       Diameter Support for the EAP Re-authentication Protocol (ERP)

Abstract

   The EAP Re-authentication Protocol (ERP) defines extensions to the
   Extensible Authentication Protocol (EAP) to support efficient
   re-authentication between the peer and an EAP Re-authentication (ER)
   server through a compatible authenticator. This document specifies
   Diameter support for ERP. It defines a new Diameter ERP application
   to transport ERP messages between an ER authenticator and the ER
   server, and a set of new Attribute-Value Pairs (AVPs) that can be
   used to transport the cryptographic material needed by the
   re-authentication server.

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 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6942.

Copyright Notice

Table of Contents

1. Introduction

    Cao, et al. [RFC6696] defines the EAP Re-authentication Protocol
    (ERP). It consists of the following steps:

    Bootstrapping

       A root key for re-authentication is derived from the Extended
       Master Session Key (EMSK) created during EAP authentication
       [RFC5295]. This root key is transported from the EAP server to
       the ER server.

    Re-authentication

       A one-round-trip exchange between the peer and the ER server,
       resulting in mutual authentication. To support the EAP
       re-authentication functionality, ERP defines two new EAP codes --
       EAP-Initiate and EAP-Finish.

    This document defines how Diameter transports the ERP messages during
    the re-authentication process. For this purpose, we define a new
    Application Identifier for ERP and reuse the Diameter EAP commands
    Diameter-EAP-Request (DER) / Diameter-EAP-Answer (DEA).

    This document also discusses the distribution of the root key during
    bootstrapping, in conjunction with either the initial EAP
    authentication (implicit bootstrapping) or the first ERP exchange
    (explicit bootstrapping). Security considerations for this key
    distribution are detailed in Section 7.4 of Salowey, et al.
    [RFC5295].

2. Terminology

    This document uses terminology defined in Aboba, et al. [RFC3748],
    Salowey, et al. [RFC5295], Cao, et al. [RFC6696], and Eronen, et al.
    [RFC4072].

    Following RFC 5295, the term "domain" herein refers to a key
    management domain unless otherwise qualified. Similarly, the terms
    "home domain" and "local domain" have the same meaning here as in RFC
    6696.

    The re-authentication Domain-Specific Root Key (rDSRK) is a
    re-authentication Root Key (rRK) [RFC6696] derived from the Domain-
    Specific Root Key (DSRK) instead of the EMSK.

"Root key" (RK) or "bootstrapping material" refers to the rRK or
rDSRK derived from an EMSK, depending on whether the ER server is
located in the home or a foreign domain.

We use the notation "ERP/DER" and "ERP/DEA" in this document to refer
to Diameter-EAP-Request and Diameter-EAP-Answer commands with the
Application Id set to <Diameter ERP> (Section 10.1); the same
commands are denoted "EAP/DER" and "EAP/DEA" when the Application Id
in the message is set to <Diameter EAP> [RFC4072].

2.1. Requirements Language

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].

3. Assumptions

This document assumes the existence of, at most, one logical ER
server entity in a given domain. If several physical servers are
deployed for robustness, a replication mechanism must be deployed to
synchronize the ERP state (e.g., root keys) between these servers.
Any such replication mechanism is outside the scope of this document.
If multiple ER servers are deployed in the domain, we assume that
they can be used interchangeably. If multiple ER servers are
deployed across multiple domains, we assume that only one ER server,
topologically close to the peer, is involved in ERP, with distance
being measured in terms of Diameter hops.

This document also assumes the existence of, at most, one EAP server
entity in the home domain. In case of multiple physical home EAP
servers, if the ER server wants to reach the same home EAP server,
the ER server SHOULD cache the Destination-Host AVP corresponding to
the home EAP server it requests.

In general, it is assumed that key management domain names and
Diameter realm names are identical for any given domain/realm.

4. Protocol Overview

   The following figure illustrates the components involved in ERP and
   their interactions.
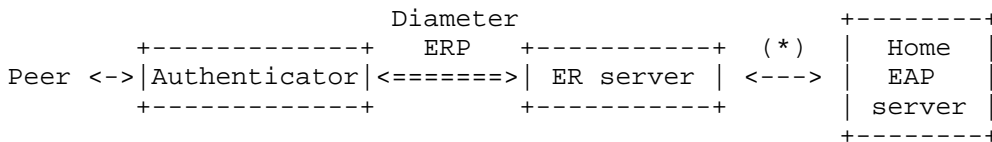
```
                        Diameter                +--------+
        +-------------+   ERP   +-----------+ (*) | Home   |
 Peer <->|Authenticator|<=======>| ER server | <---> | EAP    |
        +-------------+         +-----------+     | server |
                                                  +--------+
 (*) Diameter EAP application; explicit bootstrapping scenario only.
```

                   Figure 1: Diameter ERP Overview

   The ER server is located either in the home domain (same as the EAP
   server) or in the local domain (same as the authenticator, when it
   differs from the home domain).

   When the peer initiates an ERP exchange, the authenticator creates a
   DER message [RFC4072]. The Application Id of the message is set to
   that of the Diameter ERP application (Section 10.1) in the message.
   The generation of the ERP/DER message is detailed in Section 6.

   If there is an ER server in the same domain as the authenticator
   (i.e., the local domain), Diameter routing MUST be configured so that
   this ERP/DER message reaches that server, even if the Destination-
   Realm is not the same as the local domain.

   If there is no local ER server, the message is routed according to
   its Destination-Realm AVP content, extracted from the realm component
   of the keyName-NAI attribute. As specified in RFC 6696, this realm
   is the home domain of the peer in the case of bootstrapping exchange
   ('B' flag is set in ERP message) or the domain of the bootstrapped ER
   server otherwise.

   If no ER server is available in the home domain either, the ERP/DER
   message cannot be delivered and an error, DIAMETER_UNABLE_TO_DELIVER,
   MUST be generated, as specified in RFC 6733, and returned to the
   authenticator. The authenticator MAY cache this information (with
   limited duration) to avoid further attempts to execute ERP with this
   realm. It MAY also fallback to full EAP authentication to
   authenticate the peer.

   When an ER server receives the ERP/DER message, it searches its local
   database for a valid, unexpired root key matching the keyName part of
   the User-Name AVP. If such key is found, the ER server processes the

ERP message, as described in RFC 6696, then creates the ERP/DEA
answer, as described in Section 6. The re-authentication Master
Session Key (rMSK) is included in this answer.

Finally, the authenticator extracts the rMSK from the ERP/DEA, as
described in RFC 6696, and forwards the content of the EAP-Payload
AVP, the EAP-Finish/Re-auth message, to the peer.

The ER server may or may not possess the root key in its local
database. If the EAP-Initiate/Re-auth message has its 'B' flag set
(bootstrapping exchange) and the ER server possesses the root key,
the ER server SHOULD respond directly to the peer that initiated the
ERP exchange. Otherwise, the ER server SHOULD act as a proxy and
forward the message to the home EAP server after changing its
Application Id to Diameter EAP and adding the ERP-RK-Request AVP to
request the root key. See Section 5 for more detail on this process.

5. Bootstrapping the ER Server

The bootstrapping process involves the home EAP server and the ER
server, but also impacts the peer and the authenticator. In ERP, the
peer must derive the same keying material as the ER server. To
achieve this, it must learn the domain name of the ER server. How
this information is acquired is outside the scope of this
specification, but the authenticator might be configured to advertise
this domain name, especially in the case of re-authentication after a
handover.

The bootstrapping of an ER server with a given root key happens
either during the initial EAP authentication of the peer when the
EMSK -- from which the root key is derived -- is created, during the
first re-authentication, or sometime between those events. We only
consider the first two possibilities in this specification, in the
following subsections.

5.1. Bootstrapping during the Initial EAP Authentication

Bootstrapping the ER server during the initial EAP authentication
(also known as implicit bootstrapping) offers the advantage that the
server is immediately available for re-authentication of the peer,
thus minimizing the re-authentication delay. On the other hand, it
is possible that only a small number of peers will use
re-authentication in the local domain. Deriving and caching key
material for all the peers (for example, for the peers that do not
support ERP) is a waste of resources and should be avoided.

   To achieve implicit bootstrapping, the ER server acts as a Diameter
   EAP Proxy, and Diameter routing MUST be configured so that Diameter
   EAP application messages are routed through this proxy. The figure
   below illustrates this mechanism.

```
                               ER server &
   Authenticator              EAP Proxy                  Home EAP server
   =============              ==========                 ==============
        ------------------------>
           Diameter EAP/DER
            (EAP-Response)

                                ------------------------>
                                   Diameter EAP/DER
                                    (EAP-Response)
                                    (ERP-RK-Request)


        <===================================================>
          Multi-round Diameter EAP exchanges, unmodified


                                <------------------------
                                   Diameter EAP/DEA
                                    (EAP-Success)
                                        (MSK)
                                    (Key AVP (rRK))
        <------------------------
           Diameter EAP/DEA
            (EAP-Success)
                (MSK)
             [ERP-Realm]
```
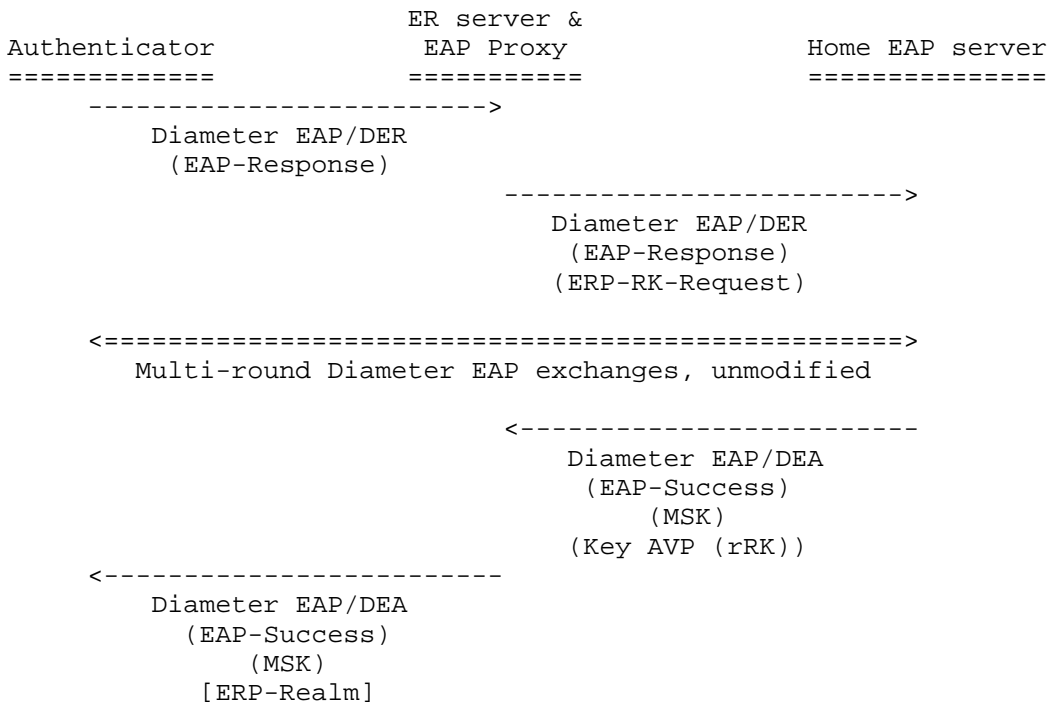
        Figure 2: ERP Bootstrapping during Full EAP Authentication

   The authenticator creates the first DER of the full EAP
   authentication and sends it to the ER server. The ER server proxies
   the first DER of the full EAP authentication and adds the
   ERP-RK-Request AVP inside, then forwards the request to the home EAP
   server.

   If the home Diameter server does not support the Diameter ERP
   extensions, it simply ignores the ERP-RK-Request AVP and continues as
   specified in RFC 4072 [RFC4072]. If the server supports the ERP
   extensions, it saves the value of the ERP-Realm AVP found inside the
   ERP-RK-Request AVP, and continues with the EAP authentication. When
   the authentication completes, if it is successful and the EAP method
   has generated an EMSK, the server MUST derive the rRK as specified in
   RFC 6696, using the saved ERP realm name. It then includes the rRK
   inside a Key AVP (Section 8.3) with the Key-Type AVP set to rRK,
   before sending the DEA as usual.

When the ER server proxies a Diameter-EAP-Answer message with a
Session-Id corresponding to a message to which it added an
ERP-RK-Request AVP, and the Result-Code is DIAMETER_SUCCESS, it MUST
examine the message and save and remove any Key AVP (Section 8.3)
with Key-Type AVP set to rRK. If the message does not contain such a
Key AVP, the ER server may cache the information that
re-authentication via ERP is not possible for the session in order to
avoid any subsequent attempts. In any case, the information stored
in the ER server concerning a session should not have a lifetime
greater than the EMSK for this session.

If the ER server is successfully bootstrapped, it should also add the
ERP-Realm AVP after removing the Key AVP with Key-Type of rRK in the
EAP/DEA message. This ERP-Realm information can be used by the
authenticator to notify the peer that the ER server is bootstrapped,
and for which domain. How this information can be transmitted to the
peer is outside the scope of this document. This information needs
to be sent to the peer if both implicit and explicit bootstrapping
mechanisms are possible, because the ERP message and the root key
used for protecting this message are different in bootstrapping
exchanges and non-bootstrapping exchanges.

5.2. Bootstrapping during the First Re-authentication

Bootstrapping the ER server during the first re-authentication (also
known as explicit bootstrapping) is only needed when there is no ER
server in the local domain and there is an ER server in the home
domain. It is less resource intensive, since the EMSK generated
during initial EAP authentication is reused to derive root keys. On
the other hand, the first re-authentication requires a one-round-trip
exchange with the home EAP server, since the EMSK is generated during
the initial EAP authentication and never leaves the home EAP server,
which is less efficient than implicit bootstrapping.

The EAP-Initiate/Re-auth message is sent to the home ER server. The
home ER server receives the ERP/DER message containing the
EAP-Initiate/Re-auth message with the 'B' flag set. It creates the
new EAP/DER message using the received ERP/DER message and performs
the following processing:

    Set the Application Id in the header of the message to
    <Diameter EAP> [RFC4072].

    Extract the ERP-RK-Request AVP from the ERP/DER message, which
    contains the name of the domain where the ER server is located,
    and add it to the newly created ERP/DER message.

Then, the newly created EAP/DER is sent and routed to the home
Diameter EAP application server.

If the home Diameter EAP server does not support ERP extensions, EAP
packets with an unknown ERP-specific code (EAP-Initiate) will not be
understood. In such a case, the home Diameter EAP server MUST send
an EAP/DEA with a Result-Code indicating a Permanent Failure (for
example, DIAMETER_ERROR_EAP_CODE_UNKNOWN or
DIAMETER_UNABLE_TO_COMPLY). The Failed-AVP AVP MUST be included and
contain a copy of the EAP-Payload AVP. Otherwise, it processes the
DSRK request, as described in RFC 6696. In particular, it includes
the Domain-Name TLV attribute with the content from the ERP-Realm
AVP. The server creates the EAP/DEA reply message [RFC4072],
including an instance of the Key AVP (Section 8.3) with the Key-Type
AVP set to rRK and an instance of the Domain-Name TLV attribute with
the content from the ERP-Realm AVP.

The ER server receives this EAP/DEA and proxies it as follows, in
addition to standard proxy operations:

    Set the Application Id back to Diameter ERP Application Id
    (Section 10.1).

    Extract and cache the content of the Key AVP with Key-Type set to
    rRK, as described in Section 5.1).

The ERP/DEA message is then forwarded to the authenticator that can
use the rMSK as described in RFC 6696.

The figure below captures this proxy behavior:

```
Authenticator                 ER server            Home Diameter server
=============                 =========            ===================
        ---------------------->
           Diameter ERP/DER
             (EAP-Initiate)
                                  ------------------------>
                                       Diameter EAP/DER
                                         (EAP-Response)
                                         (ERP-RK-Request)

                                  <------------------------
                                       Diameter EAP/DEA
                                         (EAP-Success)
                                        (Key AVP (rRK))
                                        (Key AVP (rMSK))
        <----------------------
           Diameter ERP/DEA
             (EAP-Finish)
            (Key AVP (rMSK))

        Figure 3: ERP Explicit Bootstrapping Message Flow
```
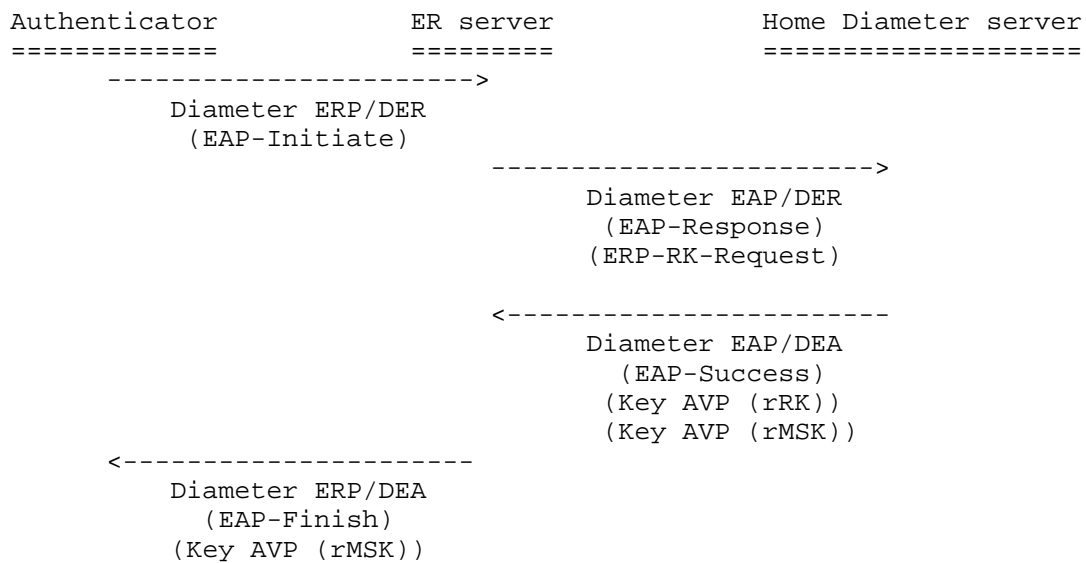
6. Re-authentication

   This section describes in detail a re-authentication exchange with an
   ER server that was previously bootstrapped. The following figure
   summarizes the re-authentication exchange.

```
                                                  ER server
    Peer                   Authenticator         (bootstrapped)
    ====                   =============       =====================
    [ <----------------------            ]
    [optional EAP-Initiate/Re-auth-start,]
    [   possibly with ERP domain name    ]

      ---------------------->
         EAP-Initiate/Re-auth

                            ===============================>
                               Diameter ERP, cmd code DER
                                  User-Name: keyName-NAI
                              EAP-Payload: EAP-Initiate/Re-auth


                            <===============================
                               Diameter ERP, cmd code DEA
                              EAP-Payload: EAP-Finish/Re-auth
                                     Key AVP: rMSK
      <----------------------
         EAP-Finish/Re-auth
```

           Figure 4: Diameter ERP Re-authentication Exchange

   The peer sends an EAP-Initiate/Re-auth message to the ER server via
   the authenticator. Alternatively, the authenticator may send an
   EAP-Initiate/Re-auth-Start message to the peer to trigger the
   mechanism. In this case, the peer responds with an
   EAP-Initiate/Re-auth message.

   If the authenticator does not support ERP (pure Diameter EAP
   [RFC4072] support), it discards the EAP packets with an unknown ERP-
   specific code (EAP-Initiate). The peer should fall back to full EAP
   authentication in this case.

   When the authenticator receives an EAP-Initiate/Re-auth message from
   the peer, the message is processed as described in RFC 6696, with
   regard to the EAP state machine. It creates a Diameter ERP/DER
   message following the general process of Diameter EAP [RFC4072], with
   the following differences:

   The Application Id in the header is set to <Diameter ERP>
   (code 13).

   The value in the Auth-Application-Id AVP is also set to
   <Diameter ERP>.

   The keyName-NAI attribute from the ERP message is used to create
   the content of the User-Name and Destination-Realm AVPs.

   The Auth-Request-Type AVP content is set to the appropriate value.

   The EAP-Payload AVP contains the EAP-Initiate/Re-auth message.

   Then, this ERP/DER message is sent as described in Section 4.

   The ER server receives and processes this request as described in
   Section 4. It then creates an ERP/DEA message following the general
   process described in Eronen, et al. [RFC4072], with the following
   differences:

   The Application Id in the header is set to <Diameter ERP>
   (code 13).

   The value of the Auth-Application-Id AVP is also set to
   <Diameter ERP>.

   The EAP-Payload AVP contains the EAP-Finish/Re-auth message.

   If authentication is successful, an instance of the Key AVP
   containing the rMSK derived by ERP is included.

   When the authenticator receives this ERP/DEA answer, it processes it
   as described in the Diameter EAP Application specification [RFC4072]
   and RFC 6696: the content of the EAP-Payload AVP is forwarded to the
   peer, and the contents of the Keying-Material AVP [RFC6734] is used
   as a shared secret for a secure association protocol specific to the
   lower layer in use.

7. Application Id

   We define a new Diameter application in this document, Diameter ERP,
   with an Application Id value of 13. Diameter nodes conforming to
   this specification in the role of the ER server MUST advertise
   support by including an Auth-Application-Id AVP with a value of
   Diameter ERP in the Capabilities-Exchange-Request and
   Capabilities-Exchange-Answer commands [RFC6733].

The primary use of the Diameter ERP Application Id is to ensure
proper routing of the messages, and that the nodes that advertise the
support for this application do understand the new AVPs defined in
Section 8, although these AVPs have the 'M' flag cleared.

8. AVPs

   The following subsections discuss the AVPs used by the Diameter ERP
   application.

8.1. ERP-RK-Request AVP

   The ERP-RK-Request AVP (AVP Code 618) is of type Grouped AVP. This
   AVP is used by the ER server to indicate its willingness to act as
   the ER server for a particular session.

   This AVP has the 'M' and 'V' bits cleared.

        ERP-RK-Request ::= < AVP Header: 618 >
                             { ERP-Realm }
                           * [AVP]

                     Figure 5: ERP-RK-Request ABNF

8.2. ERP-Realm AVP

   The ERP-Realm AVP (AVP Code 619) is of type DiameterIdentity. It
   contains the name of the realm in which the ER server is located.

   This AVP has the 'M' and 'V' bits cleared.

8.3. Key AVP

   The Key AVP [RFC6734] is of type Grouped and is used to carry the rRK
   or rMSK and associated attributes. The usage of the Key AVP and its
   constituent AVPs in this application is specified in the following
   subsections.

8.3.1. Key-Type AVP

   The value of the Key-Type AVP MUST be set to 1 for rRK or 2 for rMSK.

8.3.2. Keying-Material AVP

   The Keying-Material AVP contains the rRK sent by the home EAP server
   to the ER server, in answer to a request containing an ERP-RK-Request
   AVP, or the rMSK sent by the ER server to the authenticator. How
   this material is derived and used is specified in RFC 6696.

8.3.3. Key-Name AVP

   This AVP contains the EMSKname that identifies the keying material.
   The derivation of this name is specified in RFC 6696.

8.3.4. Key-Lifetime AVP

   The Key-Lifetime AVP contains the lifetime of the keying material in
   seconds. It MUST NOT be greater than the remaining lifetime of the
   EMSK from which the material was derived.

9. Result-Code AVP Values

   This section defines new Result-Code [RFC6733] values that MUST be
   supported by all Diameter implementations that conform to this
   specification.

9.1. Permanent Failures

   Errors that fall within the Permanent Failures category are used to
   inform the peer that the request failed and SHOULD NOT be attempted
   again.

      DIAMETER_ERROR_EAP_CODE_UNKNOWN (5048)

         This error code is used by the Diameter server to inform the
         peer that the received EAP-Payload AVP contains an EAP packet
         with an unknown EAP code.

10. IANA Considerations

   IANA has registered the following new elements in the Authentication,
   Authorization, and Accounting (AAA) Parameters registries
   [AAAPARAMS].

10.1. Diameter Application Identifier

   IANA has allocated a new value "Diameter ERP" (code: 13) in the
   "Application IDs" registry from the "Standards Action" range of
   numbers using the "Specification Required" policy [RFC5226]; see
   Section 11.3 of RFC 3588 [RFC3588] for further details.

10.2. New AVPs

   IANA has allocated new values from the "AVP Codes" registry according
   to the policy specified in Section 11.1 of Fajardo, et al. [RFC6733]
   for the following AVPs:

      ERP-RK-Request (code: 618)

      ERP-Realm (code: 619)

   These AVPs are defined in Section 8.

10.3. New Permanent Failures Result-Code AVP Values

   IANA has allocated a new value from the "Result-Code AVP Values (code
   268) - Permanent Failure" registry according to the policy specified
   in Section 11.3.2 of Fajardo, et al. [RFC6733] for the following
   Result-Code:

      DIAMETER_ERROR_EAP_CODE_UNKNOWN (code: 5048)

   This Result-Code value is defined in Section 9.

11. Security Considerations

   The security considerations from the following documents apply here:

   o Eronen, et al. [RFC4072]

   o Salowey, et al. [RFC5295]

   o Cao, et al. [RFC6696]

   o Fajardo, et al. [RFC6733]

   o Zorn, et al. [RFC6734]

   Because this application involves the transmission of sensitive data,
   including cryptographic keys, it MUST be protected using Transport
   Layer Security (TLS) [RFC5246], Datagram Transport Layer Security
   (DTLS) [RFC6347], or IP Encapsulating Security Payload (ESP)
   [RFC4303]. If TLS or DTLS is used, the bulk encryption algorithm
   negotiated MUST be non-null. If ESP is used, the encryption
   algorithm MUST be non-null.

12. Contributors

   Hannes Tschofenig wrote the initial draft of this document.

   Lakshminath Dondeti contributed to the early drafts of the document.

13. Acknowledgements

   Hannes Tschofenig, Zhen Cao, Benoit Claise, Elwyn Davies, Menachem
   Dodge, Vincent Roca, Stephen Farrell, Sean Turner, Pete Resnick, Russ
   Housley, Martin Stiemerling, and Jouni Korhonen provided useful
   reviews.

   Vidya Narayanan reviewed a rough draft version of the document and
   found some errors.

   Many thanks to these people!

14. References

14.1. Normative References

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

   [RFC3748]   Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H.
                Levkowetz, "Extensible Authentication Protocol (EAP)",
                RFC 3748, June 2004.

   [RFC4072]   Eronen, P., Hiller, T., and G. Zorn, "Diameter Extensible
                Authentication Protocol (EAP) Application", RFC 4072,
                August 2005.

   [RFC5226]   Narten, T. and H. Alvestrand, "Guidelines for Writing an
                IANA Considerations Section in RFCs", BCP 26, RFC 5226,
                May 2008.

   [RFC5295]   Salowey, J., Dondeti, L., Narayanan, V., and M. Nakhjiri,
                "Specification for the Derivation of Root Keys from an
                Extended Master Session Key (EMSK)", RFC 5295, August
                2008.

   [RFC6696]   Cao, Z., He, B., Shi, Y., Wu, Q., and G. Zorn, "EAP
                Extensions for the EAP Re-authentication Protocol (ERP)",
                RFC 6696, July 2012.

   [RFC6733]   Fajardo, V., Arkko, J., Loughney, J., and G. Zorn,
                "Diameter Base Protocol", RFC 6733, October 2012.

   [RFC6734]   Zorn, G., Wu, Q., and V. Cakulev, "Diameter Attribute-
               Value Pairs for Cryptographic Key Transport", RFC 6734,
               October 2012.

14.2. Informative References

   [AAAPARAMS] Internet Assigned Numbers Authority, "Authentication,
               Authorization, and Accounting (AAA) Parameters",
               <http://www.iana.org/assignments/aaa-parameters/>.

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

   [RFC4303]   Kent, S., "IP Encapsulating Security Payload (ESP)", RFC
               4303, December 2005.

   [RFC5246]   Dierks, T. and E. Rescorla, "The Transport Layer Security
               (TLS) Protocol Version 1.2", RFC 5246, August 2008.

   [RFC6347]   Rescorla, E. and N. Modadugu, "Datagram Transport Layer
               Security Version 1.2", RFC 6347, January 2012.

Authors' Addresses

   Julien Bournelle
   Orange Labs
   38-40 rue du general Leclerc
   Issy-Les-Moulineaux 92794
   France

   EMail: julien.bournelle@orange.com

   Lionel Morand
   Orange Labs
   38-40 rue du general Leclerc
   Issy-Les-Moulineaux 92794
   France

   EMail: lionel.morand@orange.com

   Sebastien Decugis
   INSIDE Secure
   41 Parc Club du Golf
   Aix-en-Provence 13856
   France

   Phone: +33 (0)4 42 39 63 00
   EMail: sdecugis@freediameter.net

   Qin Wu
   Huawei Technologies Co., Ltd.
   101 Software Avenue, Yuhua District
   Nanjing, JiangSu 210012
   China

   EMail: sunseawq@huawei.com

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

   EMail: glenzorn@gmail.com