Individual Contribution
Internet-Draft
Category: Standards Track
<draft-calhoun-diameter-strong-crypto-07.txt>

Pat R. Calhoun
Sun Microsystems, Inc.
William Bulley
Merit Network, Inc.
Stephen Farrell
Baltimore Technologies
March 2001

Diameter Strong Security Extension

Status of this Memo

Abstract

The Diameter base protocol defines message integrity and AVP encryption using symmetric transforms to secure the communication between two Diameter nodes. The base protocol also defines a Diameter proxy server, that forwards requests to other servers when it detects that a given request cannot be satisfied locally.

The ROAMOPS Working Group has defined a requirement that allows for the Diameter servers communicating through the proxy to be able to provide for end-to-end AVP integrity and confidentiality, making it difficult for the proxy to be able to modify, and/or be able to view sensitive information, within the message. The Mobile-IP and NASREQ Working Groups have stated that strong authentication is a requirement for AAA data, such as accounting records, for the purposes of non-repudiation.

This Diameter extension specifies how strong AVP authentication, integrity and encryption can be done using a mixture of symmetric and asymmetric transforms, by encapsulating Cryptographic Message Syntax (CMS) data into Diameter AVPs. The CMS data can also be used to carry X.509 certificates.

Table of Contents

1.0 Introduction

The Diameter base protocol [1] defines message integrity and AVP encryption using symmetric transforms to secure the communication between two Diameter nodes. The base protocol also defines a Diameter proxy server, that forwards requests to other servers when it detects that a given request cannot be satisfied locally.

The ROAMOPS Working Group has defined a requirement in [10] that allows for the Diameter servers communicating through the proxy to be able to provide for end-to-end AVP integrity and confidentiality, making it difficult for the proxy to be able to modify and see sensitive information within the message. The Mobile-IP and NASREQ Working Groups have stated in [6, 7, 8] that non-repudiation is a requirement for AAA data, such as accounting records.

When a chain of proxies use hop-by-hop security, each node in the proxy chain MUST recompute the Integrity-Value-Check (ICV) [1], making it easy for a malicious proxy to modify information in a Diameter message. It is virtually impossible for the rest of the nodes in the proxy chain to know that the message was modified in mid-stream. Figure 1 shows an example of such a network, where DIA3 modifies the contents of "foo" in both the request and the response.

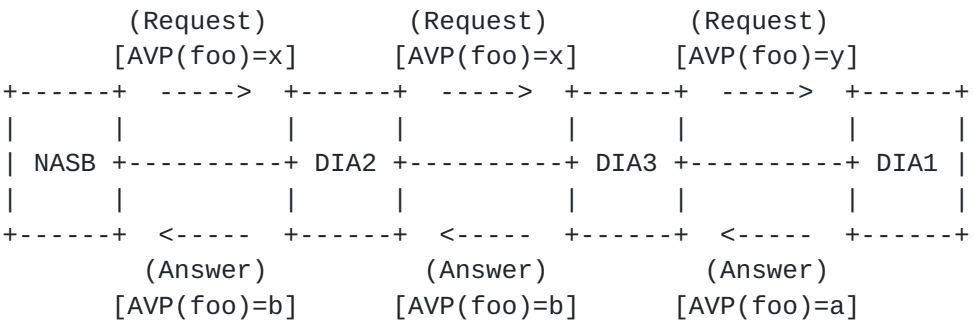


Figure 1: Proxy Chain

This document describes how strong authentication and encryption can be provided in the Diameter protocol, by encapsulating CMS objects [3] in AVPs. The CMS object can also be used to carry X.509 certificates and revocation lists.

In the example provided in Figure 1, the originator of the request and response adds a digital signature that covers a set of AVPs within the message. The protected AVPs MUST NOT be changed by an intermediate proxy server (DIA2, DIA3), since the signature validation performed by the end server would fail.

The Diameter base protocol also allows a Diameter broker to provide redirect services, as shown in Figure 2. The Diameter broker MAY

return information to a requesting server that would allow the servers to interact directly, bypassing the broker. This optimized approach reduces the complexity associated with end-to-end security.

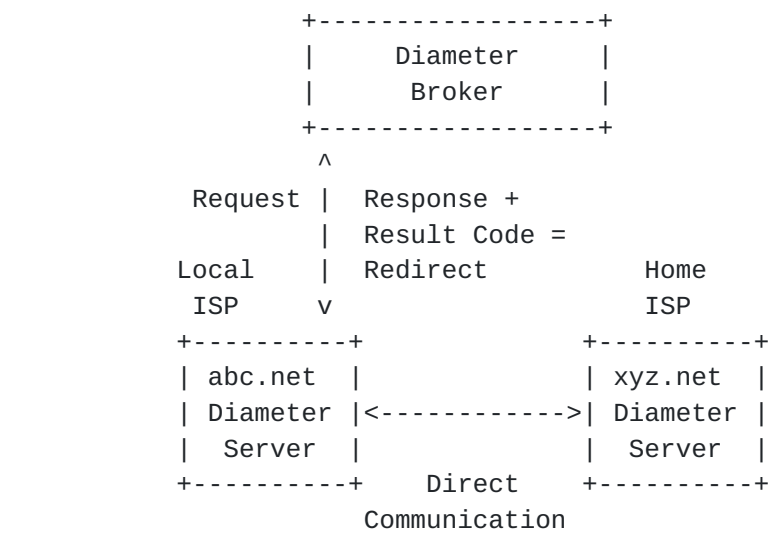


Figure 2: Diameter Broker Returning Redirect Indication

When redirect services are used, a network layer security protocol, such as IP Security, MAY be used to secure the traffic between the two Diameter servers. However, security at the application level may still be necessary in this network configuration, specifically the ability to authenticate a select set of AVPs. Brokers that operate in a redirect mode typically require that both Diameter servers sign accounting records. The accounting record, signed by both parties is then forwarded to the broker via the local Diameter server. This provides the broker with some assurances that both networks agreed on the accounting data, which it MAY use for settlement purposes. If the underlying security protocol provides confidentiality, strong encryption MAY not be necessary in the redirect case.

Given that asymmetric transform operations are expensive, Diameter servers MAY wish to use them only when dealing with inter-domain servers, as shown in Figure 3. This configuration is normally desirable since Diameter entities within a given administrative domain MAY inherently trust each other. Further, it is desirable to move this functionality to the edges, since NASes do not necessarily have the CPU power to perform expensive cryptographic operations.

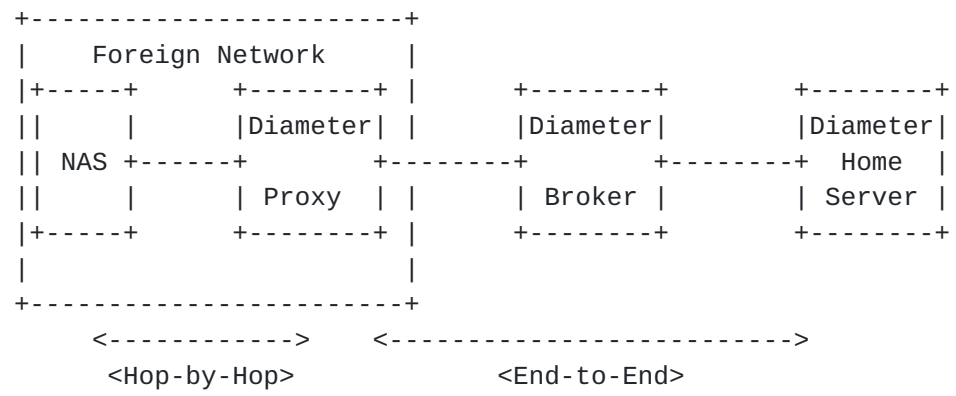


Figure 3: Mixed Diameter Security Models

The Extension number for this draft is two (2). This value is used in the Extension-Id AVP as defined in [1].

1.1 Restricted Use Cases

[ed: This section is up for discussion in Minneapolis.]

Altough this document specifies both authentication and confidentiality services, it is expected that the most common use cases will only require confidentiality service from a NAS to a "home" AAA server. It also appears that there is no use case where the "home" AAA server requires the NAS to be authenticated.

This has a number of consequences:-

- It is much simpler since the (many) NASes do not need to store or process secret information (also much more computationally intensive with asymmetric mechanisms).
- In principle, the "P" bit and the use of SignedData could be deprecated in this document, or at least changed from "MUST" or "SHOULD" to "MAY", again making conformance simpler.
- It means that the PKI required to support this Diameter extension is basically (from the PKI perspective) analagous to what has been demonstrated to work in the HTTP/SSL case - where servers on the Internet are certified and where clients verify those server certificates. PKIs that establish server identities are much easier to setup and operate in comparison to those which establish both client and server identities.

1.2 Requirements language

In this document, the key words "MAY", "MUST", "MUST NOT",

"optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [5].

2.0 Extended AVP Format

This specification introduces the 'P' bit in the AVP Header, which is defined in [1]. The 'P' bit, known as the protected AVP bit, is used to indicate whether the AVP is protected by a digital signature. When set, the AVP is protected and the contents cannot be changed by a Diameter proxy server without detection.

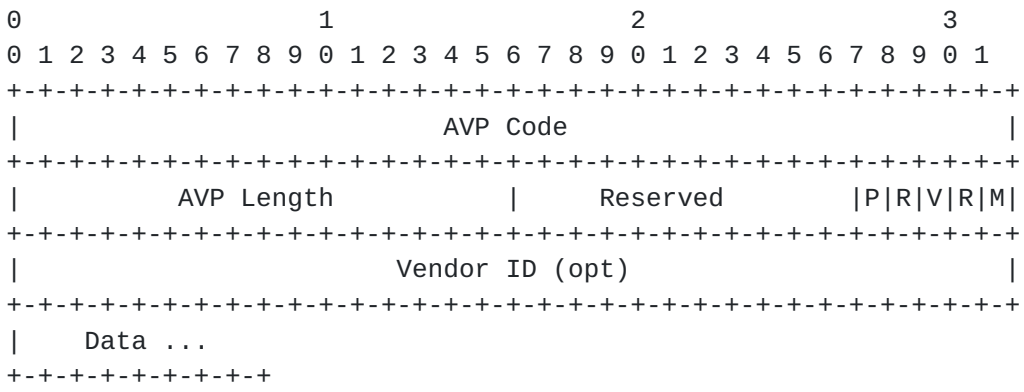


Figure 4: Extended Diameter AVP Header

Note that unless stated otherwise, the 'P' bit can be set on any Diameter AVP. The Proxy-State and Integrity-Check-Value AVPs [1] MUST NOT have the 'P' bit set. The Encrypted-Payload AVP MAY have the 'P' bit set if there is no intermediate proxy server. Any additional AVPs that MUST be removed, or changed, at each hop in a proxy chain MUST NOT have the 'P' bit set.

3.0 Key Management

For e2e origin authentication, CMS itself already provides sufficicent key management without the need for additional specification. Basically, the originating Diameter node signs and includes whatever certificates are necessary for validation of the digital signature.

However, for encryption of AVPs more work is needed. In order to be able to encrypt AVPs for a recipient, the originating Diameter node must have a copy of the recipient's public key. There are many well-known key retrieval schemes (e.g. using LDAP [17]), however, in order to simplify Diameter implementations a specific Diameter key distribution mechanism is defined here.

Another issue that must be addressed is how a Diameter node is to "know" that certain AVPs are required to use the strong security extension. The model here is that this information is to be included into the public key certificates of the Diameter servers (see below for details.)

Finally, this section addresses the certificate profile to be used for this Diameter extension, which is a simplified profile of [4].

3.1 Usage Scenario

When a Diameter node is about to send a messsage which MAY use strong security, it must determine whether to use the strong security service or not. We assume the Diameter node knows the user's NAI, which determines the user's realm.

In the present discussion we assume that the Diameter node has not cached any information. Where information can be cached this is noted.

We use Diameter E2E-SA-Setup-Request (ESSR) and E2E-SA-Setup-Answer (ESSA) messages to establish whether strong security is required and if so, for which AVPs and which public key(s) to use.

The originating node sends the ESSR message to a server in the destination realm. The ESSR message contains:

- the realm part of the user's NAI
- the list of direct trust CA's that the originating Diameter node has configured into it for certificate validation
- (optionally) a nonce to be used by the destination Diameter node in OCSP requests determining certificate status

(Note: A "direct trust" CA is one that the node is willing to use as the "top" of a certificate chain, sometimes confusingly known as a "root CA.")

The destination node returns the ESSA message which contains:

- TTL for this SA (seconds)
- a chain of CA certificates (possibly empty)
- public key certificates for the AAA servers in the realm, all of which MUST validate up to one of the CA's contained in the ESSR message, via the chain of CA certificates above; each of these public key certificates SHOULD contain a list of AVPs that MUST be protected (and how) for this realm
- (optionally, if nonce received and OCSP supported) a list of

OCSP responses for the certificates in question, each of which uses the nonce from the ESSR message [ED: question - if 1 ocsp responder used, do we need to append to the nonce for each request?]

The originating Diameter node now has to check the response. Any failure results in error messages and auditing and not sending the Diameter message.

Checks:

- the certificate chain selected is cryptographically correct, passes the (relevant parts of the) rfc2459 path validation algorithm and terminates at a CA mentioned in the ESSR message
- the realm part of the user's NAI must occur as a subjectAltName (with the rfc822Address option) in the AAA server's certificate. This rfc822Address MUST be of the form "Diameter-<XXX>@<domain>" where <domain> is the NAI's domain component and <XXX> can be anything (e.g. "Diameter-1@baltimore.com", "Diameter-west@sun.com") chosen by the AAA server administrator.

The list of AVPs which are to be protected are included as a Diamater-specific X.509 certificate extension contained within the public key certificate of the AAA server. This extension uses the Extension object identifier <<TBD>> and has an IA5String syntax. The string value consists of a space separated list of AVP numbers to be encrypted, followed by a comma, followed by a space separated list of AVP numbers where origin authentication is required. For example:

"123 456, 768" ",768 910" "123 456"

[ed: Note: The alternative is to add this information explicitly to the ESSA message, which would require origin authentication of the ESSA message itself. With the above (PKI based) approach, the sender of the ESSA is only being trusted for the TTL part of the ESSA which is not (very) sensitive.]

If certificate status (revocation) is an issue for the Diameter node, then the ESSR message MAY contain a nonce value. The idea is that the sender of the ESSA makes OCSP requests on behalf of the Diameter node and returns the OCSP responses to the Diameter node as part of the ESSA message. The use of the nonce value ensures that the sender of the ESSA cannot return cached or otherwise fake OCSP responses to the Diameter node. The nonce value is to be (the beginning of) the nonce in the OCSP response. <<Note: The reason for "beginning" above is that an OCSP responder might produce an error if presented with the same nonce more than once. This is to be investigated.>>

If e2e confidentiality is required, then the originating node prepares the CMS-Data AVP as required.

3.2 Certificate Requirements

[Ed: Some repetition here. To be fixed later.]

Certificates used for the purposes of Diameter MUST conform to the PKIX profile [4], and MUST also include a Diameter node's NAI, which is typically added in the Host-Name AVP [1], as one of the values of the subjectAltName extension of the Certificate. The NAI is to be encoded as an rfc822Name within the subjectAltName.

For Diameter nodes (capable of acting as recipients for e2e confidentiality), the NAI MUST be of the form "Diameter-<xxx>@<realm>". Other Diameter nodes MAY use this naming scheme.

These names are used for two purposes:

1. Where a Diameter node is verifying a signature it needs to be able to compare the identity of the signer against the identity in the Host-Name AVP.

2. Where a Diameter node is encrypting AVPs, it needs to be able to ensure that it uses a public key for the intended recipient. This requries comparing the identity in a Certificate against the NAI of the intended recipient (which is assumed to be known).

In either case, the presence of the required NAI as an rfc822Name value in the subjectAltName extension of a verified public key certificate satisfies the matching requirement.

Note that there MAY also be other values in the subjectAltName extension, (either using rfc822Name or other elements of the CHOICE), these can be safely ignored, but implementations MUST be able to handle their presence.

Note also that the PKIX profile [4], section 4.1.2.6, specifies the rules for the relationship between the subjectAltName extension and the subject field of public key certificates.

As noted above the public key certificates are to include the information about which AVPs are to be protected. This information is encoded using the new certificate extension documented above.

4.0 Command-Codes Values

This section defines new Command-Code [1] values that MUST be supported by all Diameter implementations that conform to this specification. The following Command Codes are currently defined in this document:

Command-Name	Abbrev.	Code	Reference
E2E-SA-Setup-Request	ESSR	304	4.1
E2E-SA-Setup-Answer	ESSA	305	4.2

[ed: The messages will be formally described in more detail later.]

4.1 E2E-SA-Setup-Request (ESSR) Command

The E2E-SA-Setup-Request command, indicated by the Command-Code field set to 304, is sent by a Diameter node to establish a Diameter End-to-End Security Association.

```
<E2E-SA-Setup-Request> ::= < Diameter-Header: 304 >
                           { Origin-FQDN }
                           { Origin-Realm }
                           { Destination-Realm }
                         + { Local-CA-info }
                           [ Destination-FQDN ]
                         * [ Ocsp-nonce ]
                         * [ Proxy-State ]
                         * [ Route-Record ]
                        0*1< Integrity-Check-Value >
```

4.2 E2E-SA-Setup-Answer (ESSA) Command

The E2E-SA-Setup-Answer command, indicated by the Command-Code field set to 305, is sent by a Diameter node in response to an ESSR message.

```
<E2E-SA-Setup-Answer> ::= < Diameter-Header: NNN+1 >
                          { Origin-FQDN }
                          { Origin-Realm }
                          { Destination-Realm }
                        0*1{ Ca-chain }
                         + { AAA-server-certs }
                         * { Ocsp-responses }
                           [ Destination-FQDN ]
                         * [ Proxy-State ]
                         * [ Route-Record ]
                        0*1< Integrity-Check-Value >
```

5.0 Strong Security AVPs

This section contains AVPs that are used to establish a Diameter End-to-End Security Association.

Attribute Name	AVP Code	Section Defined	Value Type	AVP Flag rules				
				MUST	MAY	SHLD NOT	MUST NOT	MAY Encr
CMS-Data	310	5.1	OctetString	M	P		V	N
Local-CA-Info	348	5.2	Grouped	M	P		V	N
CA-Name	349	5.2.1	OctetString	M	P		V	N
Key-Hash	350	5.2.2	OctetString	M	P		V	N

5.1 CMS-Data AVP

The CMS-Data AVP (AVP Code 310) is of type OctetString and contains the Distinguished Encoding Rules (DER) encoding of a CMS object [3] of type ContentInfo. The profile of CMS algorithm and structure usage is as specified in the S/MIME v3 message specification [11]. This means that where a set of AVPs is protected using CMS, the set MUST first be encoded according to MIME encoding rules specified below. This method of encapsulating AVPs allows existing S/MIME toolkits to be used without changes in order to produce strongly protected Diameter messages. The CMS object MAY contain any of the three methods; signed-only, enveloped-only and signed-and-enveloped. Optional certificates and CRLs MAY be present in all three methods.

To package a set of AVPs as a MIME type, the AVPs are first concatenated in the order in which they occur in the Diameter message. The entire AVP MUST be input to the signing/encryption

process, from the first byte of the AVP code to the last byte of the AVP data, including all other fields, length, reserved/flags, and optional vendor IDs, and padding. The AVP MUST be input to the signing/encryption process in network byte order. If AVPs are to be enciphered, then the encryptor is free to order AVPs whatever way it chooses. This value is then used as the value of a new MIME type application/x-Diameter-avps, which MUST be prepared in accordance with the rules specified in section 3.1 of [11]. If a receiver detects that the contents of the CMS-Data AVP is invalid, it SHOULD return the new Result-Code AVP value defined in section 6.0.

Where signing only is performed, the signature is calculated over the canonical encoding of the application/x-Diameter-avps MIME type, but the AVPs themselves are not carried within the CMS-Data AVP. Instead, the digest value within the SignedData structure contains the digest over the canonicalized encoding of application/x-Diameter-avps. Multiple Diameter entities MAY add their signatures to an existing CMS-Data AVP using the countersignature attribute, defined in section 11.4 of [3]. The countersignature attribute requires that the signatures occur sequentially, meaning that each node's signature covers the existing signatures in the CMS object.

Where encryption only is performed, the encryptedContent MUST contain the canonical encoding of the application/x-Diameter-avps MIME type.

Where signing and encryption are both performed, signing MUST occur first, the resulting CMS object MUST then be MIME encoded producing an application/pkcs7-mime MIME type which is then used as the content of the EnvelopedData.

There is no need for an 'outer' MIME encoding when only signing, or only encryption is applied.

Where AVPs are encapsulated within a CMS-Data AVP, the eContentType of the EncapsulatedContentInfo MUST be id-data [11].

The signing and encryption algorithms supported MUST be those specified in sections 2.2 and 2.3 of [11].

Conformant implementations MUST emit a CMS-Data AVP which contains only one application/x-Diameter-avps MIME type. Implementations which receive any other MIME type MUST indicate an error.

Where a CMS-Data AVP contains a set of certificates then both public key certificates (Certificate) and attribute certificates (AttributeCertificate) are allowed by CMS (as well as one other legacy format which MUST NOT be used). Support for use of the Certificate structure is REQUIRED, implementations SHOULD support use

of the AttributeCertificate structure as defined in the PKIX attribute certificate profile [12]. This allows Diameter implementations to include a certifiate from a trusted party that they are authorized to emit the AVPs contained in the message.

When a SignedData object is present, the eContent field of the EncapsulatedContentInfo structure MUST be absent since the authentication covers data outside of the object. The signature is computed over all AVPs prior to the AVP that have the 'P' bit enabled. The order of the AVPs MUST be preserved and the computation begins with the first AVP immediately following the Diameter header. If the CMS-Data AVP is present in a Grouped-AVP, it covers all AVPs within the Grouped-AVP AVP that has the 'P' bit set. An Integrity-Check-Value (ICV) AVP MUST NOT preceed a CMS-Data AVP containing a SignedData object. If the signature cannot be verified correctly, a response with the Result-Code AVP set to DIAMETER_INVALID_AUTH [1] MUST be returned.

When an EnvelopedData object is present, the encryptedContentInfo field MUST contain the Host-Name AVP, containing the host name of the encryptor, and one or more additional AVPs.

When a conforming implementation receives a Diameter message which contains encrypted AVPs within a CMS EnvelopedData, then the recipient MUST check to see if it is on the list of recipients specified in the RecipientInfos of the EnvelopedData. If not, the recipient MAY choose to process the message or indicate an error. If the recipient is in the RecipientInfos and an error occurs during decryption, then the recipient MUST indicate an error.

Diameter nodes SHOULD implement content encryption key re-use.

A CMS-Data MAY also contain a certs-only CMS structure, which is a degenerate form of CMS structure containing only PKI related information (see section 3.6 of [11] for details of the CMS certs-only structure). This use of the CMS-Data AVP can be used to "push" public key and attribute certificates and CRLs using Diameter, which MAY be useful in environments where repositories (e.g. LDAP servers) are either not used or not available (e.g. due to crossing a domain boundary). Conforming implementations MUST be able to emit a certs-only CMS structure which contains relevant PKI related information and MUST be able to process a CMS-Data AVP which contains a certs-only CMS structure. Of course, the recipient of such a certs-only CMS structure SHOULD NOT use the PKI related information without first verifying it, e.g. by checking that issuer's are trusted, signatures verify etc.

When the CMS-Data AVP contains certificates in the certificates field

of the SignedData, a CRL [4] MAY also be provided in the crls field of the SignedData, which MAY be used to assist in determining whether a certificate has been revoked. Optionally, the Diameter node MAY check the status of certificates using another mechanism, such as Online Certificate Status Protocol (OCSP) [9].

This AVP MUST have the 'M' bit enabled. The 'P' and 'V' bits MUST NOT be enabled.

The following is an example of a message that includes strong security and hop-by-hop security:

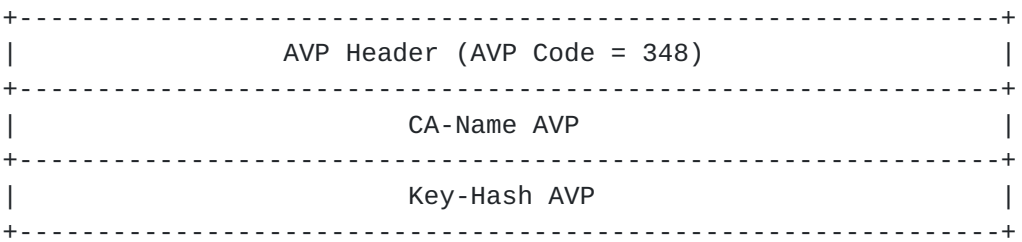
```
Example-Command ::= < Diameter-Header: 9999999 >
                    [ AVP ]
                    { CMS-Data }
                  * [ Proxy-State ]
                  * [ Route-Record ]
                  * [ Routing-Realm ]
                 0*1< Integrity-Check-Value >
```

5.2 Local-CA-Info AVP

The Local-CA-Info AVP (AVP Code = 348) is of type Grouped. The Grouped Data field has the following ABNF grammar:

```
Local-CA-Info   = CA-Name Key-Hash
   CA-Name       = ; See Section 5.2.1
   Key-Hash      = ; See Section 5.2.2
```

The Local-CA-Info AVP Data field contains the Certificate Authority's name in the CA-Name AVP, as well as a hash of the key in the Key-Hash AVP.



5.2.1 CA-Name AVP

The CA-Name AVP (AVP Code = 349) is of type OctetString, encoded in the UTF-8 [24] format. The AVP contains the FQDN of the Certificate Authority.

5.2.2 Key-Hash AVP

The Key-Hash AVP (AVP Code = 350) is of type OctetString, and contains a hash of the key. [ed: More later on how this is generated].

6.0 Result-Code AVP Values

This section defines new Result-Code [1] values that MUST be supported by all Diameter implementations that conform to this specification.

6.1 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_INVALID_CMS_DATA 5018
This error code is returned when a CMS-Data AVP is received with an invalid ContentInfo object.

7.0 IANA Considerations

The AVPs defined in Section 5.0 are AVPs whose identifiers were allocated from the AVP numbering space [1], and extended in [13], [14] and [15]. IANA should assign the values in section 5.0.

The Result-Code values defined in Section 6.0 are error codes as defined in [1] and extended in [13], [14] and [15]. They correspond to error values specific to the Strong Security extension. IANA should record the values as defined in Section 6.0.

8.0 Security Considerations

This document describes how strong security can be achieved in the Diameter protocol by allowing S/MIME Cryptographic Message Syntax [3] objects to be carried as a Diameter AVP.

Section 5.1 states that a certificate received in a CMS-Data AVP SHOULD NOT be used prior to cert verification. In most cases, the verification will be according to the rules specified in [4], however, some communities have indicated that they wish to be allowed to specify alternative certificate verification mechanisms, hence the

"SHOULD NOT" rather than the more typical "MUST NOT". The authors do however strongly RECOMMEND that the verification procedures specified in [4] are always applied, regardless of whatever other verification mechanisms are in use.

9.0 References

[1] P. Calhoun, A. Rubens, H. Akhtar, E. Guttman, "Diameter Base Protocol", draft-ietf-aaa-Diameter-01.txt, IETF work in progress, March 2001.

[2] Kaufman, Perlman, Speciner, "Network Security: Private Communications in a Public World", Prentice Hall, March 1995, ISBN 0-13-061466-1.

[3] R. Housley, "Cryptographic Message Syntax", RFC 2630, June 1999.

[4] Housley, Ford, Polk, Solo, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", RFC 2459, January 1999.

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

[6] M. Beadles, D. Mitton, "Criteria for Evaluating Network Access Server Protocols", draft-ietf-nasreq-criteria-05.txt, IETF work in progress, June 2000.

[7] T. Hiller et al., "Cdma2000 Wireless Data Requirements for AAA", draft-hiller-cdma2000-AAA-02.txt, IETF work in progress, September 2000.

[8] S. Glass, S. Jacobs, C. Perkins, "Mobile IP Authentication, Authorization, and Accounting Requirements". RFC 2977. October 2000.

[9] Myers, Ankney, Malpani, Galperin, Adams, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP)", RFC 2560, June 1999.

[10] Aboba, Zorn, "Criteria for Evaluating Roaming Protocols", RFC 2477, January 1999.

[11] B. Ramsdell, "S/MIME v2 Message Specification", RFC2633, June 1999.

[12] S. Farrell, R. Housley, "An Internet Attribute Certificate

Profile for Authorization", draft-ietf-pkix-ac509prof-05.txt, IETF work in progress, August 2000.

[13] P. Calhoun, W. Bulley, G. Zorn, "Diameter NASREQ Extension", draft-ietf-aaa-Diameter-nasreq-01.txt, IETF work in progress, March 2001.

[14] P. Calhoun, C. Perkins, "Diameter Mobile IP Extensions", draft-ietf-aaa-Diameter-mobileip-01.txt, IETF work in progress, March 2001.

[15] Arkko, Calhoun, Patel, Zorn, "Diameter Accounting Extension", draft-ietf-aaa-Diameter-accounting-01.txt, IETF work in progress, March 2001.

[16] Farrell, Turner, "Reuse of CMS Content Encryption Keys", draft-ietf-smime-rcek-01.txt, IETF work in progress, February 2001.

[17] Boyen, Howes, Richard, "Internet X.509 Public Key Infrastructure Operational Protocols - LDAPv2", RFC 2559, April 1999.

10.0 Acknowledgements

The authors would also like to acknowledge the following people for their contribution in the development of the Diameter protocol:

Bernard Aboba, Jari Arkko, William Bulley, Daniel C. Fox, Lol Grant, Ignacio Goyret, Nancy Greene, Peter Heitman, Paul Krumviede, Fergal Ladley, Ryan Moats, Victor Muslin, Kenneth Peirce, Sumit Vakil, John R. Vollbrecht, Jeff Weisberg and Glen Zorn

11.0 Authors' Addresses

Questions about this memo can be directed to:

Pat R. Calhoun
Network and Security Research Center, Sun Labs
Sun Microsystems, Inc.
15 Network Circle
Menlo Park, California, 94025
USA

Phone: +1 650-786-7733
Fax: +1 650-786-6445
E-mail: pcalhoun@eng.sun.com

William Bulley
Merit Network, Inc.
Building One, Suite 2000
4251 Plymouth Road
Ann Arbor, Michigan, 48105-2785
USA

Phone: +1 734-764-9993
Fax: +1 734-647-5185
E-mail: web@merit.edu

Stephen Farrell
Baltimore Technologies
39 Parkgate Street,
Dublin 8,
IRELAND

Phone: +353-1-881-6000
Fax: +353-1-881-7000
E-Mail: stephen.farrell@baltimore.ie

12.0 Full Copyright Statement

13.0 Expiration Date

This memo is filed as <draft-calhoun-Diameter-strong-crypto-07.txt> and expires in August 2001.