

Radext
Internet-Draft
Expires: January 9, 2005

C. Guenther
H. Tschofenig
Siemens
July 11, 2004

**Prepaid Extensions to Radius for Event-based Charging
draft-guenther-radext-ppebc-00.txt**

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of [section 3 of RFC 3667](#). By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with [RFC 3668](#).

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>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on January 9, 2005.

Copyright Notice

Copyright (C) The Internet Society (2004). All Rights Reserved.

Abstract

In event-based charging procedures, customers get charged for service usage per se. This type of charging can be independent of data volume transferred, time period of service availment, or user subscription status. This memo introduces Radius attributes appropriate for event-based charging with debiting of prepaid user accounts.

Table of Contents

1.	Introduction	3
2.	Terminology	4
3.	Framework	5
4.	Use Cases and Message Flows	8
4.1	Price Enquiry	8
4.2	Direct Debiting	9
4.3	Amount Reservation	10
4.4	Amount Capture	11
5.	New Radius Attributes for Event-based Charging	13
5.1	Service-Name	13
5.2	Requested-Action	13
5.3	Cost	14
5.4	Currency-Code	15
5.5	Charging-Session-Id	16
5.6	International Mobile Subscriber Identity (IMSI)	17
6.	Radius Messages for Event-based Charging	19
6.1	Price Enquiry Request	19
6.2	Price Enquiry Response	20
6.3	Direct Debiting Request	20
6.4	Direct Debiting Response	21
6.5	Amount Reservation Request	21
6.6	Amount Reservation Response	21
6.7	Amount Capture Request	22
6.8	Amount Capture Response	22
6.9	Reject	22
7.	IANA Considerations	24
8.	Security Considerations	25
9.	References	26
9.1	Normative References	26
9.2	Informative References	26
	Authors' Addresses	26
	Intellectual Property and Copyright Statements	28

1. Introduction

There are several models of how to charge customers for availing data services:

Volume-based charging (VBC): (e.g., 2 Cent/KiloByte),
Duration-based charging (DBC): (e.g., 3 Cent/minute),
Subscription-based charging (SBC): (e.g., 5 Dollar/month+service),
Event-based charging (EBC): (e.g., 7 Cent/URL or email).

Charging models can be further divided into those with debiting of prepaid user accounts and those with debiting of non-prepaid accounts (such as current accounts at banks).

Volume- and time-based charging with debiting of prepaid accounts is being treated in [I-D.[draft-lior-radius-prepaid-extensions-03](#)] by defining appropriate attributes for the Remote Authentication Dial-In User Service (Radius). In event-based charging procedures, customers get charged for service usage per se. This type of charging can be independent of data volume transferred, time period of service availment, or user subscription status. This memo introduces Radius attributes appropriate for event-based charging with debiting of prepaid user accounts.

2. Terminology

This document uses the following terms and acronyms:

Radius Server: This is a server that offers the backend infrastructure for authentication and authorization via the protocol described in [[RFC2865](#)], and for accounting as described in [[RFC2866](#)].

Radius Client: This entity is responsible for passing user information to designated Radius Servers and then for acting on the responses received.

Event: The occasion that triggers the execution of a charging procedure in relation to data service availment. Example: An http request demanding access to a chargeable data service.

Volume-based Charging (VBC): A service usage charging procedure that is based on the data volume transferred to the service user for the purpose of service execution. Example: 2 Cent/KByte.

Duration-based Charging (DBC): A service usage charging procedure that is based on the time period the user avails the service. Example: 3 Cent/minute.

Subscription-based Charging (SBC): A service usage charging procedure that is based on the fact that the user has previously subscribed to the service in question. Example: 5 Dollar/month and service.

Event-based Charging (EBC): A service usage charging procedure that is based on service availment per se. Example: 7 Cent/URL.

Event Handler (EH): The network entity that is responsible for detecting chargeable events (e.g., an http request for a value content) and for deciding which type of charging (e.g., VBC, DBC, EBC, SBC, or combination of them) is to be employed. From the Radius point of view, this entity acts as Radius Client.

Rating Entity (RE): The network entity that accounts for calculating cost information regarding a given data service. From a Radius perspective, this entity acts as a Radius Server.

Charging Handler (CH): The network entity that is responsible for executing charging-related tasks such as, for instance, price enquiry and debiting. From the Radius point of view, this entity can act both as Radius Server and Client.

Accounts Database (AD): The network entity that supplies the Charging Handler with information regarding user accounts (e.g., account balance information). Within the scope of this specification, the AD is concerned with prepaid user accounts only.

Service Provider (SP): The network entity that offers a chargeable data service. Access requests to chargeable data services of a SP are detected and handled by the Event Handler.

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

3. Framework

The Radius messages defined in this memo transfer information related to event-based charging among network entities such as an Event Handler (EH), a Rating Entity (RE), a Charging Handler (CH), and a Service Provider (SP). The possible communication architectures for these Radius messages can vary in terms of Radius message interaction between the entities EH, RE, CH, and SP. Figure 1 depicts a basic network structure in which the RE and CH are separated entities acting as Radius Servers towards the EH which acts the role of a Radius Client:

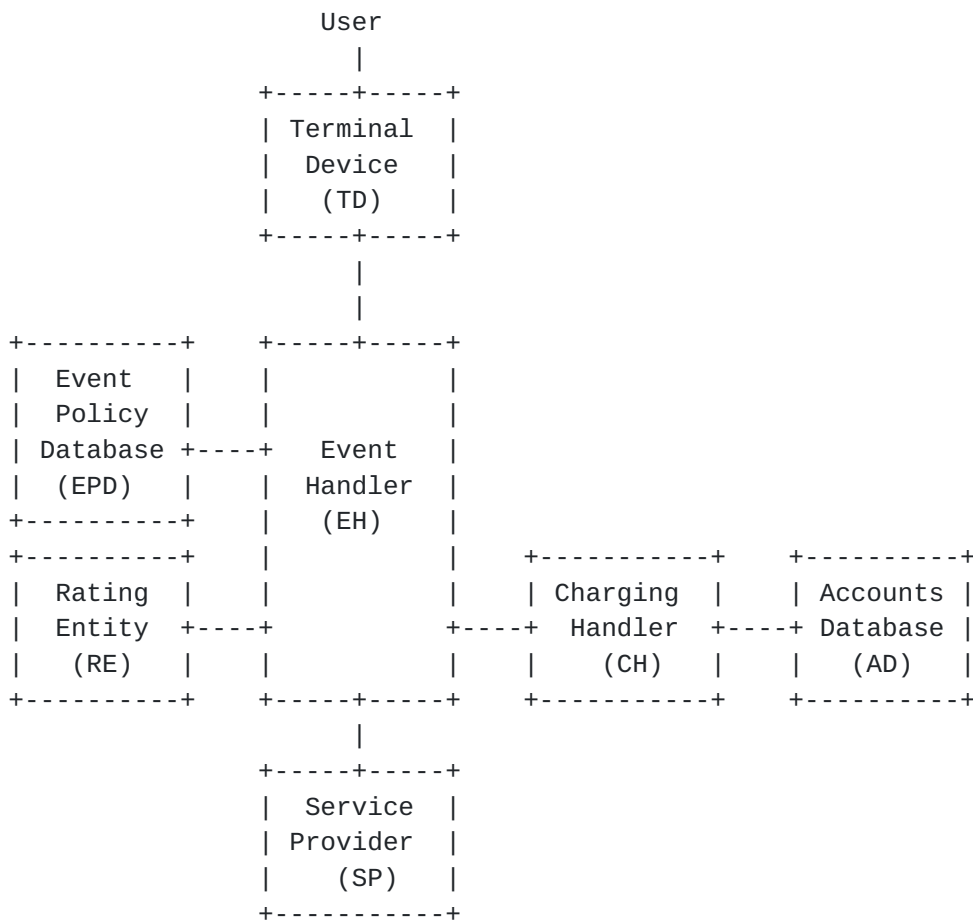


Figure 1: Framework with EH-RE Connection

The basic steps of operation in this network topology and its variations are the following: first, the user requests access to a certain data service. A user, for example, enters an URL into his or her web browser, selects an appropriate link, or clicks on a user menu item.

The EH permanently scans the service access requests it is

responsible for (e.g., it scans all http requests) in order to sort out requests for chargeable events. The EH falls back to an Event Policy Database (EPD) which helps distinguishing between chargeable and non-chargeable events and - in case of chargeable events - also helps deciding which type of charging (e.g., VBC, DBC, SBC, EBC, or combinations of them) is to be employed. A Rating Entity (RE) supplies the EH with cost information related to given data services (price enquiry).

This specification is dedicated to the case of event-based charging with debiting to prepaid user accounts. In this case, the Charging Handler (CH) accounts for the following tasks:

Debiting: To debit the user's prepaid account directly (i.e., without amount reservation) prior to service delivery or afterwards,
Amount Reservation: To reserve a certain amount of money from the user's prepaid account (not applicable in usage scenarios with direct debiting, i.e., without amount reservation), and
Amount Capture: To capture a reserved amount of money after successful service execution (not applicable in usage scenarios with direct debiting).

Prior to service execution, the user will usually get informed about the terms for service availment (especially, costs). If (s)he accepts these conditions, the desired service is delivered to the user.

In a variation of the first architectural model as shown in Figure 1, the RE has no direct Radius connection to the EH but builds up a Radius Server/Client pair along with the CH. Towards the EH, the CH acts as Radius Server:

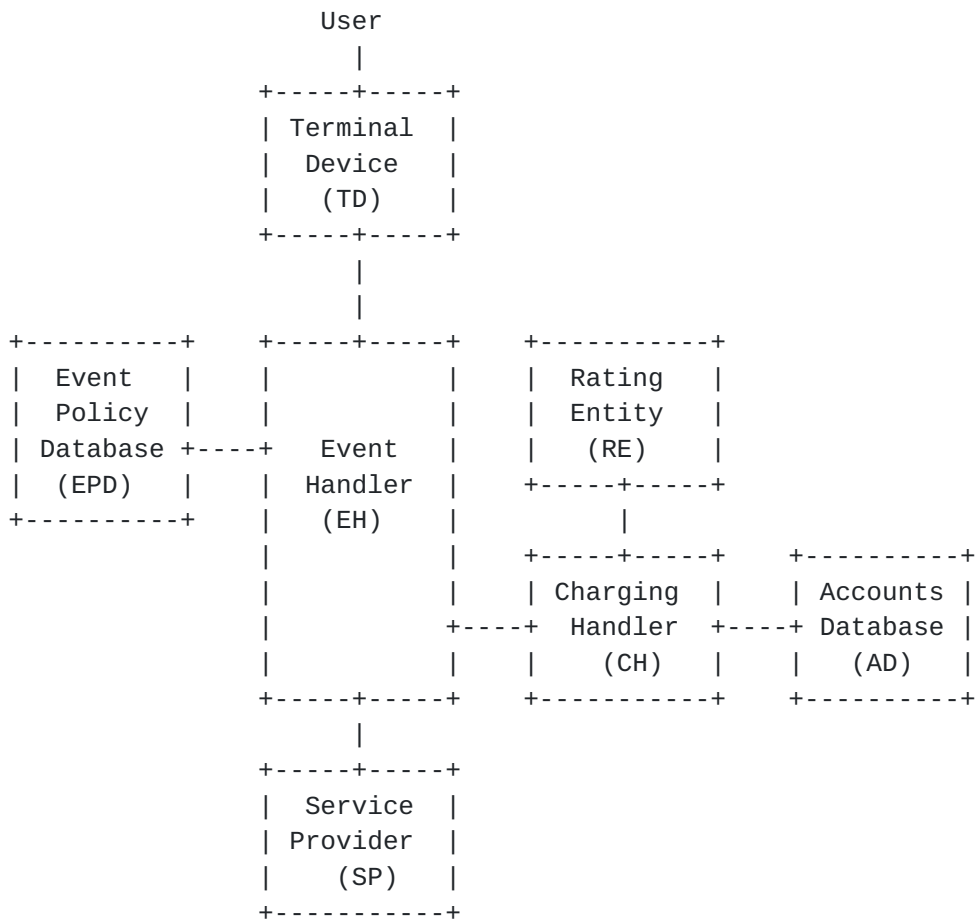


Figure 2: Framework with CH-RE Connection

The basic steps of operation in this model are mostly the same as for the first model - but this time, the EH does not only leave charging of user accounts but also rating of events to other network entities.

4. Use Cases and Message Flows

This section describes some use cases in which the Radius attributes and messages specified in this memo are helpful: price enquiry, direct debiting, amount reservation, and amount capture.

4.1 Price Enquiry

The RE is responsible for calculating price information related to a given data service. Depending on which Radius connections the RE has to the other network entities, the EH (see Figure 1), the CH (see Figure 2), or the SP can request this type of information by means of an Price Enquiry Request message. If no failure of any kind occurs, the RE sends a Price Enquiry Response back to the Charging Handler:



Figure 3: Price Enquiry Message Flow: Successful Case

In case of an error (e.g., the RE is not able to calculate the desired information, or the data provided by the EH are not sufficient to process the Price Enquiry Request appropriately), the RE will respond with a Reject message:

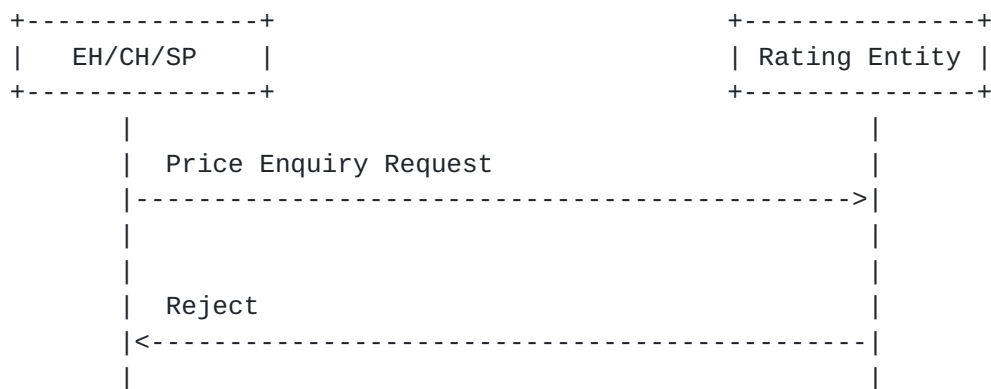


Figure 4: Price Enquiry Message Flow: Failure Case

4.2 Direct Debiting

Debiting of prepaid accounts can be preceded by reserving a sufficient amount from the prepaid account or can go without such an amount reservation. The latter case is referred to as 'direct debiting' which can occur prior to service execution or afterwards.

This specification defines Radius messages suitable for direct debiting initiation (request) and confirmation (response). These are exchanged between an Event Handler (EH) and a Charging Handler (CH) as follows:

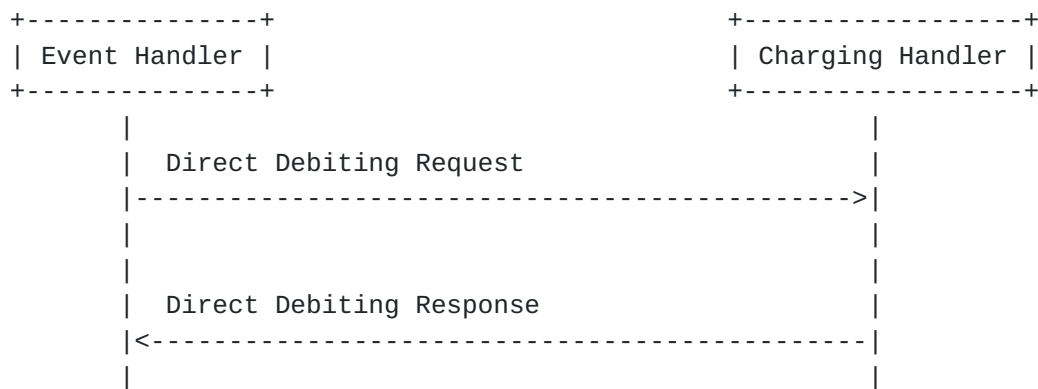


Figure 5: Direct Debiting Message Flow: Successful Case

Of course, the CH will first check whether the prepaid user account has sufficient cover. If this is not the case, the CH will substitute its response message by an error message:

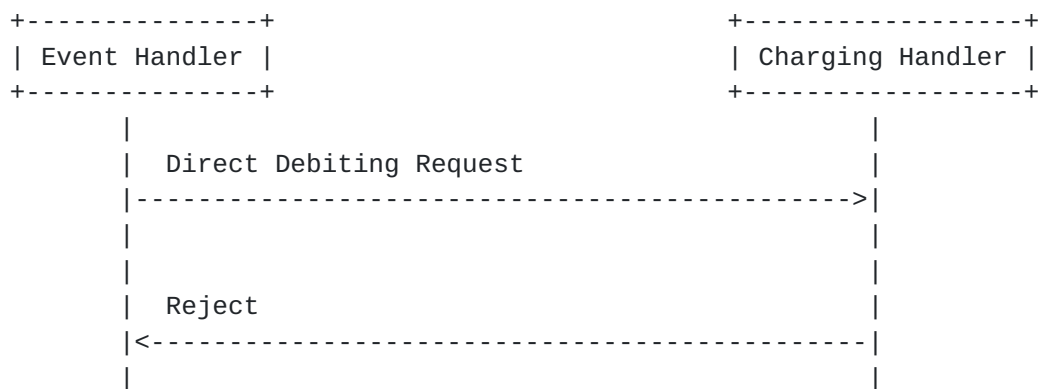


Figure 6: Direct Debiting Message Flow: Failure Case

4.3 Amount Reservation

Besides messages for direct debiting, this specification also defines Radius messages for use cases with reservation of amounts of money (or of non-monetary units; to be detailed in future versions of this document) from user accounts. Reserved amounts are then captured at a later point of time. Amount reservation is initiated by an Amount Reservation Request and confirmed in case of success by an Amount Reservation Response as follows:

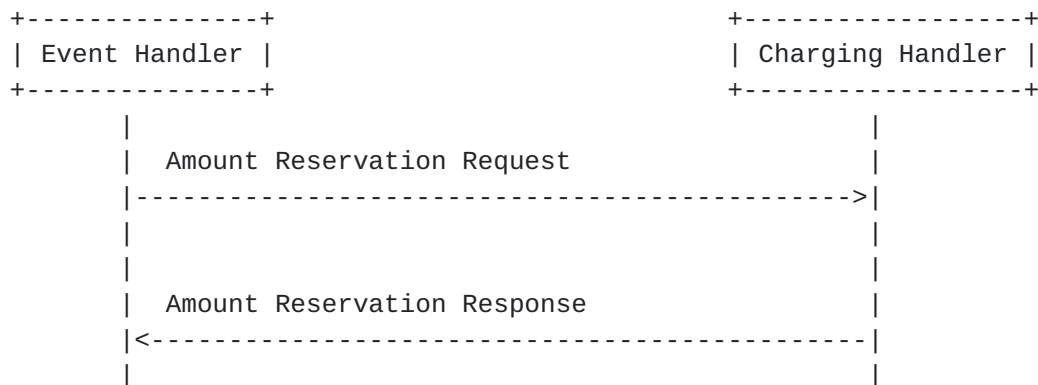


Figure 7: Amount Reservation Message Flow: Successful Case

Amount reservation cannot take place at the CH's site if there is not enough cover of the prepaid user account. This circumstance is indicated to the EH by means of a Reject message:

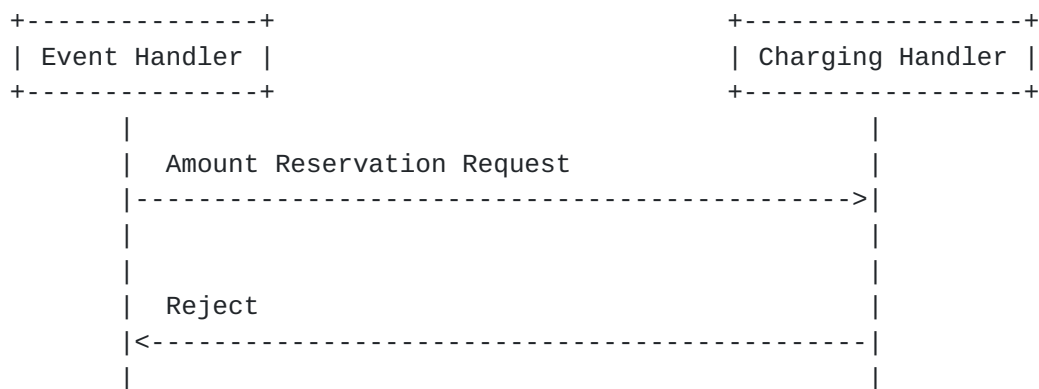


Figure 8: Amount Reservation Message Flow: Failure Case

4.4 Amount Capture

After having reserved a certain amount of a prepaid account, this amount can be captured. Capturing reserved amounts is initiated by an Amount Capture Request and - in case of success - confirmed by an Amount Capture Response:

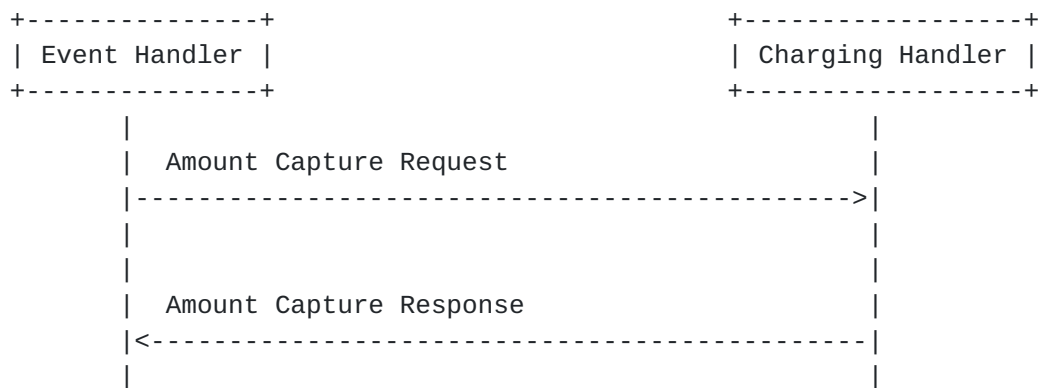


Figure 9: Amount Capture Message Flow: Successful Case

It might happen that the CH has to refuse the final amount capture for some reason, although the CH had sent a positive Amount Reservation Response to the EH before. In this case, the CH notifies this fact by means of a Reject message.



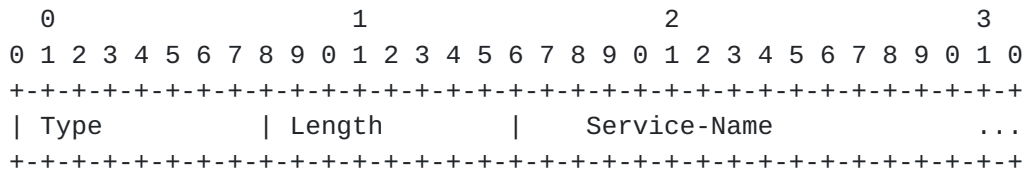
Figure 10: Amount Capture Message Flow: Failure Case

5. New Radius Attributes for Event-based Charging

This section defines a set of new Radius attributes that - along with attributes standardized at the IETF previously - constitute the Radius messages specified in [Section 6](#).

5.1 Service-Name

The Service-Name attribute specifies the service to which the user requests access.



Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

>= 3 Byte

Service-Name

The value field of the Service-Name attribute is of type "string". It identifies the service to which the user has requested access. Service names MUST be assigned in a way independent of a specific administration domain (to be detailed in future versions of this document).

5.2 Requested-Action

The Requested-Action attribute specifies what operation the entity receiving this attribute is requested to perform: price enquiry, amount reservation, amount capture, or price enquiry.

0										1					2				
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	0	1	2	3	4
Type										Length					Requested-A.				

Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

3 Byte

Requested-Action

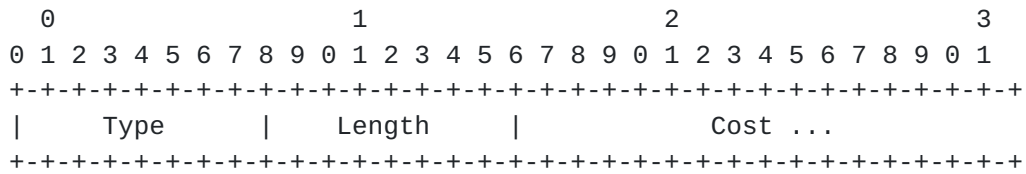
The value field of the Requested-Action attribute is of type "integer". The following integer values are supported:

- 1 price-enquiry
- 2 direct-debiting
- 3 reservation
- 4 capture

All other values are reserved.

5.3 Cost

The Cost attribute indicates price information. It contains the number (e.g., 70) of minor currency units (e.g., Cent) to be reserved or debited from the user's prepaid account. In cases where no minor currency unit is available the major currency unit must be taken.



Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

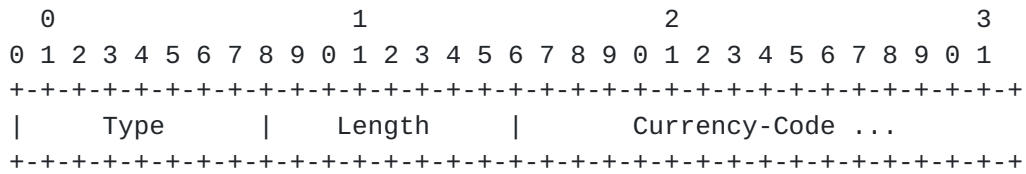
3 - 6 Byte

Cost

The value field of the Cost attribute is of type "integer" and indicates the number of minor (if possible; otherwise, major) currency units to be reserved or debited from the user's prepaid account.

5.4 Currency-Code

This attribute indicates the currency to be applied to the value of the Cost attribute.



Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

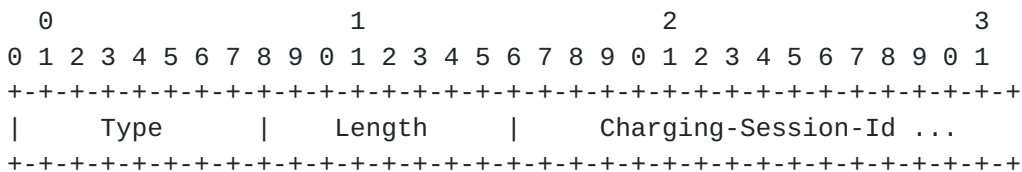
3 - 6 Byte

Currency-Code

The value field of the Currency-Code attribute is of type "string" and indicates the currency to be applied to the Cost value as indicated by the Cost attribute. The string value for a single currency is defined in [\[ISO4217\]](#).

5.5 Charging-Session-Id

This attribute is a unique identifier the EH assigns to an event. It is to facilitate the matching between different but correlated messages such as Amount Reservation Requests and Amount Capture Requests. In case of an Price Enquiry Request message, it is possible that the Charging-Session-Id identifier is not generated by the EH, but by the CH or the SP, depending on which of these entities sends the Price Enquiry Request.



Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

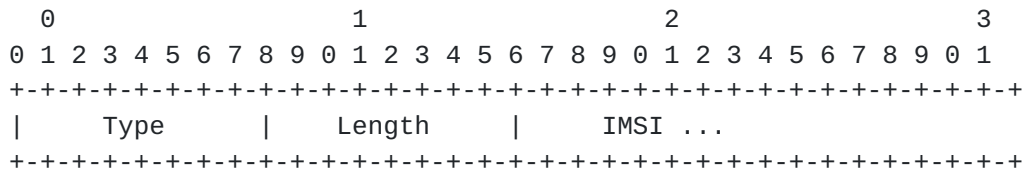
>= 3 Byte

Charging-Session-Id

The value field of the Charging-Session-Id attribute is of type "string".

5.6 International Mobile Subscriber Identity (IMSI)

If appropriate, this attribute can be used to identify a mobile subscriber. Thus, it fits in the series of standard Radius attributes such as Calling-Station-Id and Framed-IP-Address that are suitable in the scope of this document for request originator identification (especially at the Charging Handler's site which has to map debiting and reservation requests to the right user accounts). The definition of this attribute is borrowed from 3GPP where this attribute is called 3GPP-IMSI (see [[3GPP.29.061](#)]).



Type

To be assigned by IANA, or to become a sub-type of a new IANA assigned attribute type for event-based charging or charging in general.

Length

<= 17 Byte

IMSI

The value field of the IMSI attribute is of type "text". It contains an UTF-8 encoded IMSI. An IMSI consists of not more than 15 digits each of which requires one Byte in the value field of this attribute. The structure and content of IMSIs are defined in [[3GPP.23.003](#)].

6. Radius Messages for Event-based Charging

This section explicitly introduces Radius messages for event-based charging and specifies which Radius attributes are mandatory or optional. See [Section 4](#) for an informal description of these messages.

Regarding the number of occurrences of attributes in the Radius messages defined below, we use the following abbreviations:

- 0+ Zero or more instances of this attribute MAY be present.
- 0-1 Zero or one instance of this attribute MAY be present.
- 1 Exactly one instance of this attribute MUST be present.

6.1 Price Enquiry Request

Packet Type:

Access-Request (Code = 1)

Attributes:

Framed-IP-Address: 0-1 (see [[RFC2865](#)])
Calling-Station-Id: 0-1 (see [[RFC2865](#)])
IMSI: 0-1
Requested-Action: 1 (value MUST be set to 1 = price-enquiry)
Service-Name: 1
Charging-Session-Id: 1

Note 1:

None of the first three attributes must occur within this message. However, rating an event might be dependent on user identity in some scenarios (there might be, for instance, different user categories each of which has its own rules for rating).

Note 2:

According to [[RFC2865](#)], Radius Access-Requests MUST contain either a User-Password or a CHAP-Password or State attribute. An Access-Request MUST NOT contain both a User-Password and a CHAP-Password. Furthermore, an Access-Request MUST contain either a NAS-IP-Address or a NAS-Identifier (or both). Which of these attributes are transferred in a Price Enquiry Request message in addition to the ones listed above depends on the usage scenario.

6.2 Price Enquiry Response

Packet Type:
Access-Accept (Code = 2)

Attributes:
Charging-Session-Id: 1
Cost: 1
Currency-Code: 0-1 (currency might be clear from context)

Note:
The RE MUST use the same Charging-Session-Id value as presented in the corresponding Price Enquiry Request message.

6.3 Direct Debiting Request

Packet Type:
Access-Request (Code = 1)

Attributes:
Framed-IP-Address: 0-1 (see [[RFC2865](#)])
Calling-Station-Id: 0-1 (see [[RFC2865](#)])
IMSI: 0-1
Requested-Action: 1 (value MUST be set to 2 = direct-debiting)
Service-Name: 1
Charging-Session-Id: 1
Cost: 1
Currency-Code: 0-1 (currency might be clear from context)

Note 1:
At least one of the first three attributes MUST occur within this message in order to enable the CH to map this request to the right user account.

Note 2:
According to [[RFC2865](#)], Radius Access-Requests MUST contain either a User-Password or a CHAP-Password or State attribute. An Access-Request MUST NOT contain both a User-Password and a CHAP-Password. Furthermore, an Access-Request MUST contain either a NAS-IP-Address or a NAS-Identifier (or both). Which of these attributes are transferred in a Direct Debiting Request message in addition to the ones listed above depends on the usage scenario.

6.4 Direct Debiting Response

Packet Type:
Access-Accept (Code = 2)

Attributes:
Charging-Session-Id: 1

The CH MUST use the same Charging-Session-Id value as presented in the corresponding Direct Debiting Request message.

6.5 Amount Reservation Request

Packet Type:
Access-Request (Code = 1)

Attributes:
Framed-IP-Address: 0-1 (see [[RFC2865](#)])
Calling-Station-Id: 0-1 (see [[RFC2865](#)])
IMSI: 0-1
Requested-Action: 1 (value MUST be set to 3 = reservation)
Service-Name: 1
Charging-Session-Id: 1
Cost: 1
Currency-Code: 0-1 (currency might be clear from context)

At least one of the first three attributes MUST occur within this message in order to enable the CH to map this request to the right user account.

According to [[RFC2865](#)], Radius Access-Requests MUST contain either a User-Password or a CHAP-Password or State attribute. An Access-Request MUST NOT contain both a User-Password and a CHAP-Password. Furthermore, an Access-Request MUST contain either a NAS-IP-Address or a NAS-Identifier (or both). Which of these attributes are transferred in an Amount Reservation Request message in addition to the ones listed above depends on the usage scenario.

6.6 Amount Reservation Response

Packet Type:
Access-Accept (Code = 2)

Attributes:
Charging-Session-Id: 1

The CH MUST use the same Charging-Session-Id value as presented in the corresponding Amount Reservation Request message.

6.7 Amount Capture Request

Packet Type:

Access-Request (Code = 1)

Attributes:

Framed-IP-Address: 0-1 (see [[RFC2865](#)])
Calling-Station-Id: 0-1 (see [[RFC2865](#)])
IMSI: 0-1
Requested-Action: 1 (value MUST be set to 4 = capture)
Service-Name: 1
Charging-Session-Id: 1

According to [[RFC2865](#)], Radius Access-Requests MUST contain either a User-Password or a CHAP-Password or State attribute. An Access-Request MUST NOT contain both a User-Password and a CHAP-Password. Furthermore, an Access-Request MUST contain either a NAS-IP-Address or a NAS-Identifier (or both). Which of these attributes are transferred in an Amount Capture Request message in addition to the ones listed above depends on the usage scenario.

6.8 Amount Capture Response

Packet Type:

Access-Accept (Code = 2)

Attributes:

Charging-Session-Id: 1

The CH MUST use the same Charging-Session-Id value as presented in the corresponding Amount Capture Request message.

6.9 Reject

Packet Type:

Access-Reject (Code = 3)

Attributes:

Charging-Session-Id: 1
Reply-Message: 0+ (see [[RFC2865](#)])

Reply-Message attributes are used here to transport textual indications to the receiver of this message why the corresponding request could not be processed successfully. Within the scope of this document, the following character strings are apparently helpful

for describing reasons of request refusal:

requested-action-not-supported

missing-parameter (e.g., name of service is missing from request)

invalid-parameter (e.g., service name "www.supercom.com/
superservice" is invalid)

unknown-subscriber

limits-violated (e.g., due to insufficient account cover)

unspecified (no explicit reason provided)

[RFC2865] recommends to UTF-8 encode the characters of these strings.

7. IANA Considerations

This document requires the assignment of new Radius attributes number for the following attributes:

- Service-Name
- Requested-Action
- Cost
- Currency-Code
- Charging-Session-Id
- IMSI

8. Security Considerations

TBD

9. References

9.1 Normative References

- [ISO4217] ISO, "Codes for the representation of currencies and funds", August 2001.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.
- [RFC2865] Rigney, C., Willens, S., Rubens, A. and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", [RFC 2865](#), June 2000.
- [RFC2866] Rigney, C., "RADIUS Accounting", [RFC 2866](#), June 2000, <reference.RFC.2866.xml>.

9.2 Informative References

- [3GPP.23.003]
3GPP, "Numbering, addressing and identification; Release 6", 3GPP TS 23.003, June 2004.
- [3GPP.29.061]
3GPP, "Interworking between the Public Land Mobile Network (PLMN) supporting packet based services and Packet Data Networks (PDN); Release 6", 3GPP TS 29.061, June 2004.
- [I-D.[draft-lior-radius-prepaid-extensions-03](#)]
Lior, A., Yegani, P., Chowdhury, K., Madour, L. and Y. Li, "PrePaid Extensions to Remote Authentication Dial-In User Service (RADIUS)", [draft-lior-radius-prepaid-extensions-03](#) (work in progress), February 2004, <ref.I-D.[draft-lior-radius-prepaid-extensions-03.txt](#)>.

Authors' Addresses

Christian Guenther
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

E-Mail: christian.guenther@siemens.com

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany

EMail: hannes.tschofenig@siemens.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM 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.

Copyright Statement

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

