Internet Engineering Task Force (IETF)
Request for Comments: 9390
Category: Standards Track
ISSN: 2070-1721

M. Jones
Individual
M. Liebsch
NEC
L. Morand
Orange

April 2023

Diameter Group Signaling

Abstract

In large network deployments, a single Diameter node can support over a million concurrent Diameter sessions. In some use cases, Diameter nodes need 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 can result in many thousands of command exchanges enforcing the same operation on each session in the group. In order to reduce signaling, it is desirable to enable bulk operations on all (or part of) the sessions managed by a Diameter node 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 is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9390.

Table of Contents

1. Introduction

In large network deployments, a single Diameter node can support over a million concurrent Diameter sessions. In some use cases, Diameter nodes need 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 can result in many thousands of command exchanges enforcing the same operation on each session in the group. In order to reduce signaling, it is desirable to enable bulk operations on all (or part of) the sessions managed by a Diameter node 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 Attribute-Value Pairs (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:

* minimal impact to existing applications

* extension of existing commands' Command Code Format (CCF) with optional AVPs to enable grouping and group operations

* fallback to single-session operation

* 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", "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.

This document uses terminology defined in [RFC6733].

3. Protocol Overview

3.1. Building and Modifying Session Groups

In order to accommodate bulk operations on Diameter sessions, the concept of session groups is introduced. Once sessions are added to a group, a command acting on the group will affect all the member sessions.

The client and the 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 one or more such groups, e.g., to adjust common profile or policy settings.

The assignment of a Diameter session to a group can be changed during an ongoing session (mid-session). For example, if a user's subscription profile changes mid-session, a Diameter server may remove a session from an existing group and assign this session to a different group that is more appropriate for the new subscription profile.

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

It may be required to delete a session group, e.g., at the expiry of a promotional period that applied to multiple subscriber profiles. Deletion of such group requires subsequent individual treatment of each of the assigned sessions. A node 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. For example, the server issues a single Session Termination Request (STR) command, including the identifier of the group of sessions that are to be terminated. The Diameter client treats the STR as a group command and initiates the termination of all sessions associated with the identified group. Subsequently, the client confirms the 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 document 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.

When a client or server decides to create a new session group, e.g., to group all sessions that share certain characteristics, this node builds a session group identifier according to the rules described in Section 7.3 and becomes the owner of the group.

After the creation of a session group, a session can be added to this session group by either the client or the server. However, a session can only be removed from a session group by the Diameter node (client or server) that has assigned this session to the session group.

A session group can only be deleted by the owner of the session group, resulting in individual treatment of the sessions that were assigned to this session group.

Diameter applications with implicit support for session groups MAY define a more constrained permission model. For example, a more constrained model could require that a client not remove a session from a group that is owned by the server. Details about enforcing a more constrained permission model are out of scope of this specification.

4. Protocol Description

4.1. Session Grouping Capability Discovery

Diameter nodes SHOULD NOT perform group operations with peer nodes unless the node has advertised support for session grouping and group operations.

4.1.1. Capability Discovery Based on the Application Id

Newly defined Diameter applications may intrinsically support Diameter session grouping and group operations. In these cases, there is no need of a specific discovery mechanism for the support of session grouping capability besides the discovery of the Application Id assigned to the application advertised during the capability exchange phase described in Section 5.3 of [RFC6733].

System-specific and deployment-specific means, as well as out-of-band mechanisms for capability discovery, can be used to announce nodes' support for session grouping and session group operations. In such case, the optional Session-Group-Capability-Vector AVP, as described in Section 4.1.2, can be omitted in Diameter messages being exchanged between nodes.

4.1.2. Capability Discovery Based on AVP Presence

If no other mechanism for capability discovery is deployed to enable Diameter nodes to learn about nodes' capability to support session grouping and group commands for a given application, a Diameter node SHOULD append the Session-Group-Capability-Vector AVP to any Diameter application messages exchanged with the other Diameter nodes to announce its capability to support session grouping and session group operations for the advertised application. Implementations following the specification as per this document MUST 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 node with the BASE_SESSION_GROUP_CAPABILITY flag set, the receiving Diameter node discovers the supported session grouping capability of the sending Diameter node for the advertised application and MUST cache this information for the lifetime of the

routing table entry associated with the peer identity / Application Id pair (see Section 2.7 of [RFC6733]).

4.2. Session Grouping

This specification does not limit the number of session groups to which a single session is assigned. It is left to the implementation of an application to determine such limitations. If an application facilitates a session to belong to multiple session groups, the application MUST maintain consistency of associated application session states for these multiple session groups.

Either Diameter node (client or server) 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 node responsible for the session assignment to this group. Although Diameter is a peer-to-peer protocol, Diameter Authentication, Authorization, and Accounting (AAA) applications typically assign the role of a "Diameter client" to the Diameter node initiating the Diameter session and the role of "Diameter server" to the node authorizing the Diameter session. This specification does not restrict group creation, assignment, or deletion actions to a specific role. In the following sections, "Diameter node" is used to refer to either role. Section 5 describes particularities about session grouping and performing group commands when relay agents or proxies are deployed.

Any Diameter node that has advertised support of session grouping and group operations MUST store and maintain the group assignment as part of the session’s state. A list of all known session groups is locally maintained on each node, with each group pointing to individual sessions being assigned to the group. Each Diameter node MUST also keep a record about sessions that it has assigned to a session group.

4.2.1. 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., Network Access Server Requirements (NASREQ) AA-Request [RFC7155], containing one or more session group identifiers. Each of these groups MUST 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 session groups from a list of existing session groups. Alternatively, the client may decide to create a new group to which the session is assigned and identify itself in the <DiameterIdentity> portion of the Session-Group-Id AVP, as per Section 7.3. For all assignments of a session to an active session group made by the client or the server, the SESSION_GROUP_STATUS flag in the Session-Group-Info AVP, which identifies the session group, MUST be set. A set SESSION_GROUP_STATUS flag indicates that the identified session group has just been created or is still active.

The client MUST set the SESSION_GROUP_ALLOCATION_ACTION flag of the Session-Group-Control-Vector AVP in each appended Session-Group-Info AVP to indicate that the session contained in the request should be assigned to the identified session group.

The client may also indicate in the request that it supports assignment of the session to one or more groups by the server. In such case, the client MUST include the Session-Group-Info AVP in the request, including the Session-Group-Control-Vector AVP with the SESSION_GROUP_ALLOCATION_ACTION flag set but no Session-Group-Id AVP.

If the Diameter server receives a command request from a Diameter client and the command includes at least one Session-Group-Info AVP with the SESSION_GROUP_ALLOCATION_ACTION flag in the Session-Group-Control-Vector AVP set, the server can accept or reject the request for group assignment. Reasons for rejection may be, e.g., lack of resources for managing additional groups. When rejected, the session MUST NOT be assigned to any session group.

If the Diameter server accepts the client's request for a group assignment, the server MUST assign the new session to each (one or more) of the identified session groups when present in the Session-Group-Info AVP. If one or multiple identified session groups are not already stored by the server, the server MUST store the one or more newly identified groups to its local list of known session groups. When sending the response to the client, e.g., a service-specific authorization response, as per NASREQ AA-Answer [RFC7155], the server MUST include all Session-Group-Info AVPs 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 MUST add to the response the additional Session-Group-Info AVPs, each identifying a session group to which the new session is assigned by the server. Each of the Session-Group-Info AVPs added by the server MUST have the SESSION_GROUP_ALLOCATION_ACTION flag set in the Session-Group-Control-Vector AVP.

If the Diameter server rejects the client's request for a group assignment, the server sends the response to the client, e.g., a service-specific authorization response, as per NASREQ AA-Answer [RFC7155], and MUST include all Session-Group-Info AVPs received in the client's request (if any) while clearing the SESSION_GROUP_ALLOCATION_ACTION flag of the Session-Group-Control-Vector AVP. The server MAY still accept the client's request for the identified session to proceed despite rejecting the group assignment. The response sent to the client will then indicate success in the result code. In this case, the session is treated as a single session without assignment to any session group by the Diameter nodes.

If the assignment of the session to one or some of the multiple identified session groups fails, the session group assignment is treated as a failure. In such case, the session is treated as a single session without assignment to any session group by the Diameter nodes. The server sends the response to the client and MAY include those Session-Group-Info AVPs for which the group assignment failed. The SESSION_GROUP_ALLOCATION_ACTION flag of included Session-Group-Info AVPs MUST be cleared.

If the Diameter server receives a command request from a Diameter client and the command includes a Session-Group-Info AVP that does not include a Session-Group-Id AVP, the server MAY decide to assign the session to one or multiple session groups. For each session group to which the server assigns the new session, the server includes a Session-Group-Info AVP with the Session-Group-Id AVP, identifying a session group in the response sent to the client. Each of the Session-Group-Info AVPs included by the server MUST have the SESSION_GROUP_ALLOCATION_ACTION flag of the Session-Group-Control-Vector AVP set.

If the Diameter server receives a command request from a Diameter client and the command does not contain any Session-Group-Info AVPs, the server MUST NOT assign the new session to any session group but treat the request the same as for a single session. The server MUST

NOT return any Session-Group-Info AVP in the command response.

If the Diameter client receives a response to its previously issued request from the server and the response includes at least one Session-Group-Info AVP with 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 one or multiple Session-Group-Info AVPs. If the Diameter client fails to add the session to one or more session groups as identified in one or multiple Session-Group-Info AVPs, the client MUST terminate the session. The client MAY send a subsequent request for session initiation to the server without requesting the assignment of the session to a session group.

If the Diameter client receives a response to its previously issued request from the server and one or more Session-Group-Info AVPs have the SESSION_GROUP_ALLOCATION_ACTION flag of the associated Session-Group-Control-Vector AVP cleared, the client MUST terminate the assignment of the session to one or multiple groups. If the response from the server indicates success in the result code but only the assignment of the session to a session group has been rejected by the server, the client treats the session as a single session without group assignment.

If a Diameter client sends a request for session initiation containing one or more Session-Group-Info AVPs but the response from the Diameter server does not contain a Session-Group-Info AVP, the Diameter client MUST proceed as if the request was processed without group assignments. The Diameter client MUST NOT retry to request group assignment for this session but MAY try to request group assignment for other new sessions.

4.2.2. 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 that 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 that has the SESSION_GROUP_ALLOCATION_ACTION flag cleared and the Session-Group-Id AVP omitted. The Session-Id AVP in the re-authorization request identifies the session that is to be removed from all groups to which it had been previously assigned.

If the Diameter server receives a request from the client that 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 includes 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 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-Auth-Request (RAR) or a service-specific server-initiated request to the client, indicating the session in the Session-Id AVP of the request. The client sends a Re-Auth-Answer (RAA) or a service-specific answer to respond to the server's request. The client subsequently sends a 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 authorization 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 with the service-specific authorization 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. If 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 authorization response at least one Session-Group-Info AVP with the SESSION_GROUP_ALLOCATION_ACTION flag cleared and Session-Group-Id AVP absent.

4.2.3. Mid-session Group Assignment Modifications

Either Diameter node (client or server) 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 AVPs 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 authorization 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 decides to update assigned groups mid-session, it sends a Re-Auth-Request (RAR) message or a service-specific request to the client identifying the session for which the session group lists are to be updated. The client responds with a Re-Auth-Answer (RAA) message or a service-specific answer. The client subsequently sends a 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 authorization response and includes 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

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 a server wants to remove the session from all previously assigned session groups, it sends at least one Session- Group-Info AVP with the response having the SESSION_GROUP_ALLOCATION_ACTION flag cleared and no Session-Group-Id AVP present.

4.3. Deleting a Session Group

To explicitly 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 with the SESSION_GROUP_STATUS flag cleared and the Session-Group-Id AVP identifying the session group that is to be deleted. The SESSION_GROUP_ALLOCATION_ACTION flag of the associated Session-Group-Control-Vector AVP MUST be cleared.

A session group is implicitly deleted and its identifier is released after the last session has been removed from this session group.

4.4. Performing Group Operations

4.4.1. Sending Group Commands

Either Diameter node (client or server) can request the recipient of a request to process an associated command for all sessions 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 and the SESSION_GROUP_STATUS flag MUST be set.

If the Command Code Format (CCF) of the request mandates a Session-Id AVP, the Session-Id AVP MUST identify one of the single sessions that 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 multiple resulting transactions associated with a group command are to be treated by appending a single instance of a Group-Response-Action AVP. For example, when a server sends a Re-Auth-Request (RAR) or a service-specific server-initiated request to the client, it indicates to the client to follow the request according to one of three possible procedures. When the server sets the Group-Response-Action AVP to ALL_GROUPS (1), the client sends a single RAR message for all identified groups. When the server sets the Group-Response-Action AVP to PER_GROUP (2), the client sends a single RAR message for each identified group individually. When the server sets the Group-Response-Action AVP to PER_SESSION (3), the client follows up with a single RAR message per impacted session. If a session is included in more than one of the identified session groups, the client sends only one RAR message for that session.

If the sender sends a request including the Group-Response-Action AVP set to ALL_GROUPS (1) or PER_GROUP (2), it has to expect some delay before receiving one or more corresponding answers, as the answers will only be sent back when the request is processed for all the sessions or all the sessions of a session group. If the processing of the request is delay sensitive, the sender SHOULD NOT set the Group-Response-Action AVP to ALL_GROUPS (1) or PER_GROUP (2). If the answer can be sent before the complete process of the request for all the sessions or if the request timeout timer is high enough, the sender MAY set the Group-Response-Action AVP to ALL_GROUPS (1) or

PER_GROUP (2).

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

4.4.2. Receiving Group Commands

A Diameter node receiving a request to process a command for a group of sessions identifies the relevant groups according to the included Session-Group-Id AVP in the Session-Group-Info AVP and processes the group command according to the included Group-Response-Action AVP. If the received request identifies multiple groups in multiple, included 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.

4.4.3. Error Handling for Group Commands

When a Diameter node 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 node 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 the case of limited success, the sessions for which the processing of the group command failed MUST be identified by including their Session-Id AVP in the Failed-AVP AVP, as per Section 7.5 of [RFC6733]. The sender of the request MUST fall back to single-session operation for each of the identified sessions for which the group command failed. In addition, each of these sessions MUST be removed from all session groups to which the group command applied. To remove sessions from a session group, the Diameter client performs the procedure described in Section 4.2.2.

4.4.4. Single-Session Fallback

Either Diameter node 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. In such case, the response to the group command MUST NOT include any group identifier but only the Session-Id identifying the session for which the command has been processed.

5. Operation with Proxy Agents

In the case of a present stateful Proxy Agent between a Diameter client and a Diameter server, the Proxy Agent MUST perform the same mechanisms per this specification to advertise session grouping and group operation capabilities towards the client and the server, respectively. The Proxy Agent MUST update and maintain consistency of its local session states as per the result of the group commands that are operated between a Diameter client and a server. In such

case, the Proxy Agent MUST act as a Diameter server in front of the Diameter client and MUST act as a Diameter client in front of the Diameter server. Therefore, the client and the server behavior described in Section 4 applies respectively to the stateful Proxy Agent.

If a stateful Proxy Agent manipulates session groups, it MUST maintain consistency of session groups between a client and a server. This applies to a deployment where the Proxy Agent utilizes session grouping and performs group operations with, for example, a Diameter server, whereas the Diameter client is not aware of session groups. In such case, the Proxy Agent must reflect the states associated with the session groups as individual session operations towards the client and ensure the client has a consistent view of each session. The same applies to a 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.

Stateless Proxy Agents do not maintain any session states (only transaction states are maintained). Consequently, the notion of a session group is transparent for any stateless Proxy Agent present between a Diameter client and a Diameter server handling session groups. Session-group-related AVPs being defined as an optional AVP are ignored by stateless Proxy Agents and should not be removed from the Diameter commands. If they are removed by the Proxy Agent for any reason, the Diameter client and Diameter server will discover the absence of the session-group-related AVPs and will fall back to single-session processing, as described in Section 4.

6. Commands Formatting

This document does not specify new Diameter commands to enable group operations but relies on command extensibility and 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 apply the command to all sessions associated with the identified group or groups.

6.1. Formatting Example: Group Re-Auth-Request

A request for re-authentication of one or more groups of users is issued by appending one or multiple Session-Group-Id AVPs, as well as appending a single instance of a Group-Response-Action AVP to the Re-Auth-Request (RAR). One or multiple Session-Group-Id AVPs identify one or more associated groups for which 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 or groups. 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 (AVPs)

Attribute Name	AVP Code	AVP Flag rules			
	Value Type	MUST	MAY	SHOULD NOT	MUST NOT
Session-Group-Info	671 Grouped		P		V
Session-Group-Control-Vector	672 Unsigned32		P		V
Session-Group-Id	673 UTF8String		P		V
Group-Response-Action	674 Unsigned32		P		V
Session-Group-Capability-Vector	675 Unsigned32		P		V

Table 1: AVPs for the Diameter Group Signaling

7.1. Session-Group-Info AVP

The Session-Group-Info AVP (AVP Code 671) 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: 671 >
                       < Session-Group-Control-Vector >
                       [ Session-Group-Id ]
                     * [ AVP ]
```

7.2. Session-Group-Control-Vector AVP

The Session-Group-Control-Vector AVP (AVP Code 672) is of type Unsigned32 and contains a 32-bit flag field to control the group assignment at session-group-aware nodes. For defined flags, only numeric values that are 2^x (power of two, where 0<=x<32) are allowed.

The following control flags 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 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 (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 include 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 673) is of type UTF8String and identifies a group of Diameter sessions.

The Session-Group-Id MUST be globally unique. The Session-Group-Id includes a mandatory portion and an implementation-defined portion delimited by the ";" character. The Session-Group-Id MUST begin with the identity of the Diameter node that owns the session group. The remainder of the Session-Group-Id is implementation defined and MAY follow the format recommended for the implementation-defined portion of the Session-Id AVP in Section 8.8 of [RFC6733].

7.4. Group-Response-Action AVP

The Group-Response-Action AVP (AVP Code 674) is of type Unsigned32 and contains a 32-bit address space representing values indicating how the node SHOULD issue follow-up exchanges in response to a command that impacts multiple sessions. The following values are defined by this document:

ALL_GROUPS (1)
Follow-up message exchanges associated with a group command should be performed with a single message exchange for all impacted groups.

PER_GROUP (2)
Follow-up message exchanges associated with a group command should be performed with a separate message exchange for each impacted group.

PER_SESSION (3)
Follow-up message exchanges associated with a group command should be performed with a separate message exchange for each impacted session.

7.5. Session-Group-Capability-Vector AVP

The Session-Group-Capability-Vector AVP (AVP Code 675) is of type Unsigned32 and contains a 32-bit flag field to indicate capabilities in the context of session-group assignment and group operations. For defined flags, only numeric values that are 2^x (power of two, where 0<=x<32) are allowed. The value of (0) is reserved.

The following capability is 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. Documents specifying 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

IANA has registered the following new AVPs from the "AVP Codes" registry defined in [RFC6733]. The AVPs are defined in Section 7.

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

9.2. New Registries

IANA has created the following two new registries.

* The "Session-Group-Control-Vector AVP Values (code 672)" registry for control bits. Two initial assignments are described in Section 7.2. The registration assignment policy is Specification Required.
* The "Session-Group-Capability-Vector AVP Values (code 675)" registry. One initial assignment is described in Section 7.5. The registration assignment policy is Standards Action.

10. Security Considerations

The security considerations of the Diameter protocol itself are discussed in [RFC6733]. Use of the AVPs defined in this document MUST take into consideration the security issues and requirements of the Diameter base protocol. In particular, the Session-Group-Info AVP (including the Session-Group-Control-Vector and the Session-Group-Id AVPs) should be considered as a security-sensitive AVP in the same manner as the Session-Id AVP in the Diameter base protocol [RFC6733].

The management of session groups relies upon the existing trust relationship between the Diameter client and the Diameter server managing the groups of sessions. This document defines a mechanism that allows a client or a server to act on multiple sessions at the same time using only one command. If the Diameter client or server is compromised, an attacker could launch DoS attacks by terminating or applying change operations to a large number of sessions with a limited set of commands using the session group management concept.

According to the Diameter base protocol [RFC6733], transport

connections between Diameter peers are protected by TLS/TCP, DTLS / Stream Control Transmission Protocol (SCTP), or alternative security mechanisms that are independent of Diameter, such as IPsec. However, the lack of end-to-end security features makes it difficult to establish trust in the session-group-related information received from non-adjacent nodes. Any Diameter agent in the message path can potentially modify the content of the message and therefore the information sent by the Diameter client or the server. There is ongoing work on the specification of end-to-end security features for Diameter. Such features would enable the establishment of a trust relationship between non-adjacent nodes, and the security required for session group management would normally rely on this end-to-end security. However, there is no assumption in this document that such end-to-end security mechanism will be available. It is only assumed that the solution defined on this document relies on the security framework provided by the Diameter-based protocol.

In some cases, a Diameter Proxy Agent can act on behalf of a client or a server. In such case, the security requirements that normally apply to a client (or a server) apply equally to the Proxy Agent.

11. Normative References

[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>.

[RFC6733] Fajardo, V., Ed., Arkko, J., Loughney, J., and G. Zorn, Ed., "Diameter Base Protocol", RFC 6733, DOI 10.17487/RFC6733, October 2012, <https://www.rfc-editor.org/info/rfc6733>.

[RFC7155] Zorn, G., Ed., "Diameter Network Access Server Application", RFC 7155, DOI 10.17487/RFC7155, April 2014, <https://www.rfc-editor.org/info/rfc7155>.

[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>.

Appendix A. Session Management -- Exemplary Session State Machine

A.1. Use of Groups for the Authorization Session State Machine

Section 8.1 of [RFC6733] defines a set of finite state machines that represent the life cycle of Diameter sessions, 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, for example, additional state transitions related to the processing of the group commands that may impact multiple sessions.

The group membership is a 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 or NASREQ).

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

The Diameter group command in the following tables is differentiated from a single-session-related command by a preceding 'G' (Group). 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].

Additionally, the following acronyms are used in the tables below.

GASR: Group-Abort-Session-Request

GASA: Group-Abort-Session-Answer

GSTA: Group-Session-Termination-Answer

GSTR: Group-Session-Termination-Request

CLIENT, STATEFUL			
State	Event	Action	New State
Idle	Client or Device Requests access	Send service-specific authorization 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 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 sessions 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

Table 2: Group Authorization Session State Machine for Stateful Client

The following state machine is observed by a server when it is maintaining the state for the session. State transitions that 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-authorization 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

Table 3: Group Authorization Session State Machine for Stateful Server

Acknowledgments

The authors of this document want to thank Ben Campbell and Eric McMurry for their valuable comments to early draft versions of this document. Furthermore, the authors thank Steve Donovan and Mark Bales for the thorough review and comments on advanced versions of the WG document, which helped a lot to improve this specification.

Authors' Addresses

Mark Jones
Individual
Email: mark@azu.ca

Marco Liebsch
NEC Laboratories Europe GmbH
Kurfuersten-Anlage 36
D-69115 Heidelberg
Germany
Email: marco.liebsch@neclab.eu

Lionel Morand
Orange Labs
38/40 rue du General Leclerc
92794 Issy-Les-Moulineaux Cedex 9
France
Email: lionel.morand@orange.com