

Diameter Maintenance and Extensions (DIME)  
Internet-Draft  
Intended status: Standards Track  
Expires: January 5, 2015

M. Jones  
M. Liebsch  
L. Morand

July 4, 2014

**Diameter Group Signaling**  
**draft-ietf-dime-group-signaling-04.txt**

**Abstract**

In large network deployments, a single Diameter peer can support over a million concurrent Diameter sessions. Recent use cases have revealed the need for Diameter peers to apply the same operation to a large group of Diameter sessions concurrently. The Diameter base protocol commands operate on a single session so these use cases could result in many thousands of command exchanges to enforce the same operation on each session in the group. In order to reduce signaling, it would be desirable to enable bulk operations on all (or part of) the sessions managed by a Diameter peer using a single or a few command exchanges. This document specifies the Diameter protocol extensions to achieve this signaling optimization.

**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 <http://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 January 5, 2015.

**Copyright Notice**

Copyright (c) 2014 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Terminology</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Protocol Overview</a>	<a href="#">4</a>
<a href="#">3.1.</a>	<a href="#">Building and Modifying Session Groups</a>	<a href="#">4</a>
<a href="#">3.2.</a>	<a href="#">Issuing Group Commands</a>	<a href="#">4</a>
<a href="#">3.3.</a>	<a href="#">Permission Considerations</a>	<a href="#">4</a>
<a href="#">4.</a>	<a href="#">Protocol Description</a>	<a href="#">6</a>
<a href="#">4.1.</a>	<a href="#">Session Grouping</a>	<a href="#">6</a>
<a href="#">4.1.1.</a>	<a href="#">Group assignment at session initiation</a>	<a href="#">7</a>
<a href="#">4.1.2.</a>	<a href="#">Removing a session from a session group</a>	<a href="#">8</a>
<a href="#">4.1.3.</a>	<a href="#">Mid-session group assignment modifications</a>	<a href="#">9</a>
<a href="#">4.2.</a>	<a href="#">Session Grouping Capability Discovery</a>	<a href="#">10</a>
<a href="#">4.2.1.</a>	<a href="#">Explicit Capability Discovery</a>	<a href="#">10</a>
<a href="#">4.2.2.</a>	<a href="#">Implicit Capability Discovery</a>	<a href="#">11</a>
<a href="#">4.3.</a>	<a href="#">Deleting a Session Group</a>	<a href="#">11</a>
<a href="#">4.4.</a>	<a href="#">Performing Group Operations</a>	<a href="#">11</a>
<a href="#">4.4.1.</a>	<a href="#">Sending Group Commands</a>	<a href="#">11</a>
<a href="#">4.4.2.</a>	<a href="#">Receiving Group Commands</a>	<a href="#">12</a>
<a href="#">4.4.3.</a>	<a href="#">Error Handling for Group Commands</a>	<a href="#">12</a>
<a href="#">4.4.4.</a>	<a href="#">Single-Session Fallback</a>	<a href="#">13</a>
<a href="#">5.</a>	<a href="#">Operation with Proxies Agents</a>	<a href="#">13</a>
<a href="#">6.</a>	<a href="#">Commands Formatting</a>	<a href="#">13</a>
<a href="#">6.1.</a>	<a href="#">Formatting Example: Group Re-Auth-Request</a>	<a href="#">14</a>
<a href="#">7.</a>	<a href="#">Attribute-Value-Pairs (AVP)</a>	<a href="#">14</a>
<a href="#">7.1.</a>	<a href="#">Session-Group-Info AVP</a>	<a href="#">15</a>
<a href="#">7.2.</a>	<a href="#">Session-Group-Control-Vector AVP</a>	<a href="#">15</a>
<a href="#">7.3.</a>	<a href="#">Session-Group-Id AVP</a>	<a href="#">16</a>
<a href="#">7.4.</a>	<a href="#">Group-Response-Action AVP</a>	<a href="#">16</a>
<a href="#">7.5.</a>	<a href="#">Session-Group-Capability-Vector AVP</a>	<a href="#">17</a>
<a href="#">8.</a>	<a href="#">Result-Code AVP Values</a>	<a href="#">17</a>
<a href="#">9.</a>	<a href="#">IANA Considerations</a>	<a href="#">17</a>
<a href="#">9.1.</a>	<a href="#">AVP Codes</a>	<a href="#">17</a>
<a href="#">10.</a>	<a href="#">Security Considerations</a>	<a href="#">17</a>
<a href="#">11.</a>	<a href="#">Acknowledgments</a>	<a href="#">18</a>
<a href="#">12.</a>	<a href="#">Normative References</a>	<a href="#">18</a>
<a href="#">Appendix A.</a>	<a href="#">Session Management -- Exemplary Session State</a>	



Machines . . . . .	<a href="#">18</a>
<a href="#">A.1.</a> Authorization Session State Machine . . . . .	<a href="#">18</a>
Authors' Addresses . . . . .	<a href="#">22</a>

## [1.](#) Introduction

In large network deployments, a single Diameter peer can support over a million concurrent Diameter sessions. Recent use cases have revealed the need for Diameter peers to apply the same operation to a large group of Diameter sessions concurrently. For example, a policy decision point may need to modify the authorized quality of service for all active users having the same type of subscription. The Diameter base protocol commands operate on a single session so these use cases could result in many thousands of command exchanges to enforce the same operation on each session in the group. In order to reduce signaling, it would be desirable to enable bulk operations on all (or part of) the sessions managed by a Diameter peer using a single or a few command exchanges.

This document describes mechanisms for grouping Diameter sessions and applying Diameter commands, such as performing re-authentication, re-authorization, termination and abortion of sessions to a group of sessions. This document does not define a new Diameter application. Instead it defines mechanisms, commands and AVPs that may be used by any Diameter application that requires management of groups of sessions.

These mechanisms take the following design goals and features into account:

- o Minimal impact to existing applications
- o Extension of existing commands' Command Code Format (CCF) with optional AVPs to enable grouping and group operations
- o Fallback to single session operation
- o Implicit discovery of capability to support grouping and group operations in case no external mechanism is available to discover a Diameter peer's capability to support session grouping and session group operations.

## [2.](#) Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].



This document uses terminology defined [[RFC6733](#)].

### **3. Protocol Overview**

#### **3.1. Building and Modifying Session Groups**

Client and Server can assign a new Diameter session to a group, e.g. in case the subscription profile of the associated user has similar characteristics as the profile of other users whose Diameter session has been assigned to one or multiple groups. A single command can be issued and applied to all sessions associated with such group(s), e.g. to adjust common profile or policy settings.

The assignment of a Diameter session to a group can be changed mid-session. For example, if a user's subscription profile changes mid-session, a Diameter peer may remove the session from its current group and assign the session to a different group that is more appropriate for the new subscription profile.

In case of mobile users, the user's session may get transferred to a new Diameter client during handover and assigned to a different group, which is maintained at the new Diameter client, mid-session.

A session group, which has sessions assigned, can be deleted, e.g. due to a change in multiple users' subscription profile so that the group's assigned sessions do not share certain characteristics anymore. Deletion of such group requires subsequent individual treatment of each of the assigned sessions. A peer may decide to assign some of these sessions to any other existing or new group.

#### **3.2. Issuing Group Commands**

Changes in the network condition may result in the Diameter server's decision to close all sessions in a given group. The server issues a single Session Termination Request (STR) command, identifying the group of sessions which are to be terminated. The Diameter client treats the STR as group command and initiates termination of all sessions associated with the identified group. Subsequently, the client confirms successful termination of these sessions to the server by sending a single Session Termination Answer (STA) command, which includes the identifier of the group.

#### **3.3. Permission Considerations**

Permission considerations in the context of this draft apply to the permission of Diameter nodes to build new session groups, to assign/remove a session to/from a session group and to delete an existing session group.



This specification follows the most flexible model where both, a Diameter client and a Diameter server can create a new group and assign a new identifier to that session group. When a Diameter node decides to create a new session group, e.g. to group all sessions which share certain characteristics, the node builds a session group identifier according to the rules described in [Section 7.3](#)) and becomes the owner of the group. This specification does not constrain the permission to add or remove a session to/from a session group to the group owner, instead each peer can add a session to any known group or remove a session from a group. A session group is deleted and its identifier released after the last session has been removed from the session group. Also the modification of groups in terms of moving a session from one session group to a different session group is permitted to any Diameter node. A Diameter peer can delete a session group and its group identifier mid-session, resulting in individual treatment of the sessions which have been previously assigned to the deleted group.

The enforcement of more constrained permissions is left to the specification of a particular group signaling enabled Diameter application and compliant implementations of such application must enforce the associated permission model. Details about enforcing a more constraint permission model are out of scope of this specification. For example, a more constrained model could require that a client MUST NOT remove a session from a group which is owned by the server.

The following table depicts the permission considerations as per the present specification:





Operation	Server	Client
Create a new Session Group (peer becomes the group owner)	X	X
Assign a Session to an owned Session Group	X	X
Assign a Session to a non-owned Session Group	X	X
Remove a Session from an owned Session Group	X	X
Remove a Session from a non-owned Session Group	X	X
Remove a Session from a Session Group where the peer created the assignment	X	X
Remove a Session from a Session Group where the peer did not create the assignment		
Overrule a peer's group assignment *)		
Delete a Session Group owned by the peer	X	X
Delete a Session Group not owned by the peer		

Default Permission as per this Specification

\*) Editors' note: The protocol specification in this document does not consider overruling a peer's assignment of a session to a session group. Group signaling enabled applications may take such protocol support and associated protocol semantics into account in their specification.

## 4. Protocol Description

### 4.1. Session Grouping

Either Diameter peer can initiate the assignment of a session to a single or multiple session groups. Modification of a group by removing or adding a single or multiple user sessions can be initiated and performed mid-session by either Diameter peer. Diameter AAA applications typically assign client and server roles to the Diameter peers, which are referred to as relevant Diameter peers to utilize session grouping and issue group commands. [Section 5](#)



describes particularities about session grouping and performing group commands when relay agents or proxies are deployed.

Diameter peers, which are group-aware, must store and maintain an entry about the group assignment together with a session's state. A list of all known session groups should be locally maintained on each peer, each group pointing to individual sessions being assigned to the group. A peer must also keep a record about sessions, which have been assigned to a session group by that peer.

#### **4.1.1. Group assignment at session initiation**

To assign a session to a group at session initiation, a Diameter client sends a service specific request, e.g. NASREQ AAR [[RFC4005](#)], containing one or more group identifiers. Each of these groups need to be identified by a unique Session-Group-Id contained in a separate Session-Group-Info AVP as specified in [Section 7](#).

The client may choose one or multiple sessions from a list of existing session groups. Alternatively, the client may decide to create a new group and identify itself in the DiameterIdentity element of the Group-Session-Id AVP as per [Section 7.3](#)

The client MUST set the SESSION\_GROUP\_ALLOCATION\_ACTION of the Session-Group-Control-Vector AVP in each appended Session-Group-Info AVP to indicate that the identified session should be assigned to the identified session group.

If the Diameter server receives a command request from a Diameter client and the command comprises at least one Session-Group-Info AVP having the SESSION\_GROUP\_ALLOCATION\_ACTION flag of the Session-Group-Control-Vector AVP set, the server must assign the new session to each of the one or multiple identified session groups. In case one or multiple identified session groups are not known to the server, the server must add the one or multiple new groups to its local list of known session groups. When sending the response to the client, e.g. a service-specific auth response as per NASREQ AAA [[RFC4005](#)], the server must include all Session-Group-Info AVPs as received in the client's request.

In addition to the one or multiple session groups identified in the client's request, the server may decide to assign the new session to one or multiple additional groups. In such case, the server adds to the response additional Session-Group-Info AVPs, each identifying a session group, to which the server has assigned the new session. Each of the Session-Group-Info AVP added by the server must have the SESSION\_GROUP\_ALLOCATION\_ACTION flag of the Session-Group-Control-Vector AVP set.



If the Diameter client receives a response to its previously issued request from the server and the response comprises at least one Session-Group-Info AVP having the SESSION\_GROUP\_ALLOCATION\_ACTION flag of the associated Session-Group-Control-Vector AVP set, the client must add the new session to all session groups as identified in the one or multiple Session-Group-Info AVPs.

A Diameter server receiving a command for session initiation which includes at least one Session-Group-Info AVP but the server does not understand the semantics of this optional AVP because it does not support group operations according to the specification in this document, MUST ignore the optional group operations specific AVPs and proceed with processing the command for a single session.

A Diameter client, which sent a request for session initiation to a Diameter server and appended a single or multiple Session-Group-Id AVPs but cannot find any Session-Group-Info AVP in the associated response from the Diameter server proceeds with processing the command for a single session. Furthermore, the client keeps a log to remember that the server is not able to perform group operations.

#### **4.1.2. Removing a session from a session group**

When a Diameter client decides to remove a session from a particular session group, the client sends a service-specific re-authorization request to the server and adds one Session-Group-Info AVP to the request for each session group, from which the client wants to remove the session. The session, which is to be removed from a group, is identified in the Session-Id AVP of the command request. The SESSION\_GROUP\_ALLOCATION\_ACTION flag of the Session-Group-Control-Vector AVP in each Session-Group-Info AVP must be cleared to indicate removal of the session from the session group identified in the associated Session-Group-id AVP.

When a Diameter client decides to remove a session from all session groups, to which the session has been previously assigned, the client sends a service-specific re-authorization request to the server and adds a single Session-Group-Info AVP to the request which has the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and the Session-Group-Id AVP omitted. The session, which is to be removed from all groups, to which the session has been previously assigned, is identified in the Session-Id AVP of the command request.

If the Diameter server receives a request from the client which has at least one Session-Group-Info AVP appended with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared, the server must remove the session from the session group identified in the associated Session-Group-Id AVP. If the request comprises at least one Session-



Group-info AVP with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and no Session-Id AVP present, the server must remove the session from all session groups to which the session has been previously assigned. The server must include in its response to the requesting client all Session-Group-Id AVPs as received in the request.

When the Diameter server decides to remove a session from one or multiple particular session groups or from all session groups to which the session has been assigned beforehand, the server sends a Re-Authorization Request (RAR) to the client, indicating the session in the requests Session-Id AVP. The client sends a Re-Authorization Answer (RAA) to respond to the server's request. The client subsequently sends service-specific re-authorization request containing one or multiple Session-Group-Info AVPs, each indicating a session group, to which the session had been previously assigned. To indicate removal of the indicated session from one or multiple session groups, the server sends a service-specific auth response to the client, containing a list of Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and the Session-Group-Id AVP identifying the session group, from which the session should be removed. The server MAY include to the service-specific auth response a list of Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag set and the Session-Group-Id AVP identifying session groups to which the session remains subscribed. In case the server decides to remove the identified session from all session groups, to which the session has been previously assigned, the server includes in the service-specific auth response at least one Session-Group-Info AVP with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and Session-Group-Id AVP absent.

#### **4.1.3. Mid-session group assignment modifications**

Either Diameter peer can modify the group membership of an active Diameter session according to the specified permission considerations.

To update an assigned group mid-session, a Diameter client sends a service-specific re-authorization request to the server, containing one or multiple Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag set and the Session-Group-Id AVP present, identifying the session group to which the session should be assigned. With the same message, the client may send one or multiple Session-Group-Info AVP with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and the Session-Group-Id AVP identifying the session group from which the identified session is to be removed. To remove the session from all previously assigned session groups, the client includes at least one Session-Group-Info AVP with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and no Session-Group-Id





AVP present. When the server received the service-specific re-authorization request, it must update its locally maintained view of the session groups for the identified session according to the appended Session-Group-Info AVPs. The server sends a service-specific auth response to the client containing one or multiple Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag set and the Session-Group-Id AVP identifying the new session group to which the identified session has been assigned.

When a Diameter server enforces an update to the assigned groups mid-session, it sends a Re-Authorization Request (RAR) message to the client identifying the session, for which the session group lists are to be updated. The client responds with a Re-Authorization Answer (RAA) message. The client subsequently sends service-specific re-authorization request containing one or multiple Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag set and the Session-Group-Id AVP identifying the session group to which the session had been previously assigned. The server responds with a service-specific auth response and includes one or multiple Session-Group-Info AVP with the SESSION\_GROUP\_ALLOCATION\_ACTION flag set and the Session-Group-Id AVP identifying the session group, to which the identified session is to be assigned. With the same response message, the server may send one or multiple Session-Group-Info AVPs with the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and the Session-Group-Id AVP identifying the session groups from which the identified session is to be removed. When server wants to remove the session from all previously assigned session groups, it send at least on Session-Group-Info AVP with the response having the SESSION\_GROUP\_ALLOCATION\_ACTION flag cleared and no Session-Group-Id AVP present.

#### **4.2. Session Grouping Capability Discovery**

Diameter nodes should assign a session to a session group and perform session group operations with a peer only after having ensured that the peer announced associated support beforehand.

##### **4.2.1. Explicit Capability Discovery**

New Diameter applications may consider support for Diameter session grouping and for performing group commands during the standardization process. Such applications provide intrinsic discovery for the support of group commands and announce this capability through the assigned application ID.

System- and deployment-specific means for capability exchange can be used to announce peers' support for session grouping and session group operations. In such case, the optional Session-Group-



Capability-Vector AVP, as described in [Section 4.2.2](#) can be omitted in Diameter messages being exchanged between peers.

#### **[4.2.2](#). Implicit Capability Discovery**

If no explicit mechanism for capability discovery is deployed to enable Diameter nodes to learn about peers' capability to support session grouping and group commands, Diameter peers SHOULD append the Session-Group-Capability-Vector AVP to any Diameter messages exchanged with its peers to announce its capability to support session grouping and session group operations. Implementations following this specification set the BASE\_SESSION\_GROUP\_CAPABILITY flag of the Session-Group-Capability-Vector AVP.

When a Diameter node receives at least one Session-Group-Capability-Vector AVP from a peer with the BASE\_SESSION\_GROUP\_CAPABILITY flag set, the Diameter node maintains a log to remember the peer's capability to support group commands.

#### **[4.3](#). Deleting a Session Group**

To delete a session group and release the associated Session-Group-Id value, the owner of a session group appends a single Session-Group-Info AVP having the SESSION\_GROUP\_STATUS\_IND flag cleared and the Session-Group-Id AVP identifying the session group, which is to be deleted. The SESSION\_GROUP\_ALLOCATION\_ACTION flag of the associated Session-Group-Control-Vector AVP MUST be cleared.

#### **[4.4](#). Performing Group Operations**

##### **[4.4.1](#). Sending Group Commands**

Either Diameter peer can request the recipient of a request to process an associated command for all sessions being assigned to one or multiple groups by identifying these groups in the request. The sender of the request appends for each group, to which the command applies, a Session-Group-Info AVP including the Session-Group-Id AVP to identify the associated session group. Both, the SESSION\_GROUP\_ALLOCATION\_ACTION flag as well as the SESSION\_GROUP\_STATUS\_IND flag must be set.

If the CCF of the request mandates a Session-Id AVP, the Session-Id AVP MUST identify a single session which is assigned to at least one of the groups being identified in the appended Session-Group-Id AVPs.

The sender of the request MUST indicate to the receiver how follow up message exchanges should be performed by appending a single instance of the Group-Response-Action AVP. Even if the request includes



multiple instances of a Session-Group-Info AVP, the request MUST NOT comprise more than a single instance of a Group-Response-Action AVP. If the sender wants the receiver to perform follow up exchanges with a single command for all impacted groups, the sender sets the value of the Group-Response-Action AVP to ALL\_GROUPS (1). If follow up message exchanges should be performed on a per-group basis in case multiple groups are identified in the group command, the value of the Group-Response-Action AVP is set to PER\_GROUP (2). A value set to PER\_SESSION (3) indicates to the receiver that all follow up exchanges should be performed using a single message for each impacted session.

If the sender wants the receiver of the request to process the associated command solely for a single session does not append any group identifier, but identifies the relevant session in the Session-Id AVP.

#### **4.4.2. Receiving Group Commands**

A Diameter peer receiving a request to process a command for a group of sessions identifies the relevant groups according to the appended Session-Group-Id AVP in the Session-Group-Info AVP and processes the group command according to the appended Group-Response-Action AVP . If the received request identifies multiple groups in multiple appended Session-Group-Id AVPs, the receiver should process the associated command for each of these groups. if a session has been assigned to more than one of the identified groups, the receiver must process the associated command only once per session.

The Diameter peer receiving a request which requests performing the command to at least on session group SHOULD perform follow up message exchanges according to the value identified in the Session-Group-Info AVP.

#### **4.4.3. Error Handling for Group Commands**

When a Diameter peer receives a request to process a command for one or more session groups and the result of processing the command is an error that applies to all sessions in the identified groups, an associated protocol error must be returned to the source of the request. In such case, the sender of the request MUST fall back to single-session processing and the session groups, which have been identified in the group command, MUST be deleted according to the procedure described in [Section 4.3](#).

When a Diameter peer receives a request to process a command for one or more session groups and the result of processing the command succeeds for some sessions identified in one or multiple session



groups, but fails for one or more sessions, the Result-Code AVP in the response message SHOULD indicate `DIAMETER_LIMITED_SUCCESS` as per [Section 7.1.2 of \[RFC6733\]](#). In case of limited success, the sessions, for which the processing of the group command failed, MUST be identified using a Failed-AVP AVP as per Session 7.5 of [\[RFC6733\]](#).

#### **4.4.4. Single-Session Fallback**

Either Diameter peer, a Diameter client or a Diameter server, can fall back to single session operation by ignoring and omitting the optional group session-specific AVPs. Fallback to single-session operation is performed by processing the Diameter command solely for the session identified in the mandatory Session-Id AVP. The response to the group command must not identify any group but identify solely the single session for which the command has been processed.

### **5. Operation with Proxies Agents**

This specification assumes in case of a present stateful Proxy Agent between a Diameter client and a Diameter server that the Proxy Agent is aware of session groups and session group handling. The Proxy MUST reflect the state of each session associated with a session group according to the result of a group command operated between a Diameter client and a server.

In case a Proxy Agent manipulates session groups, it MUST maintain consistency of session groups between a client and a server. This applies to deployment where the Proxy Agent utilizes session grouping and performing group commands with, for example, a Diameter server, whereas the Diameter client is not group-aware. The same applies to deployment where all nodes, the Diameter client and server, as well as the Proxy Agent are group-aware but the Proxy Agent manipulates groups, e.g. to adopt different administrative policies that apply to the client's domain and the server's domain.

### **6. Commands Formatting**

This document does not specify new Diameter commands to enable group operations, but relies on command extensibility capability provided by the Diameter Base protocol. This section provides the guidelines to extend the CCF of existing Diameter commands with optional AVPs to enable the recipient of the command to perform the command to all sessions associated with the identified group(s).





### **6.1. Formatting Example: Group Re-Auth-Request**

A request that one or more groups of users are re-authentication is issued by appending one or multiple Session-Group-Id AVP(s) to the Re-Auth-Request (RAR) and a single instance of a Group-Response-Action AVP. The one or multiple Session-Group-Id AVP(s) identify the associated group(s) for which the group re-authentication has been requested. The Group-Response-Action AVP identifies the expected means to perform and respond to the group command. The recipient of the group command initiates re-authentication for all users associated with the identified group(s). Furthermore, the sender of the group re-authentication request appends a Group-Response-Action AVP to provide more information to the receiver of the command about how to accomplish the group operation.

The value of the mandatory Session-Id AVP MUST identify a session associated with a single user, which is assigned to at least one of the groups being identified in the appended Session-Group-Id AVPs.

```
<RAR> ::= < Diameter Header: 258, REQ, PXY >
        < Session-Id >
        { Origin-Host }
        { Origin-Realm }
        { Destination-Realm }
        { Destination-Host }
        { Auth-Application-Id }
        { Re-Auth-Request-Type }
        [ User-Name ]
        [ Origin-State-Id ]
        * [ Proxy-Info ]
        * [ Route-Record ]
        [ Session-Group-Capability-Vector ]
        * [ Session-Group-Info ]
        [ Group-Response-Action ]
        * [ AVP ]
```

## **7. Attribute-Value-Pairs (AVP)**



		+-----+   AVP Flag rules   +-----+-----+-----+-----+			
Attribute Name	AVP	SHOULD   MUST			
	Code Value Type	MUST	MAY	NOT	NOT
+-----+-----+-----+-----+					
Session-Group-Info	TBD1 Grouped		P		V
Session-Group-Control-Vector	TBD2 Unsigned32		P		V
Session-Group-Id	TBD3 OctetString		P		V
Group-Response-Action	TBD4 Unsigned32		P		V
Session-Group-Capability-Vector	TBD5 Unsigned32		P		V
+-----+-----+-----+-----+					

### AVPs for the Diameter Group Signaling

#### [7.1.](#) Session-Group-Info AVP

The Session-Group-Info AVP (AVP Code TBD1) is of type Grouped. It contains the identifier of the session group as well as an indication of the node responsible for session group identifier assignment.

```
Session-Group-Info ::= < AVP Header: TBD1 >
                        < Session-Group-Control-Vector >
                        [ Session-Group-Id ]
                        * [ AVP ]
```

#### [7.2.](#) Session-Group-Control-Vector AVP

The Session-Group-Control-Vector AVP (AVP Code TBD2) is of type Unsigned32 and contains a 32-bit flags field to control the group assignment at session-group aware nodes.

The following capabilities are defined in this document:

SESSION\_GROUP\_ALLOCATION\_ACTION (0x00000001)

This flag indicates the action to be performed for the identified session. When this flag is set, it indicates that the identified Diameter session is to be assigned to the session group as identified by the Session-Group-Id AVP or the session's assignment to the session group identified in the Session-Group-Id AVP is still valid. When the flag is cleared, the identified Diameter session is to be removed from at least one session group. When the flag is cleared and the Session-Group-Info AVP identifies a particular session group in the associated Session-Group-Id AVP, the session is to be removed solely from the identified session group. When the flag is cleared and the Session-Group-Info AVP does not identify a particular session group (Session-Group-Id AVP



is absent), the identified Diameter session is to be removed from all session groups, to which it has been previously assigned.

SESSION\_GROUP\_STATUS\_IND (0x00000010)

This flag indicates the status of the session group identified in the associated Session-Group-Id AVP. The flag is set when the identified session group has just been created or is still active. If the flag is cleared, the identified session group is deleted and the associated Session-Group-Id is released. If the Session-Group-Info AVP does not comprise a Session-Group-Id AVP, this flag is meaningless and MUST be ignored by the receiver.

### **7.3. Session-Group-Id AVP**

The Session-Group-Id AVP (AVP Code TBD3) is of type UTF8String and identifies a group of Diameter sessions.

The Session-Group-Id MUST be globally and eternally unique, as it is meant to uniquely identify a group of Diameter sessions without reference to any other information.

The default format of the Session-Group-id MUST comply to the format recommended for a Session-Id, as defined in the [section 8.8](#) of the [RFC6733]. The DiameterIdentity element of the Session-Group-Id MUST identify the Diameter node, which owns the session group.

### **7.4. Group-Response-Action AVP**

The Group-Response-Action AVP (AVP Code TBD4) is of type Unsigned32 and contains a 32-bit address space representing values indicating how the peer SHOULD issue follow up exchanges in response to a command which impacts multiple sessions. The following values are defined by this application:

ALL\_GROUPS (1)

Follow up exchanges should be performed with a single message exchange for all impacted groups.

PER\_GROUP (2)

Follow up exchanges should be performed with a message exchange for each impacted group.

PER\_SESSION (3)

Follow up exchanges should be performed with a message exchange for each impacted session.



### **7.5. Session-Group-Capability-Vector AVP**

The Session-Group-Capability-Vector AVP (AVP Code TBD5) is of type Unsigned32 and contains a 32-bit flags field to indicate capabilities in the context of session-group assignment and group operations.

The following capabilities are defined in this document:

BASE\_SESSION\_GROUP\_CAPABILITY (0x00000001)

This flag indicates the capability to support session grouping and session group operations according to this specification.

## **8. Result-Code AVP Values**

This document does not define new Result-Code [[RFC6733](#)] values for existing applications, which are extended to support group commands. Specification documents of new applications, which will have intrinsic support for group commands, may specify new Result-Codes.

## **9. IANA Considerations**

This section contains the namespaces that have either been created in this specification or had their values assigned to existing namespaces managed by IANA.

### **9.1. AVP Codes**

This specification requires IANA to register the following new AVPs from the AVP Code namespace defined in [[RFC6733](#)].

- o Session-Group-Info
- o Session-Group-Control-Vector
- o Session-Group-Id
- o Group-Response-Action
- o Session-Group-Capability-Vector

The AVPs are defined in [Section 7](#).

## **10. Security Considerations**

TODO





## **11. Acknowledgments**

The authors of this document want to thank Ben Campbell and Eric McMurphy for their valuable comments to early versions of this draft.

## **12. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC4005] Calhoun, P., Zorn, G., Spence, D., and D. Mitton, "Diameter Network Access Server Application", [RFC 4005](#), August 2005.
- [RFC6733] Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", [RFC 6733](#), October 2012.

## **Appendix A. Session Management -- Exemplary Session State Machines**

### **A.1. Authorization Session State Machine**

[Section 8.1 in \[RFC6733\]](#) defines a set of finite state machines, representing the life cycle of Diameter sessions, and which MUST be observed by all Diameter implementations that make use of the authentication and/or authorization portion of a Diameter application. This section defines the additional state transitions related to the processing of the new commands which may impact multiple sessions.

The group membership is session state and therefore only those state machines from [\[RFC6733\]](#) in which the server is maintaining session state are relevant in this document. As in [\[RFC6733\]](#), the term Service-Specific below refers to a message defined in a Diameter application (e.g., Mobile IPv4, NASREQ).

The following state machine is observed by a client when state is maintained on the server. State transitions which are unmodified from [\[RFC6733\]](#) are not repeated here.

A Diameter group command in the following tables is differentiated from a single-session related command by a preceding 'G'. A Group Re-Auth Request, which applies to one or multiple session groups, has been exemplarily described in [Section 6.1](#). Such Group RAR command is denoted as 'GRAR' in the following table. The same notation applies to other commands as per [\[RFC6733\]](#).

CLIENT, STATEFUL				
State	Event	Action	New State	



Idle	Client or Device Requests access	Send service specific auth req optionally including groups	Pending
Open	GASR received with Group-Response-Action = ALL_GROUPS, session is assigned to received group(s) and client will comply with request to end the session	Send GASA with Result-Code = SUCCESS, Send GSTR.	Discon
Open	GASR received with Group-Response-Action = PER_GROUPS, session is assigned to received group(s) and client will comply with request to end the session	Send GASA with Result-Code = SUCCESS, Send GSTR per group	Discon
Open	GASR received with Group-Response-Action = PER_SESSION, session is assigned to received group(s) and client will comply with request to end the session	Send GASA with Result-Code = SUCCESS, Send STR per session	Discon
Open	GASR received, client will not comply with request to end all session in received group(s)	Send GASA with Result-Code != SUCCESS	Open
Discon	GSTA Received	Discon. user/device	Idle
Open	GRAR received with Group-Response-Action = ALL_GROUPS, session is assigned to received group(s) and client will perform subsequent re-auth	Send GRAA, Send service specific group re-auth req	Pending



Open	GRAR received with Group-Response-Action = PER_GROUP, session is assigned to received group(s) and client will perform subsequent re-auth	Send GRAA, Send service specific group re-auth req per group	Pending
Open	GRAR received with Group-Response-Action = PER_SESSION, session is assigned to received group(s) and client will perform subsequent re-auth	Send GRAA, Send service specific re-auth req per session	Pending
Open	GRAR received and client will not perform subsequent re-auth	Send GRAA with Result-Code != SUCCESS, Discon. user/device	Idle
Pending	Successful service-specific group re-authorization answer received.	Provide service	Open
Pending	Failed service-specific group re-authorization answer received.	Discon. user/device	Idle

The following state machine is observed by a server when it is maintaining state for the session. State transitions which are unmodified from [[RFC6733](#)] are not repeated here.



SERVER, STATEFUL			
State	Event	Action	New State
-----			
Idle	Service-specific authorization request received, and user is authorized	Send successful service specific answer optionally including groups	Open
Open	Server wants to terminate group(s)	Send GASR	Discon
Discon	GASA received	Cleanup	Idle
Any	GSTR received	Send GSTA, Cleanup	Idle
Open	Server wants to reauth group(s)	Send GRAR	Pending
Pending	GRAA received with Result-Code = SUCCESS	Update session(s)	Open
Pending	GRAA received with Result-Code != SUCCESS	Cleanup session(s)	Idle
Open	Service-specific group re-authoization request received and user is authorized	Send successful service specific group re-auth answer	Open
Open	Service-specific group re-authorization request received and user is not authorized	Send failed service specific group re-auth answer, cleanup	Idle





Authors' Addresses

Mark Jones

Email: mark@azu.ca

Marco Liebsch

Email: marco.liebsch@neclab.eu

Lionel Morand

Email: lionel.morand@orange.com