

A Diameter accounting application for the Session Initiation Protocol

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.”

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

To view the list Internet-Draft Shadow Directories, see <http://www.ietf.org/shadow.html>.

Copyright Notice

Copyright (c) The Internet Society (2002). All Rights Reserved.

Abstract

This memo defines a SIP Diameter accounting application. It specifies how Diameter servers can be used with SIP servers to provide accounting. This accounting is expected to be compatible with existing RADIUS servers. Specific Diameter accounting attribute value pairs that are applicable to SIP are discussed, and where necessary new attribute value pairs are defined.

Contents

1	Introduction	3
2	Conventions of this document	3
3	Terminology	3
4	Architecture	4
4.1	Accounting events	4
4.2	Accounting sessions	5
4.3	Mapping signaling messages to Diameter events	5
4.4	Diameter events for media sessions	6
4.5	Other considerations	6
5	Diameter AVP mapping	6
5.1	Acct-Application-Id	6
5.2	Session-ID	6
5.3	Origin-Realm	6
5.4	Origin-Host	7

5.5	Destination-Host	7
5.6	Destination-Realm	7
5.7	Accounting-Authentication-Type	7
5.8	Accounting-Session-Id	7
5.9	Accounting-Realtime-Required	7
5.10	Acct-Multi-Session-Id	7
5.11	Accounting-Sub-Session-Id	7
5.12	Accounting-Record-Number	8
5.13	Accounting-Input-Octets	8
5.14	Accounting-Output-Octets	8
5.15	Accounting-Input-Packets	8
5.16	Accounting-Output-Packets	8
5.17	Accounting-Session-Time	8
5.18	Accounting-Interim-Interval	9
5.19	Authorization-Lifetime	9
6	SIP-service AVPs	9
6.1	Sip-Method	9
6.2	Sip-Response-Code	10
6.3	Sip-From	10
6.4	Sip-To	10
6.5	Sip-Authenticated-User	10
6.6	Sip-Translated-Request-URI	10
6.7	Sip-Remote-IP-Address	10
6.8	Sip-Remote-Port	11
6.9	Sip-Branch	11
6.10	Sip-Content-Body	11
6.11	Sip-Media-Info	11
6.12	Sip-Media-Input-Octets	11
6.13	Sip-Media-Output-Octets	11
6.14	Sip-Media-Input-Packets	12
6.15	Sip-Media-Output-Packets	12
7	SIP-service AVP command table	12
8	Additional result codes	12
9	Security considerations	12
10	Performance considerations	13
11	Limitations	13
12	IANA considerations	14
13	Acknowledgements	14

1 Introduction

The Session Initiation Protocol (SIP) [1] is an application-layer control protocol that can establish, modify and terminate multimedia sessions. A SIP system is composed of a number of logical components such as user agents, proxy servers, redirect servers and registrars.

While SIP can be used to establish a multimedia session (an audio call, for example); details of accounting and billing are beyond its scope. Diameter [2] is a AAA protocol that can be used for carrying accounting information between a SIP server and an accounting server. In this architecture, the SIP server operates as a client of the Diameter server. The client passes user accounting information derived from specific events in a SIP dialog to a designated Diameter server in a Diameter-specific packet. The Diameter server sends back an accounting response to the client indicating that it has successfully received and processed the request.

Many parameters that need to be logged for accounting purposes can be mapped to the AVPs defined in the Diameter accounting extension [3] document. However, new SIP-specific service AVPs are needed for capturing some SIP specific accounting information. This document defines a preliminary set of AVPs. The AVP's defined in this document are expected to be compatible with RADIUS [4].

2 Conventions of this document

In this document, the key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” are to be interpreted as described in RFC 2119 [5] and indicate requirement levels for compliant implementations.

3 Terminology

SIP server: As defined in RFC 3261 [1], SIP servers help in call setup and termination by routing call requests to the user's current location either directly or through a network of intermediate SIP servers. Call setup requests may “fork” (take more than one path) in the network if user is reachable at multiple locations. Call stateful servers behave like PSTN switches in the sense that they keep call state throughout the duration of the call.

SIP network element: A SIP network element (or simply, a SIP element) is any SIP-aware application that needs to use Diameter for accounting purposes. SIP network elements include SIP servers, B2BUAs [1], IP telephony gateways and SIP user agents.

Dialog: RFC 3261 defines a SIP dialog as a peer-to-peer SIP relationship between two UAs that persists for some time. A dialog is established by SIP messages, such as a 2xx response to an INVITE request. A dialog is identified by a call identifier, local tag, and a remote tag. A dialog was formerly known as a call leg in RFC 2543.

Diameter server: It is an entity that implements the Diameter protocol [2]. In the context of this document, Diameter servers provide accounting services based on Diameter accounting AVPs [3] and SIP-specific service AVPs. While the SIP element can use other services of the Diameter server such as user authorization [6], this document restricts itself to discussing protocol mechanisms needed to

provide perform accounting. Thus, for normal accounting related scenarios, a trust relationship is assumed to exist between a SIP network element and its “home” Diameter server.

Accounting records: These are messages composed of accounting AVPs that are sent by the SIP element to the Diameter server containing details of accounting events. Different types of records are sent depending on the state of the particular dialog.

AVP: A Diameter object is encapsulated by a header that describes the contents of the object. This header is known as an attribute-value pair or AVP. Each AVP is identified by a command code.

Accounting-Request (ACR): As defined in [2], this is a command used to exchange accounting information between Diameter nodes. The packet includes Diameter accounting AVPs and SIP-service specific AVP’s defined in Section 6. Unless stated otherwise, the term ACR packet refers to a Accounting-Request command sent by the SIP element.

Accounting-Answer (ACA): The Accounting-Answer command contains the response to the ACR.

4 Architecture

In this section we formally define the notion of “accounting” [7] for SIP elements, and discuss the model for mapping SIP-related events to Diameter accounting records.

4.1 Accounting events

SIP-controlled media sessions can be accounted for in a variety of ways, depending on the types of elements that are available for accounting, the precision needed and the charging model. There is not likely to be a single “right” model, so we take the approach that the SIP element should provide sufficient information to the Diameter infrastructure so that it can support a wide variety of choices. This entails reporting as much useful information as possible. That way, policy decisions are deferred to the users of the AAA service, typically the service provider, rather than the SIP elements. There are likely to be far fewer AAA elements than SIP elements, so that this approach scales better.

In circuit-switched networks, accounting by call duration leaves little opportunity for variation except as to who pays. In SIP sessions, the number of variations is far larger.

Examples of policies that can be supported by the mechanisms in this document include:

Session duration: For session duration accounting, only the time elapsed between the beginning and end of a session is accounted for. There are several plausible candidates for starting points, such as the initial INVITE request, the first 1xx response, the first 2xx response or the confirming ACK request. Media exchange can occur before the 200 response or the ACK request, so that such accounting does not necessarily reflect the duration of the media sessions.

With session duration accounting, one should be aware that this opens up opportunities for cheating unless the media flow is directly controlled by these operations. Such control could involve termination of IP telephony gateways sessions, removal of firewall permissions or removal of preferential quality-of-service treatment. For example, two consenting applications could continue to exchange media even after one has sent a BYE request. Thus, pure session-based accounting is at best an approximation where users cannot be counted on to be cooperative.

Media duration: For media duration accounting, the time from the first to the last media packet is accounted for. Since media streams typically do not have an end indication, determining the instance of the final media packet is difficult and may not be tied directly to SIP signaling events. Thus, such accounting may be better performed in conjunction with a resource reservation protocol or a traffic filter.

Media volume: For media volume accounting, the sum of all media and possibly media control (e.g., RTCP [8]) packets is accounted for. This approach has the same difficulty as media duration accounting, but can be improved by providing for intermediate events that update counters periodically. A timeout mechanism would then treat long gaps as indications that a particular accounting session has ended. As before, SIP-related signaling events may not necessarily be appropriate indications. In some cases, such as media (PSTN) gateways or conferencing servers, SIP signaling is likely to be highly correlated to the beginning and end of media flows.

We distinguish two kinds of SIP-Diameter elements, namely media-aware and media-unaware. Media-aware entities can count bytes and packets of at least some of the media streams related to a SIP dialog. Media unaware entities can only access signaling information. IP telephony gateways and user agents in general are typically media-aware, while SIP proxy servers are commonly media-unaware.

We take the position that accounting for the volume of SIP messages, rather than just the media sessions initiated by them, is useful. In particular, it may become necessary to avoid an arms race where users are tempted to add user-to-user information to signaling messages to avoid being charged for, say, a short message service.

4.2 Accounting sessions

When accounting a SIP session, we first need to determine the “accounted” user. The network can charge the caller or the callee based on URI’s present in the SIP message headers or obtain a trusted identity such as network asserted identity [9] using any authentication process. It is a policy decision on which one to use, and we assume that the SIP element can determine an identity which can then be used as a network access identifier [10] for accounting purposes.

Once the accounted entity has been determined, the SIP element establishes a Diameter session with its local AAA server and could potentially authorize the user [6] [12]. It then sends an Accounting-Request using the mechanisms for server discovery, security, and failover discussed in [2]. Since it is quite likely that the local AAA server is not necessarily the home AAA server for the accounted user, Diameter routing procedures will be used by the local AAA server to deliver the ACR request to the home AAA server.

Sessions between the SIP element and the Diameter server are distinguished by the SIP Call-ID. This value is placed in the Session-ID AVP of all ACR packets generated by the SIP entity.

4.3 Mapping signaling messages to Diameter events

For SIP sessions that can be accounted using “session duration accounting” the initial INVITE signifies start of the session, and a terminating message such as a BYE indicates a stop. Thus, when an INVITE is processed the SIP element SHOULD send an ACR with the Accounting-Record-Type AVP set to START_RECORD. Similarly when a BYE is processed it SHOULD send a STOP_RECORD. For all other intermediate messages such as ACK, 180 and other responses it MAY send an INTERIM_RECORD.

SIP signaling messages that are not necessarily preceded by an INVITE such as MESSAGE and NOTIFY MUST be treated as one-time events using the EVENT_RECORD Accounting-Record-Type AVP.

All SIP entities that act as Diameter clients MUST implement the client side state machine of [2] (Section 8.2). Similarly, Diameter servers supporting the SIP accounting application SHOULD be operated in stateless accounting mode.

4.4 Diameter events for media sessions

Media-aware SIP elements MAY generate periodic accounting events that indicate the status of a dialog using the INTERIM_RECORD Accounting-Record-Type AVP. The accounting record SHOULD contain information about the number of media octets transmitted so far and the elapsed session duration. Whether to generate interim accounting events indicating session progress or generate only a session summary information (using a STOP_RECORD) when the dialog ends is a policy decision.

4.5 Other considerations

Two SIP protocol features that need addressing are retransmissions and forking at proxies. Request and response retransmissions are normal occurrences in SIP signaling due to, for example, the use of UDP. Retransmission messages will also be recorded as accounting events in the case of stateless SIP servers. The Accounting-Record-Number AVP provided by Diameter helps distinguish the original request from subsequent retransmissions as explained in Section 5.12.

Start, Stop and Interim records MAY be generated for each branch of a forking call. Thus, if an INVITE gets forked at a proxy into two branches, then the proxy needs to generate two sets of Start, Interim, and Stop events. The Sip-Branch AVP defined in Section 6 helps group branch-related information for a transaction.

5 Diameter AVP mapping

In this section we outline the semantics of Diameter AVPs [2], [3] that are present in ACRs generated by SIP elements. Unless otherwise indicated, the protocol command table given in Section 10.2 of [2] is applicable to the SIP accounting application.

5.1 Acct-Application-Id

This AVP identifies the particular service generating the accounting records. For SIP elements this corresponds to the SIP Application-ID (value TBD).

5.2 Session-ID

This is equal to the SIP Call-ID for the SIP accounting application.

5.3 Origin-Realm

The Origin-Realm AVP SHOULD have the domain-of-record value of the SIP element.

5.4 Origin-Host

The Origin-Host AVP **MUST** have either the FQDN of the SIP element or an IP address that is reachable from the AAA server.

5.5 Destination-Host

The Destination-Host AVP **MAY** be present in ACR packets only if the SIP element can determine the host name of the AAA server that this request is destined to, as in the case of AAA requests for local users. If the network access identifier does not fall in the home domain(s) [1] of the SIP server, then this AVP **SHOULD** not be present in ACR packets.

5.6 Destination-Realm

The Destination-Realm AVP **MUST** be set to the domain portion of the network access identifier of the accounted user (see Section 4.2).

5.7 Accounting-Authentication-Type

The accounting authentication type AVP **MUST** be present in all ACR packets. If authentication other than RADIUS or Diameter are used such as SIP Digest authentication then the following rule **SHOULD** be used to determine its value. The SIP element takes the domain-of-record portion of the authentication identity and checks whether it belongs to a local domain. If it is a local domain, then it uses the value **Local**. If not, it uses the value **Remote**. This is summarized in the table below (reproduced from [3]).

- 1 RADIUS
- 2 Local
- 3 Remote
- 4 Diameter

5.8 Accounting-Session-Id

This attribute **MUST** be equal to the value of Diameter Session-ID AVP. It **MAY** be present in ACR packets.

5.9 Accounting-Realtime-Required

To be added.

5.10 Acct-Multi-Session-Id

To be added.

5.11 Accounting-Sub-Session-Id

To be added.

5.12 Accounting-Record-Number

This AVP **MUST** be present in all ACR packets. This field **SHOULD** contain an integer value that is locally unique. Use of an unique value aids in correlating requests with Diameter server responses. The **RECOMMENDED** way to generate this is by using a sequence number that is incremented by one for every ACR packet sent to the Diameter server. Any subsequent retransmission of the ACR packet uses the same sequence number used originally.

This AVP also helps in matching SIP request and response retransmissions. SIP stateless servers log all retransmissions since they do not have the ability to distinguish the original request from the duplicate ones. The value for this AVP will change between the original and the subsequent messages and hence helps detect SIP-level retransmissions.

5.13 Accounting-Input-Octets

This AVP **MUST** be present in all ACR packets. SIP elements **SHOULD** set this to the size of the signaling message that triggered this accounting event. Non-signaling related accounting events such as media events **MUST** set this field set to zero. Similarly, if the SIP entity initiated a SIP dialog (such as a SIP server acting as a Presence Agent, generating a NOTIFY), this field **MUST** be zero.

5.14 Accounting-Output-Octets

This AVP **MUST** be present in all ACR packets. SIP elements **SHOULD** set this to the size of the signaling message that was sent in response to this signaling event. For messages for which no response is sent this value **MUST** be set to zero. Non-signaling related accounting events such as media events **MUST** set this field to zero. If a forking proxy chooses to treat all branches as related to a single event, then it **SHOULD** set this field to the sum of all outgoing SIP messages.

5.15 Accounting-Input-Packets

This AVP **MUST** be present in ACR packets. This **MUST** be set to the number of packets received for the SIP event related to this accounting record. In almost all cases, this field will have a value of 1. Media-related events **MUST** set this field to zero. SIP events that initiate a dialog **MUST** set the value to zero.

5.16 Accounting-Output-Packets

This AVP **MUST** be present in an ACR packet. This is set to the number of SIP signaling messages sent in response to the action that triggered this accounting event. The SIP element **SHOULD** set this field to zero if it does not maintain enough state to compute this value.

5.17 Accounting-Session-Time

This AVP **MUST** be present in ACR packets. The value of this field **MUST** be set to zero in all ACR packets that have the Accounting-Record-Type AVP set to **EVENT_RECORD**. The value of this field **SHOULD** also be set to zero for any element that does not maintain enough state to compute a "session duration" as in SIP stateless servers or transaction stateful proxy servers.

For stateful elements, such as a SIP call stateful server or a media gateway, that use the Start, Interim and Stop event model, the value is calculated as follows. When the **START_RECORD** AVP is sent this field

MUST be set to zero, and the time value at that instant is saved as part of the session state. For all subsequent ACR packets, the field contains the difference in seconds between the current time and the time at which the START_RECORD was sent.

5.18 Accounting-Interim-Interval

A media-aware SIP element that receives an Accounting-Interim-Interval from a Diameter server MAY follow the rules of Section 3.2 of [3] and generate accounting records with the Accounting-Record-Type AVP set to INTERIM_RECORD to report status of a dialog. When such an interim record is generated, it MUST contain the AVPs listed below.

```
Sip-Input-Media-Octets  
Sip-Output-Media-Octets  
Sip-Input-Media-Packets  
Sip-Output-Media-Octets
```

Media-unaware SIP elements SHOULD ignore this AVP.

5.19 Authorization-Lifetime

This AVP will govern dialog lifetime, and the SIP element behavior.

To be added.

6 SIP-service AVPs

This section defines a preliminary set of service specific AVPs for the SIP accounting application. A protocol command table is given for ease of reference. All service specific AVPs MUST have their length parameter set properly. In the default operation mode, the 'M' bit in the AVP header will be set, and the 'V' bit will not be set for the service AVPs defined below. Additional AVPs may be defined in a subsequent revision to accommodate information that needs to be captured while offering services such as 3-way calls or credit-based charging. In addition to the AVPs listed in the section, additional SIP authorization related AVPs [6] may be necessary for a SIP element to successfully use a AAA server.

6.1 Sip-Method

The Sip-Method AVP (AVP code TBD) is of type Enumerated and indicates the SIP method.

The Value field enumeration is defined below:

0	Unknown
1	INVITE
2	BYE
3	REGISTER
4	CANCEL
5	OPTIONS
6	ACK

7 SUBSCRIBE
8 NOTIFY
9 MESSAGE

6.2 Sip-Response-Code

The Sip-Response-Code AVP (code TBD) has a type Unsigned32 and indicates the SIP Response code/Status code present in the header of the SIP response to the SIP element. For example, a successful call setup may result in 200.

Accounting records generated in response to a SIP signaling event **MUST** have either the Sip-Method AVP or the Sip-Response-Code AVP.

6.3 Sip-From

The Sip-From AVP (code TBD, type UTF8String) is the URL present in SIP From header including the tag. This AVP **MUST** be present in all ACR packets. The display name **MAY** be ignored.

6.4 Sip-To

The Sip-To AVP (code TBD, type UTF8String) is the URL present in the SIP To header including the tag, if present. This AVP **MUST** be present in all ACR packets. The display name **MAY** be ignored.

6.5 Sip-Authenticated-User

The Sip-Authenticated-User (code TBD, type UTF8String) is the identity of the “accounted entity” determined by the SIP element. It **MAY** be present in an ACR packet.

6.6 Sip-Translated-Request-URI

The Sip-Translated-Request-URI AVP (code TBD, type UTF8String) indicates the Request-URI of the SIP request, translated as per the SIP server’s processing rules into a “canonical” URI. For an INVITE request, the “canonical” URI is the URI that the SIP server uses for proxying. For example, if the Request-URI is sip:alice@wonderland.com, a SIP server might translate this to sip:alice@p42.wonderland.com. The latter is then called as the translated request URI.

In other cases, such as a REGISTER request, the Sip-Translated-Request-URI **MAY** be same as the Request-URI. For example, if the Request-URI for the registrar serving wonderland.com is sip:wonderland.com, the Sip-Translated-Request-URI will be just wonderland.com. However, other Request-URIs such as sip:registrar.wonderland.com **MAY** be translated to sip:wonderland.com.

This AVP **SHOULD** be present in all ACR packets that also have a Sip-Method.

6.7 Sip-Remote-IP-Address

The Sip-Remote-IP-Address AVP (code TBD, type IPAddress) indicates the IP address of an upstream entity such as a SIP-UAC or another SIP proxy if any, that sent the request/response which triggered this particular accounting request. This **MUST** be present in all ACR packets that have either the Sip-Method or the Sip-Response-Code AVP present.

6.8 Sip-Remote-Port

The Sip-Remote-Port AVP (code TBD, type Unsigned32) indicates the port number of an upstream entity such as a SIP-UAC or another SIP proxy from which the request or response that triggered this particular accounting request was received. For media aware SIP elements, this indicates the port number of the upstream entity from which media packets are received. This **MUST** be present in all ACR packets that have the Sip-Remote-IP-Address AVP present.

6.9 Sip-Branch

The Sip-Branch AVP (code TBD, type UTF8String) **SHOULD** be present in all the accounting records related to a dialog, which involved forking at the SIP proxy server. The rules for generating the branch identifier are discussed in [1]. This AVP **MAY** be present in ACR packets related to a call that was not forked.

6.10 Sip-Content-Body

The Sip-Content-Body AVP (code TBD, type OctetString) contains the content body of the message. The AVP-Length field **MUST** be set to Content-Length + 8 (Content-Length + 12, if the 'V' bit is enabled). This **MAY** be present in ACR packets.

6.11 Sip-Media-Info

The Sip-Media-Info AVP (code TBD) is a Grouped AVP containing four AVPs related to media streams. The ABNF specification for the Sip-Media-Info AVP is given below.

```
Sip-Media-Info ::= <AVP Header: TBD>
                { Sip-Media-Input-Octets }
                { Sip-Media-Input-Packets }
                { Sip-Media-Output-Octets }
                { Sip-Media-Output-Packets }
                * [ AVP ]
```

The individual media AVPs are discussed next.

6.12 Sip-Media-Input-Octets

The Sip-Media-Input-Octets AVP (code TBD, type Unsigned64) contains the total number of media octets received by the SIP element from in the SIP dialog, from the start of a session until the instant this accounting record is generated.

6.13 Sip-Media-Output-Octets

The Sip-Media-Output-Octets AVP (code TBD, type Unsigned64) contains the total number of media octets sent by the SIP element in the SIP dialog, from the start of a session until the instant this accounting record is generated.

6.14 Sip-Media-Input-Packets

The Sip-Media-Input-Packets AVP (code TBD, type Unsigned32) contains the total number of media packets received by the SIP element in the SIP dialog, from the start of a session until the instant this accounting record is generated.

6.15 Sip-Media-Output-Packets

The Sip-Media-Output-Packets AVP (code TBD, type Unsigned32) contains the total number of media packets sent by the SIP element in the SIP dialog, from the start of a session until the instant this accounting record is generated.

7 SIP-service AVP command table

Attribute Name	Command Code	
	ACR	ACA
Sip-Method	0-1	0
Sip-Response-Code	0-1	0
Sip-From	1	0
Sip-To	1	0
Sip-Authenticated-User	0-1	0
Sip-Translated-Request-URI	0-1	0
Sip-Remote-IP-Address	1	0
Sip-Remote-Port	1	0
Sip-Branch	0-1	0
Sip-Content-Body	0-1	0
Sip-Media-Info	0-1	0

8 Additional result codes

Result codes that are supported by the SIP accounting application in addition to the base protocol result codes are listed below.

To be added.

9 Security considerations

If a SIP proxy server is used for accounting, the proxy SHOULD use the SIP Record-Route option during dialog setup to ensure that all subsequent signaling messages traverse through it. This is needed, for example, to know when the dialog ends. Security policies should make sure that the proxy server is not bypassed.

For example, a gateway should be configured to reject all BYE requests that do not originate from the proxy server. Additional security issues considered in Diameter protocol document [2], and RFC 3261 [1] are also applicable.

10 Performance considerations

In this SIP-Diameter accounting model, the SIP element sends the ACR to the local AAA server, which is expected to route the accounting request to the appropriate home AAA server if necessary. This routing process could be non real-time. The SIP element needs to maintain accounting request state until it receives a response from the home Diameter server. This can impose performance penalties in a stateless SIP proxy that encounters more visiting users.

Careful thought should thus be given to two aspects namely, whether a SIP server participates in accounting and logging, and if so what types of messages are logged. SIP elements could choose to use a filter rule [2] that describes which of the SIP messages, if any, are recorded.

11 Limitations

We still need to address the case when multiple media such as voice, video, and data are used in a session [11] or support advanced services such as prepaid calling cards or 3-way calling. Below, we outline some possible approaches.

In order to implement authorization based services such as prepaid calling cards, one can leverage the Authorization-Lifetime AVP coupled with any SIP-Diameter Authorization mechanism (such as [6]). Thus, when the session for a particular user (identified by his Network access identifier) is established with the Diameter server, the Diameter server indicates the duration until which the user is authorized to use network resources (such as the media gateway) after which the user needs to be reauthorized by contacting the Diameter server. While this model is well-suited for media-aware entities, it is not applicable for media-unaware SIP servers. Plus, this assumes that billing is based on minutes instead of volume.

While dealing with multiple media streams in a single dialog, one can leverage the existing Accounting-Sub-Session-Id AVP to distinguish among streams. However, this also requires media-aware elements to deal with a “state table” mapping the typical 6-tuple representation of a stream (namely, the source and destination IP addresses and ports, the transport protocol, and plausibly the RTP payload type) to the Accounting-Sub-Session-Id.

We have proposed a signaling(SIP)-oriented approach for multimedia session accounting. For sophisticated media-based volume accounting such as pricing based on media type, we need to separate the media AVPs from SIP AVPs. The media and SIP AVPs are then associated using the SIP Call-ID. New media specific AVPs need to be defined that will have sufficient richness to cover all the information that one would be normally interested pertaining to a RTP stream. This approach also allows reuse of media AVPs for other applications like RTSP media servers that use RTP/RTCP for media transport.

Issues that may arise due to SIP call control, for example using the REFER method, remain to be discussed.

12 IANA considerations

A SIP application identifier needs to be obtained from IANA.

Command codes need to be assigned for the new SIP service specific AVPs defined in this document.

13 Acknowledgements

Kundan Singh provided valuable comments.

14 Authors' addresses

Sankaran Narayanan¹
One Microsoft Way
Microsoft Corporation
Redmond, WA, 98052.
electronic mail: sankaran@windows.microsoft.com

Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue, MC 0401
New York, NY 10027
USA
electronic mail: schulzrinne@cs.columbia.edu

Anshul Kundaje
Dept. of Electrical Engineering
Columbia University
1214 Amsterdam Avenue, MC 0401
New York, NY 10027
USA
electronic mail: abk2001@cs.columbia.edu

References

- [1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: session initiation protocol," RFC 3261, Internet Engineering Task Force, June 2002.
- [2] P. Calhoun *et al.*, "Diameter base protocol," Internet Draft, Internet Engineering Task Force, July 2002. Work in progress.
- [3] J. Arkko, P. Calhoun, P. Patel, and G. Zorn, "DIAMETER accounting extension," Internet Draft, Internet Engineering Task Force, Feb. 2001. Work in progress.

¹Work done while the author was with Department of Computer Science, Columbia University.

- [4] C. Rigney, S. Willens, A. Rubens, and W. Simpson, "Remote authentication dial in user service (RADIUS)," RFC 2865, Internet Engineering Task Force, June 2000.
- [5] S. Bradner, "Key words for use in RFCs to indicate requirement levels," RFC 2119, Internet Engineering Task Force, Mar. 1997.
- [6] T. Johansson *et al.*, "Diameter multimedia application," Internet Draft, Internet Engineering Task Force, Feb. 2002. Work in progress.
- [7] B. Aboba, J. Arkko, and D. Harrington, "Introduction to accounting management," RFC 2975, Internet Engineering Task Force, Oct. 2000.
- [8] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force, Jan. 1996.
- [9] M. Watson, "Short term requirements for network asserted identity," Internet Draft, Internet Engineering Task Force, June 2002. Work in progress.
- [10] B. Aboba and M. Beadles, "The network access identifier," RFC 2486, Internet Engineering Task Force, Jan. 1999.
- [11] J. Loughney and G. Camarillo, "SIP-AAA requirements," Internet Draft, Internet Engineering Task Force, July 2002. Work in progress.
- [12] H. Basilier, P. Calhoun, M. Holdrege, T. Johansson, J. Kempf, and J. Rajaniemi, "AAA requirements for IP telephony/multimedia," Internet Draft, Internet Engineering Task Force, Mar. 2002. Work in progress.

Full Copyright Statement

Copyright (c) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.