AAA Working Group                                          Jari Arkko
Internet-Draft                                       Oy LM Ericsson Ab
Category: Standards Track                              Pat R. Calhoun
<draft-ietf-aaa-diameter-accounting-01.txt>        Sun Microsystems, Inc.
                                                           Glen Zorn
                                                   Cisco Systems, Inc.
                                                          March 2001

# Diameter Accounting Extensions


Status of this Memo

Abstract

   The Diameter protocol provides Authentication, Authorization and
   Accounting for network access technologies, such as NASREQ,
   ROAMOPS and Mobile IP. This extension describes how accounting
   records can be securely transmitted over the Diameter protocol.
   When combined with the Strong Security extension, accounting
   records MAY traverse intermediate proxies in a secure fashion and
   is compatible with the referral broker model. This extension
   allows real-time accounting transfers.


Table of Contents

## 1.0  Introduction

This accounting protocol is based on an authorization-server directed
model with capabilities for real-time delivery of accounting
information. Several fault resilience methods [11] have been built in
to the protocol in order minimize loss of accounting data in various
fault situations and under different assumptions about the
capabilities of the used devices.

## 1.1  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 [6].

## 1.2  Authorization-Server Directed Model

The authorization-server directed model means that at authorization
time, the device generating the accounting data gets information from
the authorization server regarding the way accounting data shall be
forwarded. This information includes accounting record timeliness
requirements.

As discussed in [11], real-time transfer of accounting records is a
requirement, such as the need to perform credit limit checks and
fraud detection. However, [10] states that batch accounting is not a
current requirement, and is therefore not supported by this
extension. Should Batched Accounting be required in the future, a new
Diameter extension will need to be created, or it could be handled
using another protocol.

The authorization server (chain) directs the selection of proper
transfer strategy, based on its knowledge of the user and
relationships of roaming partnerships. The server (or proxies in
between) uses the Accounting-Interim-Interval AVP to control the
operation of the Diameter peer operating as a client. The
Accounting-Interim-Interval AVP, when present, instructs the Diameter
node acting as a client to produce accounting records continuously
even during a session.

The Extension number for this draft is five (5). This value is used
in the Extension-Id Attribute value Pair (AVP) as defined in [1].

## 1.3  Protocol Messages

The Diameter peer, acting as a client, generating the accounting data
will use the Accounting-Request message to send accounting records to
its peer. The receiver (server) MUST reply with the Accounting-Answer
message with appropriate confirmations.

Upon receipt of an Accounting-Request, a home Diameter server MUST
generate a response. A home server is the node that "owns" the realm
portion of the user's NAI. The response includes the Result-Code,
which MAY contain an error if the accounting information is
incorrect.

Each Diameter Accounting protocol message MAY be compressed using
IPComp [12] in order to reduce the used network bandwidth, which MAY
use IKE [7] to negotiate the compression parameters.


## 1.4  Accounting Info, Usage and Service Specific AVPs

There are three separate type of classes of Accounting AVPs;
Informational, Usage and Service-Specific. The first two are
specified in this document, while Service-Specific AVPs are described
in service-specific extension documents. Informational Accounting
AVPs are used describe the accounting message, including numbering,
the type of record, interim accounting intervals, etc.  The
Accounting Usage AVPs, on the other hand, describe usage information
for a given session, and are described in section 4.0.


### 1.4.1  Extension document requirements

Each Service-Specific Diameter extension (e.g. NASREQ, MobileIP),
MUST define their Service-Specific AVPs that MUST be present in the
Accounting-Request message in a section entitled "Accounting
Considerations". The extension MUST assume that the AVPs described in
this document will be present in all Accounting messages, so only
their respective service-specific AVPs need to be defined in this
section.


## 1.5  Fault Resilience

Diameter Base protocol [11] mechanisms are used to overcome small
message loss and network faults of temporary nature.

Diameter peers acting as clients MUST implement the use of alternate
servers to guard against server failures and certain network
failures. Diameter peers acting as servers or related off-line
processing systems MUST detect duplicate accounting records caused by

the sending of same record to several servers and duplication of
messages in transit. This detection MUST be based on the inspection
of the Session-Id [1] and Accounting-Record-Number AVP pairs.

Diameter clients MAY have non-volatile memory for the safe storage of
accounting records over reboots or extended network failures, network
partitions, and server failures.  If such memory is available the
client SHOULD store new accounting records there as soon as the
records are created and until a positive acknowledgement of their
reception from the Diameter Server has been received. Upon a reboot,
the client MUST starting sending the records in the non-volatile
memory to the accounting server with appropriate modifications in
termination cause, session length, and other relevant information in
the records.

A further extension of this protocol may include AVPs to control how
many accounting records may at most be stored in the Diameter client
without committing them to the non-volatile memory or transferring
them to the Diameter server.

The client SHOULD NOT remove the accounting data from any of its
memory areas before the correct Accounting-Answer has been received.
The client MAY remove oldest, undelivered or yet unacknowledged
accounting data if it runs out of resources such as memory. It is an
implementation dependent matter for the client to accept new sessions
under this condition.


1.6  **Session Records**

In all accounting records the Session-Id and User-Name AVPs MUST be
present. If strong authentication is required, as described in [9],
the CMS-Data AVP may be used to authenticate the Accounting Data and
Service Specific AVPs. It is not typically necessary, nor
recommended, that the strong authentication cover any additional AVPs
since the Data and Service Specific AVP, and associated CMS-Data, MAY
need to be submitted to a third party (see section 1.7 below).

Different types of session records are sent depending on the actual
type of accounted service and the authorization server's directions
for interim accounting. If the accounted service is a one-time event,
meaning that the start and stop of the event are simultaneous, then
the Accounting-Record-Type AVP MUST be present and set to the value
EVENT_RECORD.

If the accounted service is of a measurable length, then the AVP MUST
use the values START_RECORD, STOP_RECORD, and possibly,
INTERIM_RECORD.  If the authorization server has directed interim

accounting to be enabled for the session, but no interim interval was
specified, two accounting records MUST be generated for each service
of type session.  When the initial Accounting-Request is sent for a
given session is sent, the Accounting-Record-Type AVP MUST be set to
the value START_RECORD. When the last Accounting-Request is sent, the
value MUST be STOP_RECORD.

If a specified interim interval exists, the Diameter client MUST
produce additional records between the START_RECORD and STOP_RECORD,
marked INTERIM_RECORD. The production of these records is directed
both by Accounting-Interim-Interval as well as any re-authentication
or re-authorization of the session.  The Diameter client MUST
overwrite any previous interim accounting records that are locally
stored for delivery, if a new record is being generated for the same
session. This ensures that only one pending interim record can exist
on a NAS for any given session.


## 1.7  Accounting in brokering environments

The Diameter base protocol [1] describes brokers that provide
redirect services, by allowing AAA servers within a roaming
consortium to directly communicate. Referral services can be secured
using the strong security extension defined in [9]. Since brokers can
also provide settlement services, they typically need to be aware of
the accounting information exchange, and since they are no longer
part of the message exchange, the Diameter protocol MUST allow the
broker to receive the accounting record.  The strong security [9]
provides the broker with the assurances it needs that both parties
agreed with the accounting information submitted.

When the local AAA server issues an Accounting-Request to the home
AAA server, it includes accounting usage and service specific AVPs as
well as a CMS-Data AVP [9], which contains the signature of the local
AAA server. The home server MUST add it's own signature to the CMS-
Data AVP, that covers both the same AVPs as above and the local AAA's
signature.  The whole is submitted to the local AAA server in the
Accounting-Answer.  The local AAA server MUST submit the accounting
AVPs, and associated CMS-Data AVP to the broker. The broker can
verify that both parties participated and accepted the accounting
record, by validating the signatures.


## 2.0  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
    -------------------------------------------------------
    Accounting-Answer          ACA      272         2.2
    Accounting-Poll-Ind        API      273         2.4
    Accounting-Request         ACR      271         2.1
    Accounting-Status-Ind      ASI      279         2.3
```

## 2.1  Accounting-Request (ACR) Command

The Accounting-Request command, indicated by the Command-Code field
set to 271, is sent by a Diameter node, acting as a client, in order
to exchange accounting information with a peer.

When the Accounting-Request is being submitted to a broker, and
includes the CMS-Data AVP [9], the CMS-Data AVP MUST be signed by
both the local and home Diameter server using the countersignature
procedures described in [9].

The AVP listed below SHOULD include service specific accounting AVPs,
as described in section 1.4.

Message Format

```
    <Accounting-Request> ::= < Diameter Header: 271 >
                             < Session-Id >
                             { Extension-Id }
                             { Origin-FQDN }
                             { Origin-Realm }
                             { Destination-Realm }
                             { Accounting-Record-Type }
                             { Accounting-Record-Number }
                             [ Accounting-Interim-Interval ]
                             { Accounting-Session-Id }
                             { Accounting-Authentication-Type }
                             { Accounting-Input-Octets }
                             { Accounting-Output-Octets }
                             { Accounting-Session-Time }
                             { Accounting-Input-Packets }
                             { Accounting-Output-Packets }
                           * [ AVP ]
                             [ CMS-Data ]
                           * [ Proxy-State ]
                           * [ Route-Record ]
                         0*1< Integrity-Check-Value >
```

## 2.2  Accounting-Answer (ACA) Command

   The Accounting-Answer command, indicated by the Command-Code field
   set to 272, is used to acknowledge an Accounting-Request command. The
   Accounting-Answer command contains the same Session-Id and MAY
   contains the same Accounting Description and Usage AVPs that were
   sent in the Accounting-Request command. If the CMS-Data AVP was
   present in the Accounting-Request, the corresponding ACA message MUST
   include the CMS-Data AVP signed by the responder to provide strong
   AVP authentication, which MAY be used for the purposes of
   repudiation.

   Only the target Diameter Server, known as the home Diameter Server,
   SHOULD respond with the Accounting-Answer command.

   The AVP listed below SHOULD include service specific accounting AVPs,
   as described in section 1.4.

   Message Format

      <Accounting-Answer> ::= < Diameter Header: 272 >
                             < Session-Id >
                             { Extension-Id }
                             { Result-Code }
                             { Destination-Realm }
                             { Origin-FQDN }
                             { Origin-Realm }
                             { Accounting-Record-Type }
                             { Accounting-Record-Number }
                             { Accounting-Session-Id }
                             [ Error-Reporting-FQDN ]
                             [ Accounting-Interim-Interval ]
                             [ Accounting-Authentication-Type ]
                             [ Accounting-Input-Octets ]
                             [ Accounting-Output-Octets ]
                             [ Accounting-Session-Time ]
                             [ Accounting-Input-Packets ]
                             [ Accounting-Output-Packets ]
                           * [ AVP ]
                             [ CMS-Data ]
                           * [ Proxy-State ]
                           * [ Route-Record ]
                        0*1< Integrity-Check-Value >


## 2.3  Accounting-Status-Ind (ASI) Command

   The Accounting-Status-Ind command, indicated by the Command-Code

field set to 279, is sent by a Diameter node in order to inform its
peer of whether Accounting messages will be sent in the future. A
Diameter node that is about to be taken out of service SHOULD issue
an Accounting-Status-Ind message, with the Accounting-State AVP set
to DISABLED. A Diameter node that detected that it is able to issue
Accounting messages MUST issue an Accounting-Status-Ind message, with
the Accounting-State AVP set to ENABLED.

Message Format

```
   <Accounting-Status-Ind> ::= <Diameter Header: 279 >
                               { Extension-Id }
                               { Origin-FQDN }
                               { Origin-Realm }
                               { Destination-Realm }
                               { Accounting-State }
                             * [ AVP ]
                             * [ Proxy-State ]
                             * [ Route-Record ]
                          0*1< Integrity-Check-Value >
```

## 2.3  Accounting-Poll-Ind (API) Command

The Accounting-Poll-Ind command, indicated by the Command-Code field
set to 273, is sent by a Diameter Server in order to force the peer
to send current accounting data. This data MUST include not yet sent
accounting records from completed sessions, as well as INTERIM_RECORD
records from all ongoing sessions.

Diameter implementations MAY support the Accounting-Poll-Ind command.
An implementation still conforms to this specification if API is not
supported.

The receiver MUST use the Accounting-Request command to send the
accounting data.

The use of Accounting-Poll-Ind is useful in situations where a
Diameter server comes up after an unscheduled downtime, and wishes to
synchronize with the client(s) sooner than at the end of the next
INTERIM_RECORD or at the end of a session.

Warning: The use of the Accounting-Poll-Ind message is discouraged in
roaming networks, since it is unfeasible for a server to attempt to
poll all of it's roaming partner's Diameter peers.

   Message Format

      <Accounting-Poll-Ind> ::= <Diameter Header: 273 >
                                < Session-Id >
                                { Extension-Id }
                                { Destination-FQDN }
                                { Origin-FQDN }
                                { Origin-Realm }
                                { Destination-Realm }
                                { Accounting-Session-Id }
                                [ Destination-FQDN ]
                             * [ AVP ]
                             * [ Proxy-State ]
                             * [ Route-Record ]
                            0*1< Integrity-Check-Value >

## 3.0  Accounting Message Information AVPs

   The following table describes the Diameter AVPs defined in the
   Accounting extension, their AVP Code values, types, possible flag
   values and whether the AVP MAY be encrypted.

|                   |      |         |            | AVP Flag rules |       |       |       |       |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Attribute Name    | AVP Code | Section Defined | Value Type | MUST | MAY | SHLD NOT | MUST NOT | MAY Encr |
| Accounting-Authentication-Type | 45 | 3.6 | Unsigned32 | M | P | | V | Y |
| Accounting-Interim-Interval | 482 | 3.2 | Unsigned32 | M | P | | V | Y |
| Accounting-Record-Number | 485 | 3.3 | Unsigned32 | M | P | | V | Y |
| Accounting-Record-Type | 480 | 3.1 | Unsigned32 | M | P | | V | Y |
| Accounting-Session-Id | 44 | 3.5 | OctetString | M | P | | V | Y |
| Accounting-State | 486 | 3.4 | Unsigned32 | M | P | | V | Y |

## 3.1  Accounting-Record-Type AVP

   The Accounting-Record-Type AVP (AVP Code 480) is of type Unsigned32
   and contains the type of accounting record being sent. The following
   values are currently defined for the Accounting-Record-Type AVP:

   EVENT_RECORD                     1
      An Accounting Event Record is used to indicate that a one-time
      event has occurred (meaning that the start and end of the event
      are simultaneous).  This record contains all information
      relevant to the service, and is the only record of the service.

   START_RECORD                     2
      An Accounting Start, Interim, and Stop Records are used to
      indicate that a service of a measurable length has been given.
      An Accounting Start Record is used to initiate an accounting
      session, and contains accounting information that is relevant
      to the initiation of the session.

   INTERIM_RECORD                   3
      An Interim Accounting Record contains cumulative accounting
      information for an existing accounting session. Interim
      Accounting Records SHOULD be sent every time a re-
      authentication or re-authorization occurs.  Further, additional
      interim record triggers MAY be defined by application-specific
      Diameter extensions. The selection of whether to use
      INTERIM_RECORD records is directed by the Accounting-Interim-
      Interval AVP.

   STOP_RECORD                      4
      An Accounting Stop Record is sent to terminate an accounting
      session and contains cumulative accounting information relevant
      to the existing session.


## 3.2  Accounting-Interim-Interval AVP

   The Accounting-Interim-Interval AVP (AVP Code 482) is of type
   Unsigned32 and is sent from the Diameter authenticating/authorizing
   server to the Diameter client. The client uses information in this
   AVP to decide how and when to produce accounting records. With
   different values in this AVP, service sessions can result in one,
   two, or two+N accounting records, based on the needs of the home-
   organization. The following accounting record production behaviour is
   directed by the inclusion of this AVP:

      1. The omission of the Accounting-Interim-Interval AVP or its
         inclusion with Value field set to 0 means that EVENT_RECORD,
         START_RECORD, and STOP_RECORD are produced, as appropriate for
         the service.

      2. The inclusion of the AVP with Value field set to a non-zero
         value means that INTERIM_RECORD records MUST be produced
         between the START_RECORD and STOP_RECORD records.  The Value

       field of this AVP is the nominal interval between these records
       in seconds. The Diameter node that originates the accounting
       information, known as the client, MUST produce the first
       INTERIM_RECORD record roughly at the time when this nominal
       interval has elapsed from the START_RECORD, the next one again
       as the interval has elapsed once more, and so on until the
       session ends and a STOP_RECORD record is produced.

       The client MUST ensure that the interim record production times
       are randomized so that large accounting message storms are not
       created either among records or around a common service start
       time.


### 3.3  Accounting-Record-Number AVP

   The Accounting-Record-Number AVP (AVP Code 485) is of type Unsigned32
   and identifies this record within one session. As Session-Id AVPs are
   globally unique, the combination of Session-Id and Accounting-
   Record-Number AVPs is also globally unique, and can be used in
   matching accounting records with confirmations.  An easy way to
   produce unique numbers is to set the value to 0 for records of type
   EVENT_RECORD and START_RECORD, and set the value to 1 for the first
   INTERIM_RECORD, 2 for the second, and so on until the value for
   STOP_RECORD is one more than for the last INTERIM_RECORD.


### 3.4  Accounting-State AVP

   The Accounting-State AVP (AVP Code 486) is of type Unsigned32 and is
   used to communicate to a peer whether Accounting messages will be
   sent in the future. A node that issues an ASI with the Accounting-
   State AVP set to DISABLED is informing its peer that it will no
   longer be transmitting Accounting messages until a subsequent ASI
   message is sent with the Accounting-State AVP set to ENABLED.

   The following values have been defined:
      1      ENABLED
      2      DISABLED


### 3.5  Accounting-Session-Id AVP

   The Accounting-Session-Id AVP (AVP Code 44) is of type OctetString,
   and SHOULD be encoded in UTF-8 format [13]. The Accounting-Session-Id
   is not used by the Diameter protocol, since the Session-Id defined in
   [1] is used for both authentication/authorization and accounting
   purposes.  However, a RADIUS/Diameter gateway MAY need to include the

Accounting-Session-Id in Diameter accounting messages.

### 3.6  Accounting-Authentication-Type AVP

The Accounting-Authentication-Type AVP (AVP Code 45) is of type
Unsigned32, and specifies how the user was authenticated. The
following values are supported:
    1       RADIUS
    2       Local
    3       Remote
    4       Diameter

### 4.0  Accounting Usage AVPs

This section contains AVPs that describe accounting usage information
related to a specific session.

```
                                    +---------------------+
                                    |   AVP Flag rules    |
                                    |----+-----+----+-----|----+
                     AVP  Section   |    |     |SHLD| MUST|MAY |
    Attribute Name   Code Defined   Value Type |MUST| MAY | NOT|  NOT|Encr|
    ----------------------------------------|----+-----+----+-----|----|
    Accounting-       42  4.1       Unsigned32 | M  | P   |    | V   | Y  |
       Input-Octets                            |    |     |    |     |    |
    Accounting-       47  4.4       Unsigned32 | M  | P   |    | V   | Y  |
       Input-Packets                           |    |     |    |     |    |
    Accounting-       43  4.2       Unsigned32 | M  | P   |    | V   | Y  |
       Output-Octets                           |    |     |    |     |    |
    Accounting-       48  4.5       Unsigned32 | M  | P   |    | V   | Y  |
       Output-Packets                          |    |     |    |     |    |
    Accounting-       46  4.3       Unsigned32 | M  | P   |    | V   | Y  |
       Session-Time                            |    |     |    |     |    |
```

### 4.1  Accounting-Input-Octets AVP

The Accounting-Input-Octets AVP (AVP Code 42) is of type Unsigned64,
and contains the number of octets in IP packets received by the user.

### 4.2  Accounting-Output-Octets AVP

The Accounting-Output-Octets AVP (AVP Code 43) is of type Unsigned64,
and contains the number of octets in IP packets sent to the user.

## 4.3  Accounting-Session-Time AVP

   The Accounting-Session-Time AVP (AVP Code 46) is of type Unsigned32,
   and indicates the length of the current session in seconds.


## 4.4  Accounting-Input-Packets AVP

   The Accounting-Input-Packets (AVP Code 47) is of type Unsigned64, and
   contains the number of IP packets received by the user.


## 4.5  Accounting-Output-Packets AVP

   The Accounting-Output-Packets (AVP Code 48) is of type Unsigned64,
   and contains the number of IP packets sent to the user.


## 5.0  Result-Code Values

   Errors defined in this section indicates a particular type of failure
   in the accounting data transfer process. Diameter clients MAY use
   this information in deciding a retransmission strategy after the
   error has happened. For instance, an out of space condition isn't
   typically recovered very soon, so a Diameter node might wait longer
   for the next retry than after network or server outage.

      DIAMETER_OUT_OF_SPACE            4007
         A Diameter node received the accounting request but was unable
         to commit it to stable storage due to a temporary lack of
         space.


## 6.0  AVP Table

   The following table presents the AVPs defined in this document, and
   specifies in which Diameter messages they MAY, or MAY NOT be present.
   Note that AVPs that can only be present within a Grouped AVP are not
   represented in this table.

   The table uses the following symbols:
      0       The AVP MUST NOT be present in the message.
      0+      Zero or more instances of the AVP MAY be present in the
              message.
      0-1     Zero or one instance of the AVP MAY be present in the
              message.
      1       One instance of the AVP MUST be present in the message.

```
                              +-----------------------+
                              |      Command-Code      |
                              |-----+-----+-----+-----+
    Attribute Name            | ACR | ACA | API | ASI |
    --------------------------|-----+-----+-----+-----+
    Accounting-Authentication-Type| 1 | 1   | 0   | 0   |
    Accounting-Input-Octets   | 1   | 1   | 0   | 0   |
    Accounting-Input-Packets  | 1   | 1   | 0   | 0   |
    Accounting-Interim-Interval | 0-1 | 0-1 | 0   | 0   |
    Accounting-Output-Octets  | 1   | 1   | 0   | 0   |
    Accounting-Output-Packets | 1   | 1   | 0   | 0   |
    Accounting-Record-Number  | 1   | 1   | 0   | 0   |
    Accounting-Record-Type    | 1   | 1   | 0   | 0   |
    Accounting-Session-Id     | 1   | 1   | 0   | 1   |
    Accounting-Session-Time   | 1   | 1   | 0   | 0   |
    Accounting-State          | 0   | 0   | 1   | 0   |
    Destination-FQDN          | 0+  | 1   | 0+  | 0-1 |
    Destination-Realm         | 1   | 1   | 1   | 1   |
    Error-Reporting-FQDN      | 0   | 0+  | 0   | 0   |
    Extension-Id              | 1   | 1   | 1   | 1   |
    Integrity-Check-Value     | 0-1 | 0-1 | 0-1 | 0-1 |
    Origin-FQDN               | 1   | 1   | 1   | 1   |
    Origin-Realm              | 1   | 1   | 1   | 1   |
    Proxy-State               | 0+  | 0+  | 0+  | 0+  |
    Route-Record              | 0+  | 0+  | 0+  | 0+  |
    Result-Code               | 0   | 1   | 0   | 0   |
    Session-Id                | 1   | 1   | 0   | 1   |
    --------------------------|-----+-----+-----+-----+
```

## 7.0  IANA Considerations

The command codes defined in Section 2.0 are values taken from the
Command-Code [1] address space and extended in [2], [3] and [9].
IANA should record the values as defined in Section 2.0.

The AVPs defined in section 3.0 and 5.0 were alllocated from from the
AVP numbering space defined in [1], and extended in [2], [3] and [9].
IANA should record the values as defined in Section 3.0.

This document introduces the Accounting-Record-Type AVP, which
contains pre-defined values. This document defines the values 1-3.
All remaining values are available for assignment via a Designated
Expert [8].

## 8.0  Security Considerations

This Diameter extension assumes that the accounting data is secured
either through a hop-by-hop authentication mechanism, as described in
[1], or using a strong authentication mechanism as defined in [9].


**9.0  Acknowledgments**

The authors would like to thank Nenad Trifunovic, Tony Johansson and
Pankaj Patel for their participation in the Document Reading Party.
Thanks to the various people that have contributed to accounting
related requirements at the IETF's AAA Working Group and other
related WGs.


**10.0  References**

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

[2]  P. Calhoun, W. Bulley, A. Rubens, J. Haag.  "Diameter NASREQ
     Extension", draft-ietf-aaa-diameter-nasreq-01.txt, IETF work in
     progress, March 2001.

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

[4]  C. Rigney, A. Rubens, W. Simpson, S. Willens, "Remote Authenti-
     cation Dial In User Service (RADIUS)." RFC 2138, April 1997.

[5]  C. Rigney, "RADIUS Accounting." RFC 2139,  April 1997.

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

[7]  D. Harkins, D. Carrel, "The Internet Key Exchange (IKE)", RFC
     2409, November 1998.

[8]  Narten, Alvestrand, "Guidelines for Writing an IANA Considera-
     tions Section in RFCs", BCP 26, RFC 2434, October 1998

[9]  P. Calhoun, W. Bulley, S. Farrell, "Diameter Strong Security
     Extension", draft-calhoun-diameter-strong-crypto-06.txt, IETF
     work in progress, February 2001.

[10] J. Arkko, P. Calhoun, E. Guttman, B. Nelson, B. Wolff, "AAA

        Solutions", draft-ietf-aaa-solutions-01.txt, IETF work in pro-
        gress, November 2000.

   [11] B. Aboba, J. Arkko, D. Harrington. "Introduction to Accounting
        Management", RFC 2975, October 2000.

   [12] A. Shacham, R. Monsour, R. Pereira, M. Thomas, "IP Payload
        Compression Protocol (IPComp)", RFC 2393, December 1998.

   [13] F. Yergeau, "UTF-8, a transformation format of ISO 10646", RFC
        2279, January 1998.

**11.0  Authors' Addresses**

   Questions about this memo can be directed to:

      Jari Arkko
      Oy LM Ericsson Ab
      02420 Jorvas
      Finland

       Phone: +358 40 5079256
      E-Mail: Jari.Arkko@ericsson.com


      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


      Glen Zorn
      Cisco Systems, Inc.
      500 108th Avenue N.E., Suite 500
      Bellevue, WA 98004
      USA

       Phone:  +1 425 438 8218
      E-Mail:  gwz@cisco.com

## [12.0](#) **Full Copyright Statement**

## [13.0](#) **Expiration Date**

This memo is filed as <draft-ietf-aaa-diameter-accounting-01.txt> and
expires in August 2001.