Independent Submission
Request for Comments: 6159
Category: Informational
ISSN: 2070-1721

T. Tsou
Huawei Technologies (USA)
G. Zorn
Network Zen
T. Taylor, Ed.
Huawei Technologies

April 2011

Session-Specific Explicit Diameter Request Routing

Abstract

This document describes a mechanism to enable specific Diameter proxies to remain in the path of all message exchanges constituting a Diameter session.

Status of This Memo

IESG Note

Techniques similar to those discussed in this document were discussed in the IETF Diameter Maintenance and Extensions (DIME) Working Group. The group had no consensus that the problems addressed by such work are a real concern in Diameter deployments. Furthermore, there was no consensus that the proposed solutions are in line with the architectural principles of the Diameter protocol. As a result, the working group decided not to undertake the work. There has also not been a formal request for this functionality from any standards body. This RFC represents a continuation of the abandoned work. Readers of this specification should be aware that the IETF has not reviewed this specification and cannot say anything about suitability for a particular purpose or compatibility with the Diameter architecture and other extensions.

Copyright Notice

Table of Contents

1. Introduction 2
2. Terminology 3
3. The 3GPP Wireless LAN (WLAN) Access Architecture 4
   3.1. Maintaining the Routing Path 5
4. Diameter Explicit Routing (ER) 6
   4.1. Originating a Request (ER-Originator) 6
   4.2. Relaying and Proxying Requests (ER-Proxy) 8
   4.3. Receiving Requests (ER-Destination) 10
   4.4. Diameter Answer Processing 11
   4.5. Failover and Failback Considerations 12
   4.6. Attribute-Value Pairs 12
        4.6.1. Explicit-Path-Record AVP 12
               4.6.1.1. Proxy-Host AVP 13
               4.6.1.2. Proxy-Realm AVP 13
        4.6.2. Explicit-Path AVP 13
   4.7. Error Handling 13
5. Example Message Flow 14
6. RADIUS/Diameter Protocol Interactions 16
7. Security Considerations 17
8. Acknowledgements 17
9. References 18
   9.1. Normative References 18
   9.2. Informative References 18

1. Introduction

In the Diameter base protocol [RFC3588], the routing of request messages is based solely on the routing decisions made separately by each node along the path. [RFC5729] has added the ability to force messages to pass through a specified set of realms through the use of Network Access Identifier (NAI) decoration. However, no other specification provides the ability to force routing through a specific set of agents. Therefore, in a topology where multiple paths exist from source to destination, there is no guarantee that all messages relating to a given session will take the same path. In general, this has not caused problems, but some architectures (e.g., WLAN Third Generation Partnership Project (3GPP) IP access [TS23.234]) require that once certain agents become engaged in a session, they be able to process all subsequent messages for that session.

While the solution presented in this document is valid, it violates one of the basic premises of Diameter -- the robustness of its architecture. With normal Diameter routing, sessions will survive failures of agents along the routing path. With the proposals in this document, routing becomes pinned to specific agents whose failure will terminate the session.

The authors see no interaction between explicit routing and the specific applications with which it is employed. Hence, in principle it can be added to existing applications if they support the necessary extensibility, and equally can be used with new applications.

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 [RFC2119].

The following terms are used to define the functionality and participants in the routing extensions described in this document.

ER
Explicit routing -- the mechanism provided by this specification to allow proxies traversed by the initial message of a session to ensure that they remain on the messaging path for all subsequent request messages of a session.

ER-Proxy
A proxy that implements the ER mechanism and can therefore use it to remain in the path for subsequent messages of a session.

ER-Destination
A Diameter node that is capable of participating in ER and that will ultimately consume the request sent by an ER-Originator.

ER-Originator
A Diameter node initiating a session and sending the requests. The ER-Originator can be any Diameter node sending a request, i.e., a client, server or proxy capable of initiating sessions and participating in ER.

Authentication, Authorization, and Accounting (AAA) Relays

Other Diameter nodes interspersed between the ER-Originator, ER-Proxies, and the ER-Destination. These nodes represent existing Diameter agents and proxies that do not participate in ER and do not recognize Explicit-Path Attribute Value Pairs (AVPs).

3. The 3GPP Wireless LAN (WLAN) Access Architecture

The 3GPP WLAN IP access architecture [TS23.234] is one example of a system requiring that certain agents (stateful proxies, in this case) remain in the forwarding path of all session messages. The 3GPP WLAN interworking architecture extends 3GPP services to the WLAN access side, enabling a 3GPP subscriber to use a WLAN to access 3GPP services.

WLAN AAA provides access to the WLAN to be authenticated and authorized through the 3GPP system. This access control can permit or deny a subscriber access to the WLAN system and/or the 3GPP system.

There are two 3GPP WLAN interworking reference models:

1. In the non-roaming case, the model includes the WLAN access network and the 3GPP AAA server in the home network. The 3GPP AAA server is responsible for access control as well as charging.

2. In the roaming case, the model includes the WLAN access network, the 3GPP AAA proxy in the visited network, and the 3GPP AAA server in the home network. The 3GPP AAA server is responsible for access control. Charging records may be generated by the AAA proxy and/or the AAA server. The AAA proxy relays access control and charging messages to the AAA server. The AAA proxy will also do offline charging, if required.

The roaming case presents two problems for which the Diameter routing mechanism described in [RFC3588] does not offer any unambiguous and standard solution.

Network Selection

Selecting an initial message path for the Diameter session through (possibly many) alternative visited network(s) to the home network.

Explicit Routing (ER)

Maintaining the selected message path for all messages in the Diameter session.

Selecting an initial message path is outside the scope of this document. A mechanism for maintaining the selected message path is described in detail below.

3.1. Maintaining the Routing Path

After a successful authentication, a Diameter session is established involving (at least) the following stateful entities:

- the Diameter client in the WLAN access node (e.g., the 3GPP AAA client in the terminal visited network),
- a Diameter proxy in the visited mobile network (e.g., the 3GPP AAA proxy in the terminal visited network), and
- a Diameter server in the user’s home realm (e.g., the destination 3GPP AAA server in the terminal home network).

Message routing for the initial session request uses the normal Diameter routing tables (Section 2.7 of [RFC3588]) in the 3GPP AAA client, the 3GPP AAA proxy in the visited network, and any intermediate proxies after that. The 3GPP AAA client sends the initial session request to the 3GPP AAA proxy in the visited network. The 3GPP AAA proxy processes the request, then forwards it towards the destination 3GPP AAA server, through an intermediate proxy if necessary. The request may be forwarded through other intermediate proxies in the same way, until it reaches the destination 3GPP AAA server in the terminal home network.

The functions assigned to the 3GPP AAA proxy include:

- Reporting charging information to the offline charging system in the visited network,
- Policy enforcement based on roaming agreements, and
- Service termination initiated by the visited network’s operator.

These functions all require that state be maintained within the visited network. The 3GPP’s choice is to maintain that state at the 3GPP AAA proxy. This means that the latter must remain in the messaging path for all subsequent messages relating to the same session.

4. Diameter Explicit Routing (ER)

This section outlines a Diameter ER mechanism by which Diameter nodes participating in ER can remain in the path of all request messages for a specific session. A new Explicit-Path AVP is defined to enable ER participants to manipulate the Destination-Host and/or Destination-Realm AVPs of request messages in order to ensure the correct routing behavior. The following sections describe the extensions to the request routing in [RFC3588] to implement the ER mechanism. The proposed extensions utilize existing routing strategies in [RFC3588] and do not mandate modifications to it. The mechanism imposes loose rather than strict source routing, in that subsequent messages of a session are forced through the participating nodes, but not through any individual non-participating nodes. In summary, only Diameter nodes interested in participating in the ER scheme will be involved in it.

4.1. Originating a Request (ER-Originator)

A Diameter node acting as an ER-Originator for a particular session MUST maintain a local cache that enumerates all the Diameter identities of the ER-Proxies that the request messages must traverse along the path to the ER-Destination. The identity of a Diameter node is defined in [RFC3588]. The local cache MAY also include the node’s realm. The data structure of the cache is left up to the implementation and SHOULD persist as part of the session attributes or properties.

An ER-Originator sending request messages MUST add an Explicit-Path AVP to these requests. The contents of the cache SHOULD be used to populate the Explicit-Path AVP, with each cached entry represented by a corresponding instance of the Explicit-Path-Record AVP. ER-Proxies along the path of the request message MUST examine the contents of the Explicit-Path AVP and make routing adjustments based on records it contains. An example of the message flow is shown in Section 5. Note that the ER-Originator can be any Diameter node, i.e., a client, server, or proxy.

The ER-Originator can populate the cache either by pre-configuring its contents or by using the first request message of the session to gather identities of participating ER-Proxies along the routing path. The latter scheme is known as Explicit-Path discovery. The contents of the cache can be pre-configured if the ER-Originator has explicit knowledge of the ER-Proxies the request messages must traverse; otherwise, the ER-Originator can use Explicit-Path discovery. It is RECOMMENDED that Explicit-Path discovery be used whenever possible since pre-configuration is less flexible by nature.

Explicit-Path discovery is useful if the identities of the ER-Proxies are not known or if there are several ER-capable proxies (a cluster of proxies) that can be dynamically chosen based on other routing policies. In Explicit-Path discovery, the cache of the ER-Originator is initially empty. To initiate discovery, when the ER-Originator sends the first request message of a session, it MUST include the Explicit-Path AVP containing a single Explicit-Path-Record AVP with the identity and/or the realm of the ER-Originator. The ER-Originator MUST set the Destination-Host and/or Destination-Realm AVP of the request message to the identity and/or the realm of the ER-Destination, respectively, as specified in [RFC3588].

> Note that ER-Originator initial request message routing procedures and the process of population of the Destination-Realm may be affected by the User-Name AVP NAI decoration [RFC5729]. NAI decoration is a form of request message source routing and defines realms that the request message must traverse through before routing towards the ER-Destination. Diameter nodes participating in request message routing must examine and process the User-Name AVP, and modify the Destination-Realm AVP accordingly as long as there are realms left in the decorated NAI. Source routing based upon NAI decoration does not affect Explicit-Path discovery as defined in this document.

If the path taken by the initial request encounters one or more participating ER-Proxies and a participating ER-Destination, the procedures described in Section 4.2 and Section 4.3 ensure that a successful response to that request will contain an Explicit-Path AVP that includes one or more Explicit-Path-Records containing the ER-Originator’s identity, the identities of all participating ER-Proxies, and the identity of the ER-Destination. The ER-Originator SHOULD populate its local cache with the contents of the Explicit-Path AVP received in this initial answer message.

If the answer message does not contain an Explicit-Path AVP or the Result-Code AVP is set to DIAMETER_ER_NOT_AVAILABLE (Section 4.7), it is an indication to the ER-Originator that the destination of the request does not support ER and that the ER-Originator SHOULD avoid sending an Explicit-Path AVP in subsequent request messages.

If the initial request message initiated Explicit-Path discovery, but the Explicit-Path AVP in the answer message contains Explicit-Path-Records for the ER-Originator and ER-Destination only, it is an indication to the ER-Originator that there are no Diameter proxies capable of participating in ER along the path and that the ER-Originator SHOULD NOT send an Explicit-Path AVP in subsequent request messages of this session. See Section 4.5 for more discussion. In such cases, the situation may be transient, and

Explicit-Path discovery may find participating proxies in succeeding sessions. It is left up to the ER-Originator to decide if Explicit-Path discovery should be attempted in succeeding sessions.

Once the ER-Originator’s local cache has been populated, whether by pre-configuration or through Explicit-Path discovery, all request messages for the session MUST include the Explicit-Path AVP using the contents of the local cache. The Explicit-Path AVP MUST contain the Explicit-Path-Records of all the nodes enumerated in the cache except that of the ER-Originator itself. The identities enumerated in the Explicit-Path AVP MUST appear in the order they will be traversed in the routing path. The last entry in the Explicit-Path AVP MUST be the Explicit-Path-Record of the ER-Destination. In addition, the value of the Destination-Host and possibly the Destination-Realm in the request message MUST be copied from the values of the Proxy-Host AVP and, if present, the Proxy-Realm AVP of the first Explicit-Path-Record AVP present in the Explicit-Path AVP.

> This ensures that the ER-Originator as well as any AAA relays between the ER-Originator and the first ER-Proxy will route the message towards the first ER-Proxy as specified in RFC 3588 [RFC3588].

Subsequent actions taken by the first ER-Proxy upon receipt of the message are described in Section 4.2 and will mimic those of the ER-Originator.

Answer messages received by the ER-Originator to subsequent request messages after the Explicit-Path has been established SHOULD NOT have an Explicit-Path AVP. If they do, this SHOULD be considered a suspect condition that may be caused by a misbehaving ER participant. It is left up to the ER-Originator whether to continue using the ER scheme when such a condition arises or to attempt another Explicit-Path discovery for subsequent sessions.

4.2. Relaying and Proxying Requests (ER-Proxy)

The basic action taken by an ER-Proxy upon receiving a request is to check whether explicit routing is supported in the request and if so, check whether it is already a participant in explicit routing for the said request. If it is not an existing participant, if Explicit-Path discovery is in progress, and if it wishes to participate, it appends an Explicit-Path-Record AVP identifying itself to the end of the Explicit-Path AVP. If it is an existing participant, the ER-Proxy pops/removes the Explicit-Path-Record AVP pertaining to itself from the Explicit-Path AVP and then uses the next Explicit-Path-Record AVP for subsequent routing. Details of this operation follow.

An ER-Proxy is not required to keep local state or cache state regarding the explicit routing procedure. However, it MUST check whether an incoming request contains an Explicit-Path AVP. The following cases can occur.

1. If an incoming request does not contain an Explicit-Path AVP, then the ER-Proxy takes no action beyond processing and forwarding the request as specified in [RFC3588].

2. If the incoming request contains an Explicit-Path AVP, the ER-Proxy MUST check whether its identity is present in the Explicit-Path AVP. Determining whether its identity is present can be done by matching its identity to the Proxy-Host AVP contained in each Explicit-Path-Record. If its identity is not present, then:

   A. If it wishes to participate in explicit routing, the ER-Proxy MUST verify that Explicit-Path discovery is in progress by verifying that the Proxy-Host AVP in the first Explicit-Path-Record AVP in the Explicit-Path AVP does not match the Destination-Host AVP (if present). If this verification succeeds or the Destination-Host AVP is absent, the ER-Proxy MAY append a new Explicit-Path-Record as the last AVP in the Explicit-Path AVP prior to forwarding the request. The new Explicit-Path-Record MUST contain a Proxy-Host AVP set to the proxy’s identity, and MAY contain a Proxy-Realm AVP giving the proxy’s realm. If, however, the Destination-Host AVP is present and matches the Proxy-Host AVP of the first Explicit-Path-Record AVP, then the Explicit-Path contains an already-defined source route that does not include the ER-Proxy. The ER-Proxy SHOULD process the request as if the ER-Path AVP were absent.

   B. If the ER-Proxy does not wish to participate in the ER, it SHOULD NOT modify the Explicit-Path AVP and SHOULD simply process and forward the request as specified in [RFC3588] using the existing values of the Destination-Host and/or Destination-Realm AVPs. Non-ER-Proxies and relays that do not support ER and do not recognize Explicit-Path AVP will take the same action.

3. If the identity of the ER-Proxy is present in the Explicit-Path AVP, then:

   A. If it is not the first Explicit-Path-Record in the AVP, this MUST be considered an error, and an answer message with the 'E' bit set and the Result-Code set to DIAMETER_INVALID_PROXY_PATH_STACK MUST be sent back to the ER-Originator (Section 4.7).

   B. If the identity of the ER-Proxy matches the first Explicit-Path-Record, the ER-Proxy MUST remove this record from the Explicit-Path AVP and repopulate the Destination-Host and possibly the Destination-Realm AVP from the next Explicit-Path-Record present in the Explicit-Path AVP. Setting the Destination-Host and possibly the Destination-Realm AVP will ensure that the ER-Proxy as well as all AAA relays between the current ER-Proxy and the next ER-Proxy enumerated in the Explicit-Path AVP will route the message towards the next ER-Proxy. The process of removing the ER-Proxy's record is analogous to popping an entry from a stack represented by the Explicit-Path AVP.

The behavior specified above also applies to a Diameter node that acts as a relay agent and participates in the ER scheme.

4.3. Receiving Requests (ER-Destination)

A Diameter node that locally processes requests sent by the ER-Originator (Section 4.1) and is able to support ER (an ER-Destination) MUST check for the presence of an Explicit-Path AVP in the request message.

1. If an incoming request does not contain an Explicit-Path AVP, then it is an indication that messages belonging to this session will not use ER. The ER-Destination MUST simply process the request for local consumption and formulate an answer message as specified in [RFC3588].

2. If the incoming request contains an Explicit-Path AVP, the ER-Destination MUST check whether its identity is present in the Explicit-Path AVP. If its identity is not present, indicating that Explicit-Path discovery is in progress, then:

   A. If it wishes to participate in the ER, and subject to paragraph B below, the ER-Destination MUST append a new Explicit-Path-Record to the Explicit-Path AVP in the received message. The new Explicit-Path-Record MUST contain at the least a Proxy-Host AVP set to the ER-Destination's identity. The ER-Destination MUST then copy the resulting Explicit-Path AVP to the subsequent answer message.

   B. If there is only one Explicit-Path-Record in the incoming Explicit-Path AVP, then this is an indication of a successful Explicit-Path discovery, but with no participating ER-Proxies. The ER-Destination SHOULD NOT copy the Explicit-Path AVP into the subsequent answer message.

   C. If the ER-Destination supports ER but does not wish to or cannot participate, it MAY send a Result-Code AVP set to DIAMETER_ER_NOT_AVAILABLE as defined in Section 4.7. The ER-Destination MUST NOT include any Explicit-Path AVP in the subsequent answer. Diameter servers that do not support ER and do not recognize the Explicit-Path AVP will also omit the Explicit-Path AVP from the answer message.

3. If the identity of the ER-Destination matches a record in the Explicit-Path AVP, then it MUST be the only Explicit-Path-Record present in the Explicit-Path AVP. Otherwise, this MUST be considered an error, and an answer message with the 'E' bit set and containing an Experimental-Result-Code AVP set to DIAMETER_INVALID_PROXY_PATH_STACK MUST be sent back to the ER-Originator (Section 4.7). If the identity of the ER-Destination does match the only existing Explicit-Path-Record, then this is an indication that the request reached the ER-Destination by way of a successfully executed explicit route. The ER-Destination MUST NOT include the Explicit-Path AVP in the subsequent answer message.

4.4. Diameter Answer Processing

There is no requirement on Diameter nodes participating in ER to provide special handling or routing of answer messages. Answer messages SHOULD be processed normally as specified in [RFC3588]. However, a Diameter node acting as an ER-Destination MUST formulate a proper Explicit-Path AVP in answer messages as described in Section 4.3.

4.5. Failover and Failback Considerations

If there is no ER-Proxy along the selected path, the answer message MAY contain an Explicit-Path AVP that contains only the Explicit-Route-Records of the ER-Originator and the ER-Destination, indicating that there is no ER support found in Diameter nodes along the path. It is left to the ER-Originator to continue with processing of the request without ER support or terminate the session. The ER-Originator SHOULD NOT attempt to perform Explicit-Path discovery in subsequent request messages of this session in such cases, to protect against failback conditions where an ER-Proxy suddenly appears in the path and attempts to add a new Explicit-Path-Record for request messages other than the initial request.

> Allowing an ER-Proxy to join the session after the initial request makes sense only if the application requirements do not mandate that every participating ER-Proxy receive all of the messages of a session.

However, depending on local policy, the ER-Originator MAY attempt ER path discovery in subsequent sessions despite the lack of proxy participants in the earlier attempt.

If a failover occurs in a Diameter node preceding an ER-Proxy when the Explicit-Path is already established, it is possible that a DIAMETER_UNABLE_TO_DELIVER error will be received by the ER-Originator if there are no alternative paths towards the ER-Proxy. In such a case, it is left to the ER-Originator to handle the error as specified in the Diameter application or in [RFC3588].

4.6. Attribute-Value Pairs

The following sections define the AVPs used in the ER process. All of these AVPs MUST have the ’V’ bit set and the ’M’ bit cleared, with the Vendor-ID field set to 2011 (as assigned by IANA in "Private Enterprise Numbers" registry; see http://www.iana.org/).

4.6.1. Explicit-Path-Record AVP

The Explicit-Path-Record AVP (AVP Code 35001) is of type Group. The identity added in the Proxy-Host [RFC3588] element of this AVP MUST be the same as the one advertised by the Diameter node in the Origin-Host AVP during the Capabilities Exchange messages.

```
     Explicit-Path-Record ::= < AVP Header: 35001 >
                              { Proxy-Host }
                              [ Proxy-Realm ]
```

4.6.1.1. Proxy-Host AVP

The Proxy-Host AVP (AVP Code 35004) is of type DiameterIdentity. It identifies the ER node that is inserting the record. The Proxy-Host AVP MUST be present.

4.6.1.2. Proxy-Realm AVP

The Proxy-Realm AVP (AVP Code 35002) is of type DiameterIdentity, and contains the realm of the ER node inserting the record. The Proxy-Realm AVP MAY be present in the Explicit-Path-Record. If it is present, the realm name included in the value of the Proxy-Host AVP MUST match the value of the Proxy-Realm AVP.

4.6.2. Explicit-Path AVP

The Explicit-Path AVP (AVP Code 35003) is of type Grouped. This AVP MUST be present in all request messages performing ER. It MAY be present in the answer to the initial session request message if Explicit-Path discovery was successfully executed for the request.

```
      Explicit-Path ::= < AVP Header: 35003 >
                     1* [ Explicit-Path-Record ]
                      * [ AVP ]
```

4.7. Error Handling

The following error conditions may occur during ER processing. All error indications MUST be encapsulated in an instance of the Experimental-Result AVP [RFC3588] with the Vendor-ID AVP set to 2011 and the Experimental-Result-Code set as specified below.

DIAMETER_INVALID_PROXY_PATH_STACK 3501

A request message received by an ER-Proxy or ER-Destination after an Explicit-Path has been established has the first or only Explicit-Path-Record AVP not matching the ER-Proxy’s or the ER-Destination’s identity. The same error applies to ER-Destinations receiving an Explicit-Path-AVP containing more than one Explicit-Path-Record or an Explicit-Path-AVP with only one Explicit-Path-Record not matching its own identity.

This error SHOULD be considered a protocol failure and SHOULD be treated on a per-hop basis; Diameter proxies may attempt to correct the error, if possible. Diameter answer messages containing this error indication MUST have the ’E’ bit set and MUST conform to Section 7.2 of [RFC3588].

DIAMETER_ER_NOT_AVAILABLE 4501

An ER-Destination that supports ER routing but is unable to comply for unknown reasons MAY send an answer message with the Result-Code AVP set to this error code. This error value SHOULD be considered a transient failure indicating that subsequent ER attempts may succeed.

5. Example Message Flow

The example presented here illustrates the flow of Diameter messages with the typical attributes present in the ER scenario.

The ER-Originator in the example below shows the use of Explicit-Path discovery with the first request. However, the ER-Originator could also use a pre-configured cache. The ER-Originator can be any Diameter node sending a request, i.e., a client, server, or proxy. In this scenario, the local cache of the ER-Originator is initially empty.

The AAA relays between the ER-Proxies, ER-Originator, and ER-Destination may or may not be present and are shown here to depict routing paths that the requests may take prior to being processed by nodes participating in the ER scheme. The AAA relays also depict existing Diameter relays or proxies that do not recognize Explicit-Path AVPs and therefore do not participate in ER.

```
      ER-                    ER-                  ER-         ER-
  Originator   AAA relays   Proxy1   AAA relays   Proxy2    Destination
     (o.r1                 (p.r1                (p.r2       (d.r2
    .example)              .example)            .example)   .example)
                    |          |          |          |          |
  cache=(empty)     |          |          |          |          |
      ------------->|--------->|          |          |          |
   (1st request of the session)|          |          |          |
        Explicit-Path=         |          |          |          |
          o.r1.example,r1.example         |          |          |
    dest-host=d.r2.example     |          |          |          |
    dest-realm=r2.example      |          |          |          |
                    |          |          |          |          |
                    |          |--------->|--------->|          |
                    |          |  (forwarded request)|          |
                    |          |  Explicit-Path=     |          |
                    |          |    record1=o.r1.example,r1.example
                    |          |    record2=p.r1.example,r1.example
                    |          |  dest-host=d.r2.example        |
                    |          |  dest-realm=r2.example         |
                    |          |          |          |          |
                    |          |          |          |--------->|
                    |          |          |      (forwarded request)
                    |          |          |      Explicit-Path=
                    |          |          |       record1=o.r1.example,
                    |          |          |               r1.example
                    |          |          |       record2=p.r1.example,
                    |          |          |               r1.example
                    |          |          |       record3=p.r2.example,
                    |          |          |               r2.example
                    |          |          |     dest-host=d.r2.example
                    |          |          |     dest-realm=r2.example
                    |          |          |          |          |
   cache=           |<---------|<---------|<---------|<---------|
     record1=o.r1.example,r1.example         (answer)           |
     record2=p.r1.example,r1.example   Explicit-Path=
     record3=p.r2.example,r2.example    record1=o.r1.example,r1.example
     record4=d.r2.example,r2.example    record2=p.r1.example,r1.example
                    |          |        record3=p.r2.example,r2.example
                    |          |        record4=d.r2.example,r2.example
   Note: An originator pre-configuring    |          |          |
         its local cache can skip the     |          |          |
         exchange above and send the      |          |          |
         initial request as shown below.  |          |          |
                    |          |          |          |          |
      ------------->|--------->|          |          |          |
   (subsequent request of the session)    |          |          |
        Explicit-Path=         |          |          |          |
  record1=p.r1.example,r1.example         |          |          |
  record2=p.r2.example,r2.example         |          |          |
  record3=d.r2.example,r2.example         |          |          |
    dest-host=p.r1.example     |          |          |          |
    dest-realm=r1.example      |          |          |          |
                    |          |--------->|--------->|          |
                    |          |  (forwarded request)|          |
                    |          |  Explicit-Path=     |          |
                    |          |      record1=p.r2.example,r2.example
                    |          |      record2=d.r2.example,r2.example
                    |          |  dest-host=p.r2.example        |
                    |          |  dest-realm=r2.example         |
                    |          |          |          |          |
                    |          |          |          |--------->|
                    |          |          |     (forwarded request)
                    |          |          |     Explicit-Path
                    |          |          |       record1=d.r2.example,
                    |          |          |               r2.example
                    |          |          |     dest-host=d.r2.example
                    |          |          |     dest-realm=r2.example
                    |          |          |          |          |
   cache=           |<---------|<---------|<---------|<---------|
     record1=o.r1.example,r1.example    (answer)     |          |
     record2=p.r1.example,r1.example    * no Explicit-Path-AVP present
     record3=p.r2.example,r2.example      |          |          |
     record4=d.r2.example,r2.example      |          |          |
                    |          |          |          |          |
                    |          |          |          |          |
    (subsequent request of the session will repeat the process above)
                    |          |          |          |          |
                    |          |          |          |          |
```

Figure 1: Example ER Message Flow

6. RADIUS/Diameter Protocol Interactions

No actions need to be taken with regards to RADIUS/Diameter interaction. The routing extension described in this document is transparent to any translation gateway and relevant only to Diameter routing. The assumption is that if there is a RADIUS proxy chain between Diameter translation agents, the route between translation agents remains stable during the session and does not cause an invalidation of the proxy path stack.

7. Security Considerations

The security considerations in [RFC3588] apply to this extension. In addition, this extension raises questions of authorization and can potentially allow a new denial-of-service attack.

The authorization issue comes about because the proxies that participate in ER are self-selected. An ER-Proxy is able, through the operation of ER, to guarantee that it can monitor every message of a session. This is in contrast to ordinary Diameter routing, where some messages may pass by an alternate route. The question is whether the originating party is prepared to extend this additional degree of trust to arbitrary parties along the path. If not, the ER-Originator requires a mechanism to determine whether an ER-Proxy listed in the returned Explicit-Path AVP can be trusted. If it has such a mechanism, then an unwanted ER-Proxy can be deleted from its cache and thus not appear in the ER-Path AVP in subsequent requests. This specification assumes that either the originating party is prepared to allow arbitrary Diameter nodes along the path to attach themselves to the session as ER-Proxies, or the ER-Originator maintains a pre-configured list of ER-Proxies in its cache.

The potential denial-of-service attack is not a serious one because the same result can be obtained more directly. An attacker with control of a Diameter node along the path of the original request could insert an Explicit-Path-Record containing the identity of another node or a non-existent node, rather than its own identity. Routing subsequent messages of the session through another node could result in violation of the trust assumptions made upstream. Routing subsequent messages to a non-existent node causes them to be lost and terminates the session. It would seem simpler to perpetrate whatever harm the attacker intends at the subverted Diameter node itself. The advantage of using ER to accomplish either of the attacks is that it makes it more difficult to determine which node misbehaved, but the extra effort involved to implement the attack does not seem to be worth the potential gain.

8. Acknowledgements

The authors gratefully acknowledge the contributions of Tony Zhang, Fortune Huang, Rajith R., Victor Fajardo, Jouni Korhonen, Tolga Asveren, Mark Jones, Avi Lior, Steve Norreys, Lionel Morand, Dave Frascone, and Hannes Tschofenig.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC5729] Korhonen, J., Ed., Jones, M., Morand, L., and T. Tsou, "Clarifications on the Routing of Diameter Requests Based on the Username and the Realm", RFC 5729, December 2009.

9.2. Informative References

[TS23.234] 3GPP, "3GPP system to Wireless Local Area Network (WLAN) interworking; System description", TS 23.234 Version 7.4.0, 2006.

Authors' Addresses

Tina Tsou
Huawei Technologies (USA)
2330 Central Expressway
Santa Clara, CA 95050
USA

Phone: +1 408 330 4424
EMail: tena@huawei.com
URI: http://tinatsou.weebly.com/contact.html

Glen Zorn
Network Zen
227/358 Thanon Sanphawut
Bang Na, Bangkok 10260
Thailand

Phone: +66 (0) 87-040-4617
EMail: gwz@net-zen.net

Tom Taylor (editor)
Huawei Technologies
1852 Lorraine Ave.
Ottawa
Canada

EMail: tom111.taylor@bell.net