Diameter Maintenance and
Extensions (DIME)
Internet-Draft
Intended status: Standards Track
Expires: December 10, 2009

S. Decugis, Ed.
NICT

June 8, 2009

Diameter support for EAP Re-authentication Protocol (ERP)
draft-sdecugis-dime-diameter-erp-01

Status of This Memo

Copyright Notice

Abstract

The EAP Re-authentication Protocol (ERP) provides a mechanism to optimize EAP authentication delay in the case of re-authentication,

which can be significant in roaming mobile situation. This mechanism assumes that a protocol for Authentication, Authorization and Accounting (AAA) is available to transport ERP between the authenticator(s) and the EAP/ERP server. draft-gaonkar-radext-erp-attrs-03 specifies the transport of ERP using RADIUS. This document specifies the transport of ERP using Diameter.

Table of Contents

1. Introduction

[RFC5296] defines the EAP Re-authentication Protocol (ERP) and mechanism that consists in the two following steps:

1. Bootstrapping: creation of a root key for re-authentication, after initial EAP authentication of the peer. This root key is distributed from the EAP server to the ER server. How this key is tranported is not specified in the ERP mechanism.

2. Re-authentication: one-round-trip exchange between the peer and the ER server, functionally equivalent to a full EAP authentication.

This document specifies how Diameter is used to carry the re-authentication exchange (second step). For this purpose, we define a new Application Id for ERP, and re-use the Diameter EAP commands (DER/DEA).

We also discuss the key distribution (first step, bootstrapping) and propose some solutions for different architectures. Anyway, implementors are free to choose a different mechanism for key distribution, as for example using RADIUS [I-D.ietf-hokey-key-mgm]. Security considerations for key distribution are explained in [RFC5295].

1.1. Differences with other documents

This document differs from [I-D.ietf-dime-erp] in its design and scope. In this new version, we use a new Diameter application id for messages with ERP payload exchanged between authenticator and ER server. This simplifies the routing of Diameter messages to the appropriate server, and allows more flexibility in the deployment of ERP.

The scope of previous documents ([I-D.ietf-dime-erp] and

[I-D.wu-dime-local-keytran]) was focused on the bootstrapping of the mechanism. In particular, these documents did not consider the routing of Diameter message for re-authentication exchanges (ERP exchange, and also [RFC4187] for the second document). By re-using the Diameter EAP application, they create implicit constraints on routing of messages that cannot be met by standard Diameter routing algorithm, as defined in the Diameter Base Protocol [RFC3588].

A separate Diameter application solves this routing issue, and can also allow the authenticator to dynamically discover if the local domain supports re-authentication or not.

2. Terminology

We re-use in this document the terminology from [RFC5296]. In particular, unless specified otherwise, the EAP server has implicit support for ERP extensions for generation of ERP keying material and its transmission to ER server. These terms "authenticator", "ER server", "EAP server" designate logical functional entities and make no assumption on the real implementation and deployment.

"Root key" (RK) or "bootstrapping material" refer to the rRK or rDSRK derived from an EMSK, depending on the location of the ER server in home or foreign domain.

We re-use also some terminology and abbreviations from [RFC4072], for example DER/DEA.

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. Overview

During the lifetime of an EMSK (derived during a full EAP authentication by compatible EAP methods), a peer may attach to several authenticators. In this case, re-authentication is more efficient than full authentication, especially in the case of roaming. ERP provides a mechanism for re-authentication independent

of the link layer, so it can be used in case of multihoming or handovers between different access technologies. The following figure shows the components involved in ERP, and their interactions. When the peer attaches to a new authenticator, the ER server involved in the transaction may change, for example in the case of inter-domain roaming. The home EAP server is assumed to be constant for a given peer.

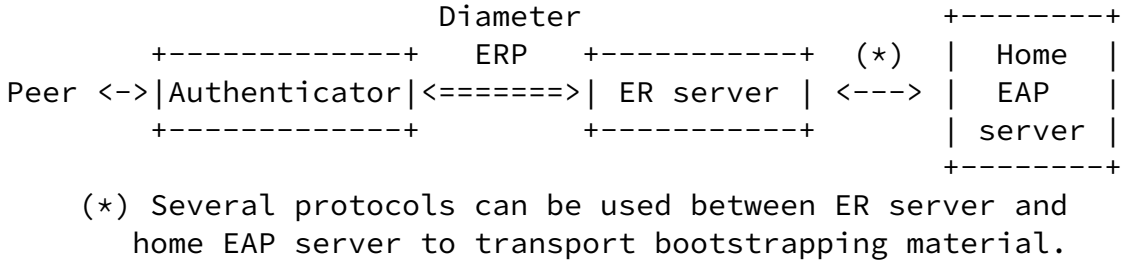
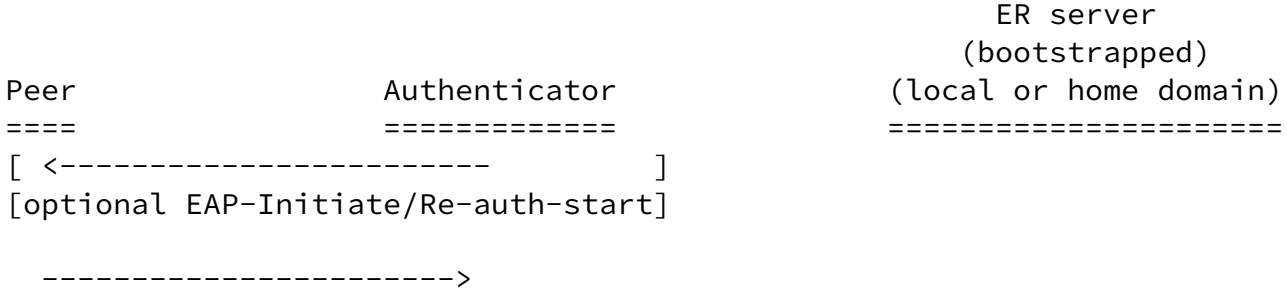


Figure 1. Diameter applications used in the ERP mechanism.

The ER server may be located in the home domain (same as EAP server)

or the visited domain (same as authenticator, when it differs from the home domain). [[Editor1: Can the ER server be located in a third domain (ex: broker's)?]]

The bootstrapping of the ER server has to occur sometime between the initial EAP authentication and the first ERP re-authentication with this ER server. See section Section 7 for detail on this process. Then, the peer re-authenticates, for example after a movement that makes it attach to a new authenticator. The following figure describes this re-authentication, and shows how Diameter is used in this context. See section Section 8 for a detailed description, and following sections for details on the Diameter messages format.



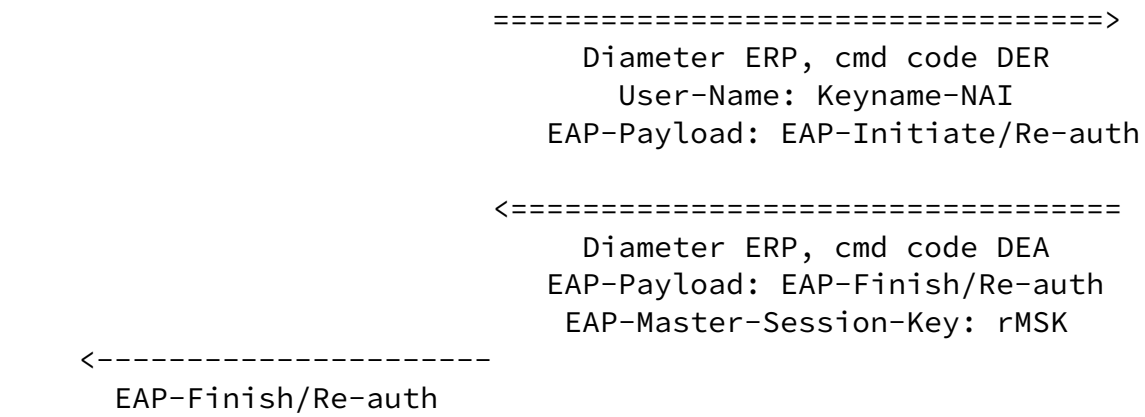


Figure 2. Diameter ERP exchange.

4. Application Id

We define a Diameter ERP Application in this document, with an Application Id value of [[IANA1: TBD]]. Diameter nodes conforming to this specification (in the role of ER server or authenticator) MUST advertise support by including the Diameter ERP Application ID value in the Auth-Application-Id AVP of the Capabilities-Exchange-Request and Capabilities-Exchange-Answer command [RFC3588].

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

the next section (although these AVP have the 'M' flag cleared).

5. AVPs

This specification defines the following new AVPs.

5.1. ERP-RK-Request AVP

The ERP-RK-Request AVP (AVP Code [[IANA2: TBD]]) is of type grouped AVP. It is used by the ER server to request root key material used in ERP.

This AVP has the M and V bits cleared.

```
ERP-RK-Request ::= < AVP Header: TBD >
```

```
                    { ERP-Realm }
                  * [ AVP ]
```

Figure 3. ERP-RK-Request ABNF

5.2. ERP-Realm AVP

The ERP-Realm AVP (AVP Code [[IANA3: TBD]]) is of type [[Editor2: DiameterIdentity? OctetString?]]. It contains the name of the realm in which the ER server is located.

[[Editor3: FFS: We may re-use Origin-Realm here instead? On the other hand, ERP-Realm may be useful in CER/CEA with a NAS...]]

This AVP has the M and V bits cleared.

5.3. ERP-RK-Answer AVP

The ERP-RK-Answer AVP (AVP Code [[IANA4: TBD]]) is of type grouped AVP. It is used by the home EAP server to provide ERP root key material to the ER server.

This AVP has the M and V bits cleared.

```
       ERP-RK-Answer ::= < AVP Header: TBD >
                         { ERP-RK }
                         { ERP-RK-Name }
                         { ERP-RK-Lifetime }
                       * [ AVP ]
```

Figure 4. ERP-RK-Answer ABNF

5.4. ERP-RK AVP

The ERP-RK AVP (AVP Code [[IANA5: TBD]]) is of type OctetString. It contains the root key (either rRK or rDSRK) to be used for ERP with the peer to which the current session belongs. How this material is derived and used is specified in [RFC5296].

[[Editor4: Can we re-use EAP-Master-Session-Key here?]]

This AVP has the M and V bits cleared.

5.5. ERP-RK-Name AVP

The ERP-RK AVP (AVP Code [[IANA6: TBD]]) is of type OctetString. This AVP contains the EMSKname which identifies the keying material. How this name is derived is beyond the scope of this document and defined in [RFC5296].

[[Editor5: Can we re-use EAP-Key-Name here?]]

This AVP has the M and V bits cleared.

5.6. ERP-RK-Lifetime AVP

The ERP-RK-Lifetime AVP (AVP Code [[IANA7: TBD]]) is of type [[Editor6: Unsigned64? 32?]] and contains the root key material remaining lifetime in seconds. It MUST not be greater than the remaining lifetime of the EMSK it is derived from. [[Editor7: FFS: is it better to pass an absolute value here, for example expiration date? How to express it then (TZ, ...)? Synchronization problems?]]

This AVP has the M and V bits cleared.

6. Commands

We do not define any new command in this specification. We reuse the Diameter-EAP-Request and Diameter-EAP-Answer commands defined in [RFC4072].

The Application Id field in the command header [[Editor8: and the value in Auth-Application-Id AVP which is redundant???]] can be set to Diameter EAP application or Diameter ERP application, depending on the situation, as explained in the next sections.

Since the original ABNF of these commands allow other optional AVPs ("* [AVP]"), and the new AVPs defined in this specification do not have the 'M' flag set, the ABNF does not need any change. Anyway, a Diameter node that advertize support for the Diameter ERP application

MUST support the ERP-RK-Request and ERP-RK-Answer AVP [[Editor9:

Therefore, in DER/DEA with Diameter ERP application ID, do we set the 'M' flag to these AVPs?]].

Command-Name	Abbrev.	Code	Reference	Application
Diameter-EAP-Request	DER	268	RFC 4072	Diameter ERP
Diameter-EAP-Answer	DEA	268	RFC 4072	Diameter ERP

Figure 5. Command Codes

7. Bootstrapping options

Bootstrapping involves the ER server and the EAP server directly, but also indirectly the peer and the authenticator. For ERP to be successful, the peer must derive the same keying material as the ER server. To make this possible, it must learn the domain name of the ER server. How this is achieved is outside the scope of this specification, but it usually involves that the authenticator is configured to advertize this domain name. This could be achieved for example by including the ERP-Realm AVP in a CER/CEA exchange.

As stated in the Section 3, the bootstrapping of an ER server has to happen between the initial EAP authentication of the peer, when the EMSK is created, and the moment the peer re-authenticates with this ER server, when the bootstrapping material is needed. While asynchrounous solutions are perfectly possible, it is usually easier to bootstrap the ER server during one of these events.

7.1. Bootstrapping during initial EAP authentication

Bootstrapping an ER server during the initial EAP authentication offers the advantage that the server is immediatly available for re-authentication of the peer, thus minimizing the re-authentication delay.

On the other hand, re-authentication may only concern a small number of peers in the visited domain. Deriving and caching key material for all the peers (for example, for the peers that do not support ERP, or that are not mobile) is a waste of resources and SHOULD be avoided. In addition, bootstrapping ERP during full EAP authentication may prevent re-authentication in case of inter-domain roaming. Hence, while this mecanism is useful in some situations, it should be deployed with care.

In the case where ER server is collocated with the Home EAP server, ER bootstrapping is transparent with regards to this specification, although some sort of communication might be needed inside the node.

In this case, the server MUST advertise support of both the Diameter EAP application and the Diameter ERP application, but the new AVPs defined in this specification are not used.

When ER server and EAP server are different entities with regards to Diameter, one or more Diameter EAP proxy(ies) is(are) needed in the same domain as the ER server. While this(these) proxy(ies) might be separate entity(ies) with secure communication channel with the ER server, it is functionally equivalent to consider that the ER server acts as a Diameter EAP proxy. In the rest of this section, we consider that the ER server acts as a Diameter EAP proxy in its domain.

In order to bootstrap the ER server during full EAP authentication, this server must be on the route, and act as a proxy, for the first and last round of exchanges of the full EAP authentication, as captured in the figure bellow.

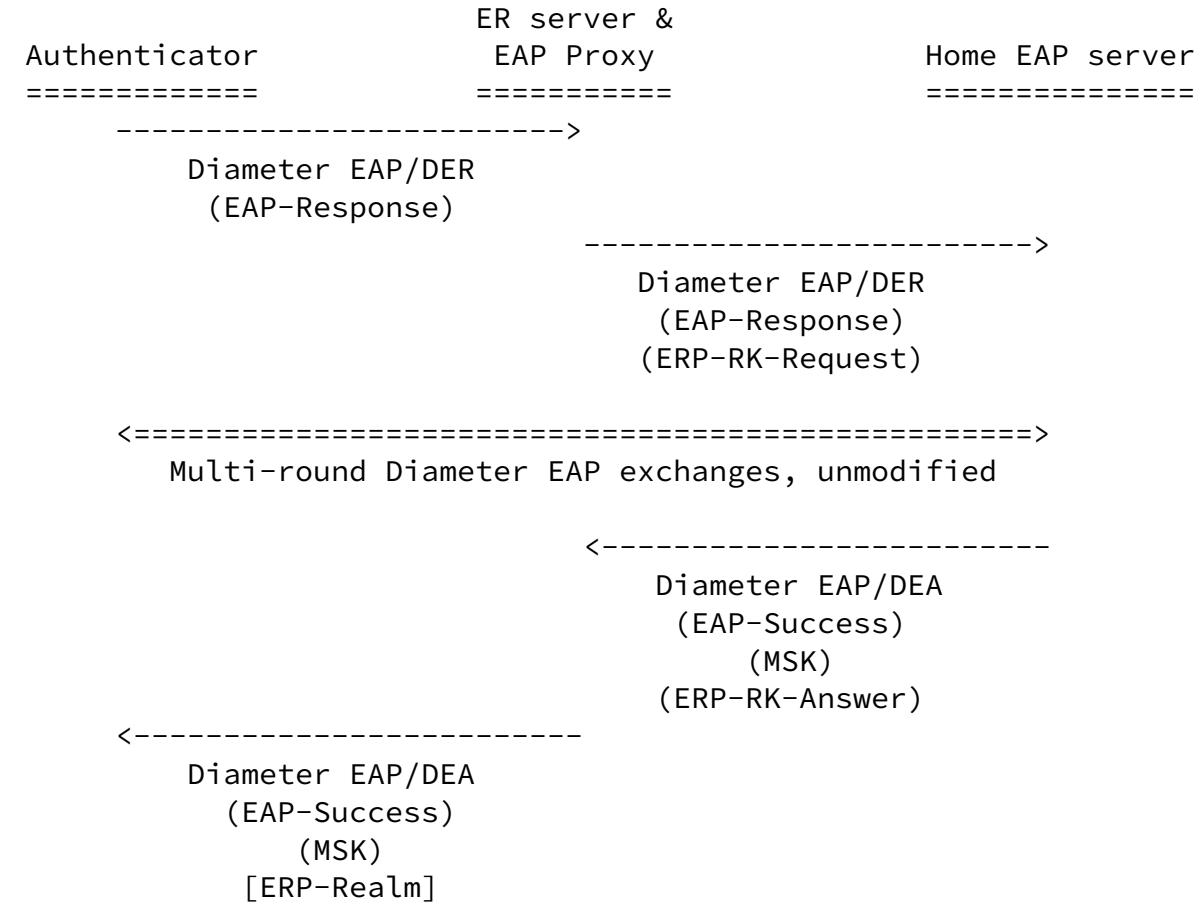


Figure 6. ERP bootstrapping during full EAP authentication

The ER server proxies the first DER of the full EAP authentication

and adds the ERP-RK-Request AVP inside, if this AVP is not already in the message, then forwards the request.

If the EAP server does not support ERP extensions, it will simply ignore this grouped AVP and continue as specified in [RFC4072]. If the server supports the ERP extensions, it caches the ERP-Realm value with the session, and continues the EAP authentication. When the authentication is complete, if it is successful and the EAP method generated an EMSK, the server MUST compute the rRK or rDSRK (depending on the value of ERP-Realm) as specified in [RFC5296], and add an ERP-RK-Answer AVP in the Diameter-EAP-Request message, in addition to the MSK and EAP-Success payload. [[Editor10: FFS: is it important to check that the server that added the ERP-RK-Request is in the path of the answer? What's the worst that can happen?]]

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-Answer, and the Result-Code is DIAMETER_SUCCESS, it MUST examine the message and remove any ERP-RK-Answer AVP, and save its content. If the message does not contain an ERP-RK-Answer AVP, the ER server MAY save this information to avoid possible attempts later for this session. In any case, the information stored SHOULD NOT have a lifetime greater than the EMSK lifetime [[Editor11: FFS: how does the ER server knows the EMSK lifetime, if there is no ERP-RK-Answer? What is the lifetime of the MSK for example?]]

If the ER server is successfully bootstrapped, it MAY also add the ERP-Realm AVP after removing the ERP-RK-Answer AVP in the Diameter-EAP-Answer message. This could be used by the authenticator to notify the peer that ERP is bootstrapped, with the ER domain information. How this information can be transmitted to the peer is outside the scope of this document. [[Editor12: Is it possible? It would be useful...]]

7.2. Bootstrapping during first re-authentication

Bootstrapping the ER server during the first re-authentication offers several advantages: it saves resources, since we generate and cache only useful keying material, it can also accomodate inter-domain roaming or ER servers that loose their state (for example after reboot).

On the other hand, the first re-authentication with the ER server requires a one-round-trip with the home EAP server, which adds some delay to the process (but it is more efficient than a full EAP authentication in any case). Note that following re-authentications for the same session with the same ER server will not have this additional delay.

[RFC5296] describes two types of bootstrapping for ERP: implicit bootstrapping and explicit bootstrapping. In implicit bootstrapping,

the peer knows the domain it is located in, and assumes that the ER server already possess the keying material for the session. In this case, the peer uses a Keyname-NAI in the form "EMSKname@localdomain". In explicit bootstrapping, the Keyname-NAI is in the form "EMSKname@homedomain". As we will see in next section Section 8, the domain part of the Keyname-NAI becomes the Destination-Realm of the Diameter message, and the Application Id is set to Diameter ERP application.

In the case of implicit bootstrapping (how the peer learns that the ER server is bootstrapped is outside the scope of this specification) or after a first succesful re-authentication in the visited domain, the message is routed to the local ER server following normal Diameter routing. If the ER server does not have key information corresponding to this EMSKname, [[Editor13: return an error to the peer? proxy the request and send ERP-RK-Request to the home EAP server? How do we learn which is the home domain?]]. See the next section Section 8 for detail.

In the case of explicit bootstrapping (the ERP message has its 'B' flag set), if an ER server exists in the visited domain, it SHOULD be configured for and act as a Diameter ERP proxy, and process the messages as described below. If not, the ER server in the home domain will be used, which is less efficient. The description that follow for the ER server in the visited domain is also valid for the ER server in the home domain.

[[Editor14: What should we do if the ER server receives an explicit bootstrapping request but already possess the rDSRK?]]

The ER server proxies the request (DER with Diameter ERP application code) as follow, in addition to standard proxy operations:

Change the Application Id in the header of the message to Diameter EAP Application (code 5). [[Editor15: What about the Application-Auth-Id AVP?]]

Add the ERP-RK-Request AVP, which contains the name of the domain the ER server is located in (with regards to ERP).

Then the request is forwarded as usual. With its Diameter EAP application id and Destination-Realm set to the home domain of the peer, the request reaches the home EAP server. If this server does not support ERP extensions, it replies with an error since the encapsulated EAP-Initiate/Re-auth command is not understood. Otherwise, it processes the ERP request as described in [RFC5296]. In particular, it includes the Domain-Name TLV attribute with the content from the ERP-Realm AVP. It creates the DEA reply message

following standard processing from [RFC4072] (in particular EAP-Master-Session-Key AVP is used to transport the rMSK), and includes the ERP-RK-Answer AVP.

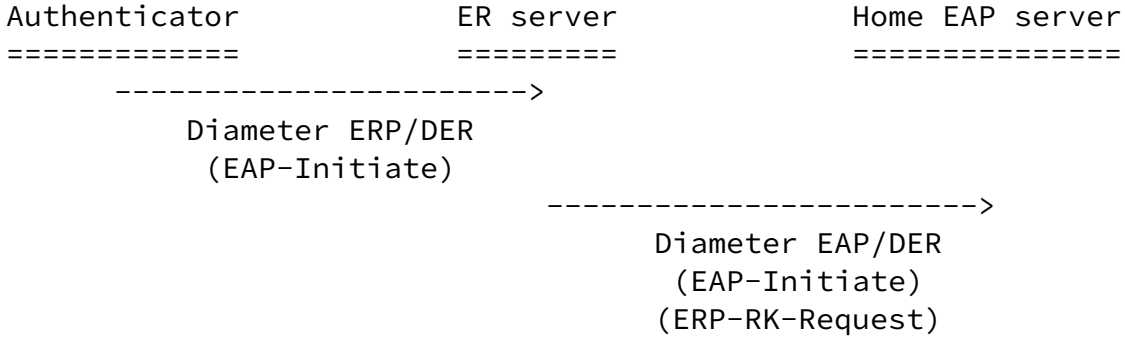
The ER server receives this DEA and proxies it as follow, in addition to standard proxy operations:

Set the Application Id back to Diameter ERP (code [[IANA8: TBD]])

Extract and cache the content of the ERP-RK-Answer.

The DEA is then forwarded to the authenticator, that can use the rMSK as described in [RFC5296].

The figure below captures this Diameter ERP Proxy behavior:



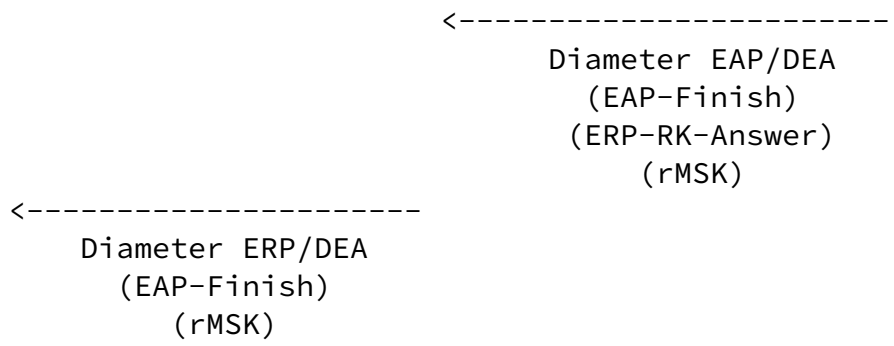


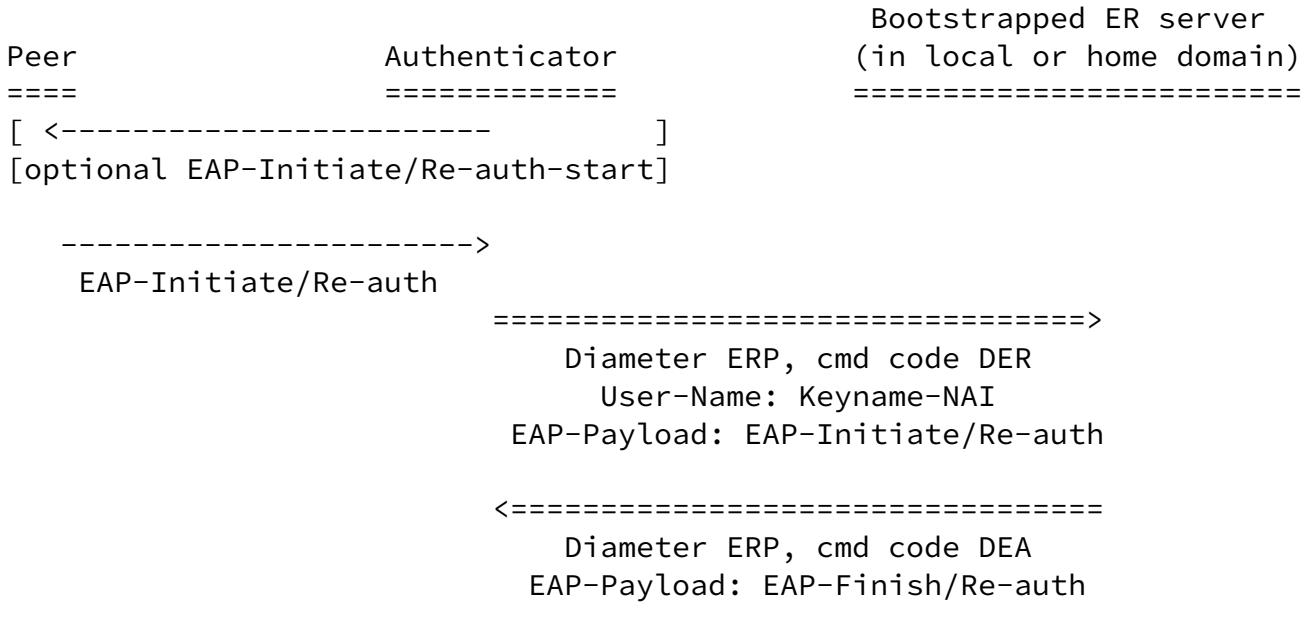
Figure 7. ERP explicit bootstrapping message flow

8. Re-Authentication

This section describes a re-authentication exchange with a bootstrapped ER server. The peer is assumed to know the domain of the bootstrapped ER server in advance. See previous section Section 7 for more information. EAP-Initiate/Re-auth-start with Domain-Name TLV is a possibility for the peer to learn the domain of the ER server attached to the authenticator.

The following figure describes the re-authentication exchange.

Explication follow.



```
                              EAP-Master-Session-Key: rMSK
   <-----------------------
     EAP-Finish/Re-auth
```

Figure 8. Diameter ERP exchange

The authenticator that does not support ERP [RFC4072] discards EAP packets with unknown ERP-specific code (EAP-Initiate). The peer falls back to full EAP authentication in that case.

When the ERP-compatible authenticator receives an EAP-Initiate/ Re-auth message from the peer (or after having sent a EAP-Initiate/ Re-auth-start packet), it process as described in [RFC5296] with regards to the EAP state machine, and similarly to Diameter EAP [RFC4072], with regards to Diameter, with the following differences:

The application id is set to Diameter ERP instead of Diameter EAP.

The User-Name and Destination-Realm are derived from the Keyname-NAI.

[[Editor16: How do we create / retrieve the Session-Id?]]

The ER server receives this request and process the ERP payload as described in [RFC5296]. If re-authentication is successful, it creates a DEA answer as described in Diameter EAP, with the following differences:

The application id is set to Diameter ERP.

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

The EAP-Master-Session-Key AVP contains the rMSK

The Result-Code AVP contains DIAMETER_SUCCESS.

In case the re-authentication fails, the Result-Code AVP contains an error code, and no EAP-Master-Session-Key AVP is included.

When the authenticator receives this answer, it processes it as described in Diameter EAP: forwards the EAP payload to the peer, and

use the rMSK as a shared secret in Secure Association Protocol.

9. Sessions

This section describes how sessions are handled in case of re-authentication.

[[Editor17: The content of this section is to be written, I am just listing the ideas here.]]

See guidelines in [I-D.ietf-dime-app-design-guide].

During initial full EAP authentication, the identity of the peer is used to create the Session-Id AVP, which is then used during accounting. When the peer attaches to a new authenticator and performs ERP, its identity is not disclosed to the authenticator. Instead, the peer presents the Keyname-NAI. This identifiers contains the EMSKName as user part. The new authenticator will therefore derive the new Session-Id from this EMSKName and use this for accounting purpose.

Although the home EAP server is able to link EMSKName with the peer's identity, the other Diameter entities do not have this mapping. In particular, the realm part of Keyname-NAI is the visited network. How does the authenticator figures out that the account records must be sent to the home domain of the peer?

It is possible to cache the necessary information at the ER server level. Is it useful to specify this mechanism in this document? It would involve:

An additional AVP during bootstrapping of ER server, in the ERP-RK-Answer, to pass the real User-Name and Session-Id (only in case of explicit bootstrapping)

An additional AVP in Diameter ERP/DEA (EAP-Finish/Re-Auth) to pass the real Session-Id and User-Name and Destination-Realm of the re-

authenticated peer, for accounting messages.

10. Contributors

Hannes Tschofenig, Lakshminath Dondeti, Julien Bournelle, and Lionel Morand wrote the initial Diameter ERP draft document.

11. Acknowledgements

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

Qin Wu and Glen Zorn actively participated in the discussions on the design for Diameter ERP, providing the point of view and experience from HOKEY workgroup.

Hannes Tschofenig provided useful advices for the writing of this document.

Many thanks to these people!

12. IANA Considerations

This document requires IANA registration of the following new elements in the Authentication, Authorization, and Accounting (AAA) Parameters [1] registries.

12.1. Diameter ERP application

This specification requires IANA to allocate a new value "Diameter ERP" in the "Application IDs" registry created by in [RFC3588].

Application Identifier	Value
Diameter ERP	TBD

IANA consideration for Diameter ERP application

12.2. New AVPs

This specification requires IANA to allocate new values from the "AVP Codes" registry defined in [RFC3588] for the following AVPs:

ERP-RK-Request

ERP-Realm

ERP-RK-Answer

ERP-RK

ERP-RK-Name

ERP-RK-Lifetime

These AVPs are defined in section Section 5.

13. Security Considerations

The security considerations from the following RFC apply here: [RFC3588], [RFC4072], [RFC5247], [RFC5295], and [RFC5296].

[[Editor18: FFS: Do we really respect these security considerations with the mechanism we describe here? Is it safe to use ERP-RK-Request / Answer AVPs? What is the worst case?]]

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.

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

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

[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.

[RFC5296] Narayanan, V. and L. Dondeti, "EAP

authentication Protocol (ERP)", RFC 5296, August 2008.

14.2. Informative References

[I-D.gaonkar-radext-erp-attrs] Gaonkar, K., Dondeti, L., Narayanan, V., and G. Zorn, "RADIUS Support for EAP Re-authentication Protocol", draft-gaonkar-radext-erp-attrs-03 (work in progress), February 2008.

[I-D.ietf-dime-app-design-guide] Fajardo, V., Asveren, T., Tschofenig, H., McGregor, G., and J. Loughney, "Diameter Applications Design Guidelines", draft-ietf-dime-app-design-guide-08 (work in progress), November 2008.

[I-D.ietf-dime-erp] Dondeti, L., Bournelle, J., Morand, L., and S. Decugis, "Diameter Support for EAP Re-authentication Protocol", draft-ietf-dime-erp-00 (work in progress), January 2009.

[I-D.ietf-hokey-key-mgm] Hoeper, K. and Y. Ohba, "Distribution of EAP based keys for handover and re-authentication", draft-ietf-hokey-key-mgm-06 (work in progress), April 2009.

[I-D.wu-dime-local-keytran] Wu, W., "Diameter support for local key transport protocol between local server and home AAA server", draft-wu-dime-local-keytran-00 (work in progress), May 2009.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol

(EAP)", RFC 3748, June 2004.

[RFC4187] Arkko, J. and H. Haverinen, "Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)", RFC 4187, January 2006.

[RFC5247] Aboba, B., Simon, D., and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework", RFC 5247, August 2008.

URIs

[1] <http://www.iana.org/assignments/aaa-parameters/>

Author's Address

Sebastien Decugis (editor)
NICT
4-2-1 Nukui-Kitamachi
Koganei, Tokyo 184-8795
JP

EMail: sdecugis@nict.go.jp