joining_uri" : "coap://[2001:db8::ab]/ace-group/gp4/",
  "as_uri" : "coap://as.example.com/token"
}
```

Example in CoRAL:

```
=> 0.06 PATCH
Uri-Path: manage
Uri-Path: gp4
Content-Format: TBD1 (application/coral+cbor)
```

```
#using <http://coreapps.org/core.osc.gconf#>
sign_enc_alg 10
app_group_del "room1"
app_group_add "room3"
app_group_add "room4"
```

```
<= 2.04 Changed
Content-Format: TBD1 (application/coral+cbor)
```

```
#using <http://coreapps.org/core.osc.gconf#>
group_name "gp4"
joining_uri <coap://[2001:db8::ab]/ace-group/gp4/>
as_uri <coap://as.example.com/token>
```

6.7.1. Effects on Joining Nodes

After having selectively updated part of a group configuration, the effects on candidate joining nodes are the same as defined in [Section 6.6.1](#) for the case of group configuration overwriting.

6.7.2. Effects on the Group Members

After having selectively updated part of a group configuration, the effects on the current group members are the same as defined in [Section 6.6.2](#) for the case of group configuration overwriting.

6.8. Delete a Group Configuration

The Administrator can send a DELETE request to the group-configuration resource, in order to delete that OSCORE group.

Consistently with what is defined at step 4 of [Section 4](#), the Group Manager MUST check whether GROUPNAME matches with the group name pattern specified in any scope entry of the 'scope' claim in the stored Access Token for the Administrator. In case of a positive match, the Group Manager MUST check whether the permission set in the found scope entry specifies the permission "Delete".

If the verification above fails (i.e., there are no matching scope entries specifying the "Delete" permission), the Group Manager MUST reply with a 4.03 (Forbidden) error response. The response MUST have Content-Format set to application/ace-groupcomm+cbor and is formatted as defined in [Section 4.1.2](#) of [[I-D.ietf-ace-key-groupcomm](#)].

Otherwise, the Group Manager continues processing the request, which would be successful only on an inactive OSCORE group. That is, the DELETE request actually yields a successful deletion of the OSCORE group, only if the corresponding status parameter 'active' has current value "false" (0xf4). The Administrator can ensure that, by first performing an update of the group-configuration resource associated with the OSCORE group (see [Section 6.6](#)), and setting the corresponding status parameter 'active' to "false" (0xf4).

If, upon receiving the DELETE request, the current value of the status parameter 'active' is "true" (0xf5), the Group Manager MUST respond with a 4.09 (Conflict) response. The response MUST have Content-Format set to application/ace-groupcomm+cbor and is formatted as defined in [Section 4.1.2](#) of [\[I-D.ietf-ace-key-groupcomm\]](#). The value of the 'error' field MUST be set to 8 ("Group currently active").

After a successful processing of the DELETE request, the Group Manager performs the following actions.

First, the Group Manager deletes the OSCORE group and deallocates both the group-configuration resource as well as the group-membership resource associated with that group.

Then, the Group Manager replies to the Administrator with a 2.02 (Deleted) response.

Example:

=> 0.04 DELETE

Uri-Path: manage

Uri-Path: gp4

<= 2.02 Deleted

6.8.1. Effects on the Group Members

After having deleted an OSCORE group, the Group Manager can inform the group members by means of the following two methods. When contacting a group member, the Group Manager uses the pairwise secure communication association established with that member during its joining process (see [Section 6](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)).

*The Group Manager sends an individual request message to each group member, targeting the respective resource used to perform the group rekeying process (see [Section 20.1](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#)). The Group Manager uses the same format of the Joining Response message in [Section 6.4](#) of [\[I-D.ietf-ace-key-groupcomm-oscore\]](#), where only the parameters 'gkty', 'key' and

'ace-groupcomm-profile' are present, and the 'key' parameter is the empty CBOR map.

*A group member may subscribe for updates to the group-membership resource associated with the OSCORE group. In particular, if this relies on CoAP Observe [[RFC7641](#)], a group member would receive a 4.04 (Not Found) notification response from the Group Manager, since the group-configuration resource has been deallocated upon deleting the OSCORE group (see [Section 6.1](#) of [[I-D.ietf-ace-key-groupcomm](#)]). The response MUST have Content-Format set to application/ace-groupcomm+cbor and is formatted as defined in [Section 4.1.2](#) of [[I-D.ietf-ace-key-groupcomm](#)]. The value of the 'error' field MUST be set to 5 ("Group deleted").

When being informed about the deletion of the OSCORE group, a group member deletes the OSCORE Security Context that it stores as associated with that group, and possibly deallocates any dedicated control resource intended for the Group Manager that it has for that group.

7. ACE Groupcomm Error Identifiers

In addition to what is defined in [Section 9](#) of [[I-D.ietf-ace-key-groupcomm](#)], this document defines a new value that the Group Manager can include as error identifiers, in the 'error' field of an error response with Content-Format application/ace-groupcomm+cbor.

+-----+-----+ Value Description +-----+-----+	
10	Group currently active
+-----+-----+	

Figure 4: ACE Groupcomm Error Identifiers

A Client supporting the 'error' parameter (see [Sections 4.1.2](#) and [8](#) of [[I-D.ietf-ace-key-groupcomm](#)]) and able to understand the specified error may use that information to determine what actions to take next. If it is included in the error response and supported by the Client, the 'error_description' parameter may provide additional context. In particular, the following guidelines apply.

*In case of error 10, the Client should stop sending the request in question to the Group Manager, until the group becomes inactive. As per this document, this error is relevant only for the Administrator, if it tries to delete a group without having set its status to inactive first (see [Section 6.8](#)). In such a case, the Administrator should take the expected course of

actions, and set the group status to inactive first (see [Section 6.6](#) and [Section 6.7](#)), before proceeding with the group deletion.

8. Security Considerations

Security considerations are inherited from the ACE framework for Authentication and Authorization [[I-D.ietf-ace-oauth-authz](#)], and from the specific transport profile of ACE used between the Administrator and the Group Manager, such as [[I-D.ietf-ace-dtls-authorize](#)] and [[I-D.ietf-ace-oscore-profile](#)].

9. IANA Considerations

RFC Editor: Please replace "[[this document]]" with the RFC number of this document and delete this paragraph.

This document has the following actions for IANA.

9.1. ACE Groupcomm Parameters

IANA is asked to register the following entries in the "ACE Groupcomm Parameters" registry defined in [Section 11.7](#) of [[I-D.ietf-ace-key-groupcomm](#)].

Name	CBOR Key	CBOR Type	Reference
hkdf	TBD	tstr / int	[[this document]]
cred_fmt	TBD	int	[[this document]]
group_mode	TBD	simple value	[[this document]]
sign_enc_alg	TBD	tstr / int / simple value	[[this document]]
sign_alg	TBD	tstr / int / simple value	[[this document]]
sign_params	TBD	array / simple value	[[this document]]
pairwise_mode	TBD	simple value	[[this document]]
alg	TBD	tstr / int / simple value	[[this document]]
ecdh_alg	TBD	tstr / int / simple value	[[this document]]
ecdh_params	TBD	array / simple value	[[this document]]
det_req	TBD	simple value	[[this document]]
det_hash_alg	TBD	tstr / int	[[this document]]
active	TBD	simple value	[[this document]]
group_name	TBD	tstr	[[this document]]
group_title	TBD	tstr / simple value	[[this document]]
app_groups	TBD	array	[[this document]]
joining_uri	TBD	tstr	[[this document]]
max_stale_sets	TBD	uint	[[this document]]
as_uri	TBD	tstr	[[this document]]
conf_filter	TBD	array	[[this document]]

app_groups_diff	TBD	array	[[this document]]
-----------------	-----	-------	-------------------

Figure 5: ACE Groupcomm Parameters

9.2. ACE Groupcomm Errors

IANA is asked to register the following entry in the "ACE Groupcomm Errors" registry defined in [Section 11.13](#) of [[I-D.ietf-ace-key-groupcomm](#)].

*Value: 10

*Description: Group currently active.

*Reference: [[This document]]

9.3. Resource Types

IANA is asked to enter the following values in the "Resource Type (rt=) Link Target Attribute Values" registry within the "Constrained Restful Environments (CoRE) Parameters" registry group.

Value	Description	Reference
core.osc.gcoll	Group-collection resource of an OSCORE Group Manager	[[this document]]
core.osc.gconf	Group-configuration resource of an OSCORE Group Manager	[[this document]]

9.4. Group OSCORE Admin Permissions

This document establishes the IANA "Group OSCORE Admin Permissions" registry. The registry has been created to use the "Expert Review" registration procedure [[RFC8126](#)]. Expert review guidelines are provided in [Section 9.8](#).

This registry includes the possible permissions that Administrators can have to perform operations on an OSCORE Group Manager, each in combination with a numeric identifier. These numeric identifiers are used to express authorization information about performing administrative operations concerning OSCORE groups under the control of the Group Manager, as specified in [Section 3](#) of [[this document]].

The columns of this registry are:

*Name: A value that can be used in documents for easier comprehension, to identify a possible permission that

Administrators can perform when interacting with an OSCORE Group Manager.

*Value: The numeric identifier for this permission. Integer values greater than 65535 are marked as "Private Use", all other values use the registration policy "Expert Review" [[RFC8126](#)].

Note that, in general, a single permission can be associated with multiple different operations that are possible to be performed when interacting with the Group Manager.

*Description: This field contains a brief description of the permission.

*Reference: This contains a pointer to the public specification for the permission.

This registry will be initially populated by the values in [Figure 2](#).

The Reference column for all of these entries will be [[this document]].

9.5. AIF

For the media-types application/aif+cbor and application/aif+json defined in [Section 5.1](#) of [[I-D.ietf-ace-aif](#)], IANA is requested to register the following entries for the two media-type parameters Toid and Tperm, in the respective sub-registry defined in [Section 5.2](#) of [[I-D.ietf-ace-aif](#)] within the "MIME Media Type Sub-Parameter" registry group.

*Name: oscore-group-name-pattern

*Description/Specification: wildcard pattern of OSCORE group names

*Reference: [[This document]]

*Name: oscore-group-admin-permissions

*Description/Specification: permission(s) to perform administrative operations at the OSCORE Group Manager

*Reference: [[This document]]

9.6. CoAP Content-Format

IANA is asked to register the following entries to the "CoAP Content-Formats" registry within the "Constrained RESTful Environments (CoRE) Parameters" registry group.

*Media Type: application/aif+cbor;Toid="oscore-group-name-pattern",Tperm="oscore-group-admin-permissions"

*Encoding: -

*ID: TBD

*Reference: [[This document]]

*Media Type: application/aif+json;Toid="oscore-group-name-pattern",Tperm="oscore-group-admin-permissions"

*Encoding: -

*ID: TBD

*Reference: [[This document]]

9.7. ACE Scope Semantics

IANA is asked to register the following entry in the "ACE Scope Semantics" registry defined in [Section 11.12](#) of [[I-D.ietf-ace-key-groupcomm](#)].

*Value: SEM_ID_TBD

*Description: Permissions to perform administrative operations at the ACE Group Manager for Group OSCORE.

*Reference: [[This document]]

9.8. Expert Review Instructions

The IANA registry established in this document is defined as "Expert Review". This section gives some general guidelines for what the experts should be looking for, but they are being designated as experts for a reason so they should be given substantial latitude.

Expert reviewers should take into consideration the following points:

*Clarity and correctness of registrations. Experts are expected to check the clarity of purpose and use of the requested entries. Experts should inspect the entry for the considered permission,

to verify the correctness of its description against the permission as intended in the specification that defined it. Expert should consider requesting an opinion on the correctness of registered parameters from the Authentication and Authorization for Constrained Environments (ACE) Working Group and the Constrained RESTful Environments (CoRE) Working Group.

Entries that do not meet these objective of clarity and completeness should not be registered.

*Duplicated registration and point squatting should be discouraged. Reviewers are encouraged to get sufficient information for registration requests to ensure that the usage is not going to duplicate one that is already registered and that the point is likely to be used in deployments.

*Experts should take into account the expected usage of permissions when approving point assignment. Given a 'Value' V as code point, the length of the encoding of $(2^{(V+1)} - 1)$ should be weighed against the usage of the entry, considering the resources and capabilities of devices it will be used on. Additionally, given a 'Value' V as code point, the length of the encoding of $(2^{(V+1)} - 1)$ should be weighed against how many code points resulting in that encoding length are left, and the resources and capabilities of devices it will be used on.

*Specifications are recommended. When specifications are not provided, the description provided needs to have sufficient information to verify the points above.

10. References

10.1. Normative References

[COSE.Algorithms] IANA, "COSE Algorithms", <<https://www.iana.org/assignments/cose/cose.xhtml#algorithms>>.

[I-D.ietf-ace-aif] Bormann, C., "An Authorization Information Format (AIF) for ACE", Work in Progress, Internet-Draft, draft-ietf-ace-aif-06, 4 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-ace-aif-06.txt>>.

[I-D.ietf-ace-key-groupcomm] Palombini, F. and M. Tiloca, "Key Provisioning for Group Communication using ACE", Work in Progress, Internet-Draft, draft-ietf-ace-key-groupcomm-15, 23 December 2021, <<https://www.ietf.org/archive/id/draft-ietf-ace-key-groupcomm-15.txt>>.

[I-D.ietf-ace-key-groupcomm-oscore] Tiloca, M., Park, J., and F. Palombini, "Key Management for OSCORE Groups in ACE",

Work in Progress, Internet-Draft, draft-ietf-ace-key-groupcomm-oscore-13, 7 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-ace-key-groupcomm-oscore-13.txt>>.

[I-D.ietf-ace-oauth-authz] Seitz, L., Selander, G., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "Authentication and Authorization for Constrained Environments (ACE) using the OAuth 2.0 Framework (ACE-OAuth)", Work in Progress, Internet-Draft, draft-ietf-ace-oauth-authz-46, 8 November 2021, <<https://www.ietf.org/archive/id/draft-ietf-ace-oauth-authz-46.txt>>.

[I-D.ietf-ace-oscore-profile] Palombini, F., Seitz, L., Selander, G., and M. Gunnarsson, "OSCORE Profile of the Authentication and Authorization for Constrained Environments Framework", Work in Progress, Internet-Draft, draft-ietf-ace-oscore-profile-19, 6 May 2021, <<https://www.ietf.org/archive/id/draft-ietf-ace-oscore-profile-19.txt>>.

[I-D.ietf-core-coral] Amsüss, C. and T. Fossati, "The Constrained RESTful Application Language (CoRAL)", Work in Progress, Internet-Draft, draft-ietf-core-coral-04, 25 October 2021, <<https://www.ietf.org/archive/id/draft-ietf-core-coral-04.txt>>.

[I-D.ietf-core-groupcomm-bis] Dijk, E., Wang, C., and M. Tiloca, "Group Communication for the Constrained Application Protocol (CoAP)", Work in Progress, Internet-Draft, draft-ietf-core-groupcomm-bis-06, 7 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-core-groupcomm-bis-06.txt>>.

[I-D.ietf-core-oscore-groupcomm]
Tiloca, M., Selander, G., Palombini, F., Mattsson, J. P., and J. Park, "Group OSCORE - Secure Group Communication for CoAP", Work in Progress, Internet-Draft, draft-ietf-core-oscore-groupcomm-14, 7 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-core-oscore-groupcomm-14.txt>>.

[I-D.ietf-cose-rfc8152bis-algs]
Schaad, J., "CBOR Object Signing and Encryption (COSE): Initial Algorithms", Work in Progress, Internet-Draft, draft-ietf-cose-rfc8152bis-algs-12, 24 September 2020,

<<https://www.ietf.org/archive/id/draft-ietf-cose-rfc8152bis-alsg-12.txt>>.

[I-D.ietf-cose-rfc8152bis-struct]

Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", Work in Progress, Internet-Draft, draft-ietf-cose-rfc8152bis-struct-15, 1 February 2021, <<https://www.ietf.org/archive/id/draft-ietf-cose-rfc8152bis-struct-15.txt>>.

[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>>.

[RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", RFC 6690, DOI 10.17487/RFC6690, August 2012, <<https://www.rfc-editor.org/info/rfc6690>>.

[RFC6749] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework", RFC 6749, DOI 10.17487/RFC6749, October 2012, <<https://www.rfc-editor.org/info/rfc6749>>.

[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>>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.

[RFC8132] van der Stok, P., Bormann, C., and A. Sehgal, "PATCH and FETCH Methods for the Constrained Application Protocol (CoAP)", RFC 8132, DOI 10.17487/RFC8132, April 2017, <<https://www.rfc-editor.org/info/rfc8132>>.

[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>>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and

JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <<https://www.rfc-editor.org/info/rfc8610>>.

[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>>.

[RFC8742] Bormann, C., "Concise Binary Object Representation (CBOR) Sequences", RFC 8742, DOI 10.17487/RFC8742, February 2020, <<https://www.rfc-editor.org/info/rfc8742>>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <<https://www.rfc-editor.org/info/rfc8949>>.

10.2. Informative References

[I-D.amsuess-core-cachable-oscore] Amsüss, C. and M. Tiloca, "Cacheable OSCORE", Work in Progress, Internet-Draft, draft-amsuess-core-cachable-oscore-04, 6 March 2022, <<https://www.ietf.org/archive/id/draft-amsuess-core-cachable-oscore-04.txt>>.

[I-D.hartke-t2trg-coral-reef] Hartke, K., "Resource Discovery in Constrained RESTful Environments (CoRE) using the Constrained RESTful Application Language (CoRAL)", Work in Progress, Internet-Draft, draft-hartke-t2trg-coral-reef-04, 9 May 2020, <<https://www.ietf.org/archive/id/draft-hartke-t2trg-coral-reef-04.txt>>.

[I-D.ietf-ace-dtls-authorize] Gerdes, S., Bergmann, O., Bormann, C., Selander, G., and L. Seitz, "Datagram Transport Layer Security (DTLS) Profile for Authentication and Authorization for Constrained Environments (ACE)", Work in Progress, Internet-Draft, draft-ietf-ace-dtls-authorize-18, 4 June 2021, <<https://www.ietf.org/archive/id/draft-ietf-ace-dtls-authorize-18.txt>>.

[I-D.ietf-core-resource-directory] Amsüss, C., Shelby, Z., Koster, M., Bormann, C., and P. V. D. Stok, "CoRE Resource Directory", Work in Progress, Internet-Draft, draft-ietf-core-resource-directory-28, 7 March 2021, <<https://>

www.ietf.org/archive/id/draft-ietf-core-resource-directory-28.txt>.

[I-D.ietf-cose-cbor-encoded-cert]

Mattsson, J. P., Selander, G., Raza, S., Höglund, J., and M. Furuheid, "CBOR Encoded X.509 Certificates (C509 Certificates)", Work in Progress, Internet-Draft, draft-ietf-cose-cbor-encoded-cert-03, 10 January 2022, <<https://www.ietf.org/archive/id/draft-ietf-cose-cbor-encoded-cert-03.txt>>.

[I-D.ietf-tls-dtls13] Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", Work in Progress, Internet-Draft, draft-ietf-tls-dtls13-43, 30 April 2021, <<https://www.ietf.org/internet-drafts/draft-ietf-tls-dtls13-43.txt>>.

[I-D.tiloca-core-oscore-discovery] Tiloca, M., Amsuess, C., and P. V. D. Stok, "Discovery of OSCORE Groups with the CoRE Resource Directory", Work in Progress, Internet-Draft, draft-tiloca-core-oscore-discovery-11, 7 March 2022, <<https://www.ietf.org/archive/id/draft-tiloca-core-oscore-discovery-11.txt>>.

[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>>.

[RFC7925] Tschofenig, H., Ed. and T. Fossati, "Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things", RFC 7925, DOI 10.17487/RFC7925, July 2016, <<https://www.rfc-editor.org/info/rfc7925>>.

[RFC8392] Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "CBOR Web Token (CWT)", RFC 8392, DOI 10.17487/RFC8392, May 2018, <<https://www.rfc-editor.org/info/rfc8392>>.

Appendix A. Document Updates

RFC EDITOR: PLEASE REMOVE THIS SECTION.

A.1. Version -04 to -05

*Defined format of scope based on a new AIF data model.

*Specified authorization checks at the Group Manager.

*Revised resource handlers based on the new scope format.

- *Renamed 'pub_key_enc' to 'cred_fmt'.
- *Mandatory to include 'group_name' in the group creation request.
- *Suggesting a used 'group_name' results in a new name, not in an error.
- *Distinction between authentication credentials and public keys.
- *More details on informing group members about changes in the group configuration.
- *Revised order of sections; editorial improvements.

A.2. Version -03 to -04

- *Clarifications on what to do in case of enhanced error responses.
- *Clarifications on handling default values for group parameters.
- *New configuration parameters to support OSCORE deterministic requests.
- *IANA considerations - Use RFC8126 terminology.
- *Author's change of address.
- *Editorial improvements.

A.3. Version -02 to -03

- *Aligned new and old parameters to core-groupcomm-oscore and ace-key-groupcomm-oscore.
- *Removed 'cs_key_params' and 'ecdh_key_params' to avoid redundant COSE capabilities of key types, consistently with draft-ietf-ace-key-groupcomm-oscore.
- *Revised examples and side effects due to parameter changes.
- *New error type "Group currently active".

A.4. Version -01 to -02

- *Admit multiple Administrators and limited access to admin resources.
- *Early design considerations for defining the format of scope.
- *Additional error handling, using also error types.

*Selective update of group-configuration resources with PATCH/iPATCH.

*Editorial improvements.

A.5. Version -00 to -01

*Names of application groups as status parameter.

*Parameters related to the pairwise mode of Group OSCORE.

*Defined FETCH for group-configuration resources.

*Policies on registration of links to the Resource Directory.

*Added resource type for group-configuration resources.

*Fixes, clarifications and editorial improvements.

Acknowledgments

Klaus Hartke provided substantial contribution in defining the resource model based on group collection and group configurations, as well as the interactions with the Group Manager using CoRAL.

The authors sincerely thank Christian Amsuess, Carsten Bormann and Jim Schaad for their comments and feedback.

The work on this document has been partly supported by VINNOVA and the Celtic-Next project CRITISEC; and by the H2020 project SIFIS-Home (Grant agreement 952652).

Authors' Addresses

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

Email: marco.tiloca@ri.se

Rikard Höglund
RISE AB
Isafjordsgatan 22
SE-16440 Stockholm Kista
Sweden

Email: rikard.hoglund@ri.se

Peter van der Stok
Consultant

Phone: [+31-492474673 \(Netherlands\)](tel:+31-492474673), [+33-966015248 \(France\)](tel:+33-966015248)
Email: stokcons@bbhmail.nl

Francesca Palombini
Ericsson AB
Torshamnsgatan 23
SE-16440 Stockholm Kista
Sweden

Email: francesca.palombini@ericsson.com