

DIME  
Internet-Draft  
Intended status: Standards Track  
Expires: August 22, 2013

H. Tschofenig, Ed.  
Nokia Siemens Networks  
February 18, 2013

A Keying Database for Diameter End-to-End Security  
draft-tschofenig-dime-keying-database-00.txt

## Abstract

The Diameter Base specification offers security protection between neighboring Diameter peers using TLS, DTLS, and IPsec. The development of a solution to protect Diameter Attribute Value Pairs between non-neighboring nodes is currently work in progress.

Diameter nodes maintain different types of databases, depending on their functions. Examples include the peer table and the realm-based routing table. This document describes a conceptual model for a keying database as it would be used by a Diameter node to determine what AVPs to protect, and what keys / algorithms to use. On the receiving side it allows the receiving node to select the appropriate security association for verifying the protected AVPs. The design is similar to IPsec and inspired by the routing protocol security key table.

## Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 22, 2013.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

Diameter Keying Database

February 2013

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">2.</a>	Terminology . . . . .	<a href="#">4</a>
<a href="#">3.</a>	Conceptual Keying Database . . . . .	<a href="#">5</a>
<a href="#">4.</a>	Examples . . . . .	<a href="#">8</a>
<a href="#">5.</a>	Security Considerations . . . . .	<a href="#">10</a>
<a href="#">6.</a>	IANA Considerations . . . . .	<a href="#">11</a>
<a href="#">7.</a>	Acknowledgments . . . . .	<a href="#">12</a>
<a href="#">8.</a>	References . . . . .	<a href="#">13</a>
<a href="#">8.1.</a>	Normative References . . . . .	<a href="#">13</a>
<a href="#">8.2.</a>	Informative References . . . . .	<a href="#">13</a>
	Author's Address . . . . .	<a href="#">14</a>

## 1. Introduction

The Diameter Base specification [[RFC6733](#)] offers security protection between neighboring Diameter peers and mandates that either TLS (for TCP), DTLS (for SCTP), or IPsec is used. These security protocols offer a wide range of security properties, including entity authentication, data-origin authentication, integrity, confidentiality protection and replay protection. They also support a large number of cryptographic algorithms, algorithm negotiation, and different types of credentials.

In the meanwhile Diameter had received a lot of deployment interest and the need for protecting Diameter AVPs between non-neighboring nodes has been created. The requirements for Diameter end-to-end security at the level of individual AVPs is provided in [I-D.tschofenig-dime-e2e-sec-req].

This document describes a conceptual model for a keying database as it would be used by a Diameter node to determine what AVPs to protect, what keys and algorithms to use. On the receiving side it allows the receiving node to select the appropriate security association for verifying the protected AVPs. The design is similar to IPsec [[RFC4301](#)] and inspired by the routing protocol security key table [[I-D.ietf-karp-crypto-key-table](#)].

## [2.](#) Terminology

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

### [3.](#) Conceptual Keying Database

Diameter nodes maintain different types of databases, depending on their functions. Examples include the peer table and the realm-based routing table. The usage of the end-to-end security mechanisms described in this document adds another database to those nodes supporting this functionality.

We describe the keying database in a conceptual way as a table, where each row represents a single symmetric or asymmetric key.

The columns that the table consists of are listed as follows:

**KeyName:** The KeyName is a string identifying the key. On the sender-side it provides information about what key to use for applying integrity and/or confidentiality protection. On the receiver-side this value provides information about the key to use for verification.

**DestinationRealm:** The DestinationRealm provides information for the sending host to decide what messages to protect. This selector field contains information about the designation realm. The

format of a DiameterIdentity. This field may be empty.

**DestinationHost:** The DestinationHost provides information for the sending host to decide what messages to protect. This selector field contains information about the destination host. The format of a DiameterIdentity. This field may be empty.

**ApplicationID:** The ApplicationID provides information for the sending host to decide what messages to protect. This field contains a list of comma separated application id values. This field may be empty. The value "\*" refers to all application ids that match the DestinationRealm and/or DestinationHost field.

**AVPCodeList:** The AVPCodeList provides information for the sending host to decide what AVPs need to experience integrity, and optionally confidentiality protection. This selector field contains a list of AVP codes. The value "\*" indicates that the protection covers all AVP fields included in the message.

**KDF:** The KDF field indicates which key derivation function is used to generate short-lived keys from a long-lived symmetric key in the Key field. For symmetric keys, when the long-lived shared key is intended for direct use, the KDF field is set to "none". This document re-used the KDF algorithm registry established in [\[I-D.ietf-karp-crypto-key-table\]](#). The protocol indicates what (if any) KDFs are valid. For asymmetric algorithm the KDF is left

empty.

**AlgID:** The AlgID field indicates the cryptographic algorithm used with the security protocol. The algorithm may be an encryption algorithm and mode (such as AES-128-CBC), an authentication algorithm (such as HMAC-SHA1-96 or AES-128-CMAC), or an algorithm applicable to asymmetric cryptography (such as RS256 indicating RSA with SHA-256). If the KDF field contains "none", then a long-lived shared secret key is used directly with this algorithm, otherwise the derived short-lived symmetric key is used with this algorithm. When the long-lived key is used to generate a set of short-lived keys for use with the security protocol, the AlgID field identifies a ciphersuite rather than a single cryptographic algorithm.

**KeyType:** The KeyType provides information about the type of key found in the Key field. Two values are possible: "SymmetricKey" and "AsymmetricKey".

**Key:** The Key field contains a symmetric or an asymmetric key. A lower-case hexadecimal string is used for representing a symmetric key. For asymmetric keys the NI URI format [[I-D.farrell-decade-ni](#)] is used. The size of the Key field depends on the type of key, the selected KDF, and the AlgID. For instance, a KDF=none and AlgID=AES128 requires a 128-bit symmetric key, which is represented by 32 hexadecimal digits.

**Direction:** The Direction field indicates whether this key may be used for inbound traffic, outbound traffic, both, or whether the key has been disabled and may not currently be used at all. The supported values are "in", "out", "both", and "disabled", respectively.

**SendNotBefore:** The NotBefore field specifies the earliest date and time in Universal Coordinated Time (UTC) at which this key should be considered for use. The format is YYYYMMDDHHSSZ, where four digits specify the year, two digits specify the month, two digits specify the day, two digits specify the hour, two digits specify the minute, and two digits specify the second. The "Z" is included as a clear indication that the time is in UTC. This field is empty if the key is for immediate use. If the Direction field indicates that the key is used not used for outbound traffic then this field is ignored.

**SendNotAfter:** The SendNotAfter field specifies the latest date and time at which this key should be considered for use when sending messages. The format is the same as the SendNotBefore field. If the Direction field indicates that the key is used not used for

outbound traffic then this field is ignored.

**RcvNotBefore:** The RcvNotBefore field specifies the earliest date and time in Universal Coordinated Time (UTC) at which this key should be considered for use when processing received messages. The format is YYYYMMDDHHSSZ, where four digits specify the year, two digits specify the month, two digits specify the day, two digits specify the hour, two digits specify the minute, and two digits

specify the second. The "Z" is included as a clear indication that the time is in UTC. This field is empty if there is no restriction regarding the use of the key when processing received messages. If the Direction field indicates that the key is used not used for inbound traffic then this field is ignored.

**RcvNotAfter:** The RcvNotAfter field specifies the latest date and time at which this key should be considered for use when processing received traffic. The format of this field is identical to the format of NotBefore. If the Direction field indicates that the key is used not used for inbound traffic then this field is ignored.

**KeyManagement:** Specifies whether a entry was statically configured or dynamically discovered. This field may help to create a new key when the existing key is expired. An empty field indicates a statically configured key. Values are reserved for automated key management protocols.



This section gives a few examples. For editorial reasons (i.e., the per-line character limit of Internet drafts) a list representation is used instead of a table.

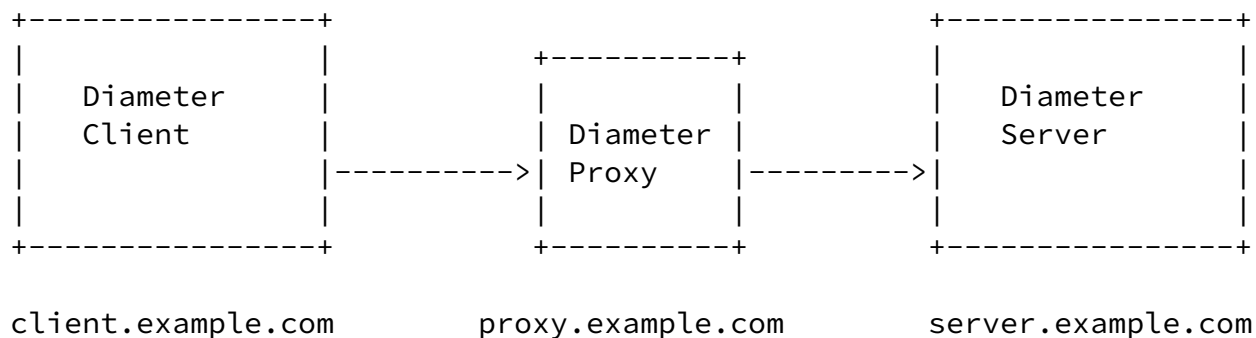


Figure 1: Example Diameter Deployment Setup.

The first example illustrates an entry in the key table at a Diameter client.

```

KeyName: abc123
DestinationRealm:
DestinationHost: server.example.com
ApplicationID: *
AVPCodeList: *
KDF: none
AlgID: HMAC-SHA1-96
KeyType: SymmetricKey
Key: 617CAA833BEF64D88E45
Direction: out
SendNotBefore:
SendNotAfter: 201302142000Z
RcvNotBefore:
RcvNotAfter :
KeyManagement:

```

The second example illustrates the entries of the key database for an asymmetric key as stored at the Diameter client.

KeyName: abc123  
DestinationRealm:  
DestinationHost: server.example.com  
ApplicationID: \*  
AVPCodeList: \*  
KDF: none  
AlgID: RS256  
KeyType: AsymmetricKey  
Key: ni:///sha-256;UyaQV-Ev4rdLoHyJJWCi110HfrYv9E1aGQAlM02X\_-Q  
Direction: both  
SendNotBefore:  
SendNotAfter: 201302142000Z  
RcvNotBefore:  
RcvNotAfter : 201302142000Z  
KeyManagement:

## 5. Security Considerations

This document focuses on the description of a keying database for usage with Diameter to protect AVPs end-to-end.

It has been recognized in [[RFC4107](#)] that automated key management is not viable in multiple scenarios. The conceptual database specified in this document is designed to accommodate both manual key management and automated key management. A future specification to automatically populate rows in the database is envisioned.

Designers should recognize the warning provided in [[RFC4107](#)]:

"Automated key management and manual key management provide very different features. In particular, the protocol associated with an automated key management technique will confirm the liveness of the peer, protect against replay, authenticate the source of the short-term session key, associate protocol state information with the short-term session key, and ensure that a fresh short-term session key is generated. Moreover, an automated key management protocol can improve the interoperability by including negotiation mechanisms for cryptographic algorithms. These valuable features are impossible or extremely cumbersome to accomplish with manual key management."

## [6.](#) IANA Considerations

[Editor's Note: An IANA consideration section will be provided in a future version of this document.]

## [7.](#) Acknowledgments

I would like to thank the authors of [[I-D.ietf-karp-crypto-key-table](#)] for their work. This document is inspired by their writeup.

## [8.](#) References

### [8.1.](#) Normative References

- [I-D.farrell-decade-ni]  
Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B., Keraenen, A., and P. Hallam-Baker, "Naming Things with Hashes", [draft-farrell-decade-ni-10](#) (work in progress), August 2012.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC6733] Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", [RFC 6733](#), October 2012.

### [8.2.](#) Informative References

[I-D.ietf-karp-crypto-key-table]

Housley, R., Polk, T., Hartman, S., and D. Zhang,  
"Database of Long-Lived Symmetric Cryptographic Keys",  
[draft-ietf-karp-crypto-key-table-05](#) (work in progress),  
February 2013.

[RFC4107] Bellovin, S. and R. Housley, "Guidelines for Cryptographic  
Key Management", [BCP 107](#), [RFC 4107](#), June 2005.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the  
Internet Protocol", [RFC 4301](#), December 2005.

#### Author's Address

Hannes Tschofenig (editor)  
Nokia Siemens Networks  
Linnoitustie 6  
Espoo 02600  
Finland

Phone: +358 (50) 4871445  
Email: [Hannes.Tschofenig@gmx.net](mailto:Hannes.Tschofenig@gmx.net)  
URI: <http://www.tschofenig.priv.at>

