RADIUS Working Group C Rigney
INTERNET-DRAFT Livingston
expires in six months June 1996

RADIUS Accounting
draft-ietf-radius-accounting-04.txt

Status of this Memo

This document is a submission to the RADIUS Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the ietf-radius@livingston.com mailing list.

Distribution of this memo is unlimited.

This document is an Internet-Draft. 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.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This document describes a protocol for carrying accounting information between a Network Access Server and a shared Accounting Server.

Table of Contents

1. Introduction 1
1.1 Specification of Requirements 2
1.2 Terminology 2

2. Operation 3

3. Packet Format 4

4. Packet Types 6
4.1 Accounting-Request 6
4.2 Accounting-Response 7

5. Attributes 9
5.1 Acct-Status-Type 10
5.2 Acct-Delay-Time 11
5.3 Acct-Input-Octets 12
5.4 Acct-Output-Octets 13
5.5 Acct-Session-Id 13
5.6 Acct-Authentic 14
5.7 Acct-Session-Time 15
5.8 Acct-Input-Packets 16
5.9 Acct-Output-Packets 17
5.10 Acct-Terminate-Cause 17
5.11 Acct-Multi-Session-Id 20
5.12 Acct-Link-Count 20
5.13 Table of Attributes 22

Security Considerations 24

References 24

Acknowledgements 24

Chair's Address 25

Author's Address 25

1. Introduction

Managing dispersed serial line and modem pools for large numbers of users can create the need for significant administrative support. Since modem pools are by definition a link to the outside world, they require careful attention to security, authorization and accounting. This can be best achieved by managing a single "database" of users, which allows for authentication (verifying user name and password) as well as configuration information detailing the type of service to deliver to the user (for example, SLIP, PPP, telnet, rlogin).

The RADIUS (Remote Authentication Dial In User Service) Internet-Draft specifies the RADIUS protocol used for Authentication and Authorization. This Internet-Draft extends the use of the RADIUS protocol to cover delivery of accounting information from the Network Access Server (NAS) to a RADIUS accounting server.

Key features of RADIUS Accounting are:

Client/Server Model

A Network Access Server (NAS) operates as a client of the RADIUS accounting server. The client is responsible for passing user accounting information to a designated RADIUS accounting server.

The RADIUS accounting server is responsible for receiving the accounting request and returning a response to the client indicating that it has successfully received the request.

The RADIUS accounting server can act as a proxy client to other kinds of accounting servers.

Network Security

Transactions between the client and RADIUS accounting server are authenticated through the use of a shared secret, which is never sent over the network.

Extensible Protocol

All transactions are comprised of variable length Attribute-Length-Value 3-tuples. New attribute values can be added without disturbing existing implementations of the protocol.

1.1. Specification of Requirements

In this document, several words are used to signify the requirements of the specification. These words are often capitalized.

MUST This word, or the adjective "required", means that the definition is an absolute requirement of the specification.

MUST NOT This phrase means that the definition is an absolute prohibition of the specification.

SHOULD This word, or the adjective "recommended", means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications must be understood and carefully weighed before choosing a different course.

MAY This word, or the adjective "optional", means that this item is one of an allowed set of alternatives. An implementation which does not include this option MUST be prepared to interoperate with another implementation which does include the option.

1.2. Terminology

This document uses the following terms:

service The NAS provides a service to the dial-in user, such as PPP or Telnet.

session Each service provided by the NAS to a dial-in user constitutes a session, with the beginning of the session defined as the point where service is first provided and the end of the session defined as the point where service is ended. A user may have multiple sessions in parallel or series if the NAS supports that, with each session generating a separate start and stop accounting record with its own Acct-Session-Id.

silently discard
This means the implementation discards the packet without further processing. The implementation SHOULD provide the capability of logging the error, including the contents of the silently discarded packet, and SHOULD record the event in a statistics counter.

2. Operation

When a client is configured to use RADIUS Accounting, at the start of service delivery it will generate an Accounting Start packet describing the type of service being delivered and the user it is being delivered to, and will send that to the RADIUS Accounting server, which will send back an acknowledgement that the packet has been received. At the end of service delivery the client will generate an Accounting Stop packet describing the type of service that was delivered and optionally statistics such as elapsed time, input and output octets, or input and output packets. It will send that to the RADIUS Accounting server, which will send back an acknowledgement that the packet has been received.

The Accounting-Request (whether for Start or Stop) is submitted to the RADIUS accounting server via the network. It is recommended that the client continue attempting to send the Accounting-Request packet until it receives an acknowledgement, using some form of backoff. If no response is returned within a length of time, the request is re-sent a number of times. The client can also forward requests to an alternate server or servers in the event that the primary server is down or unreachable. An alternate server can be used either after a number of tries to the primary server fail, or in a round-robin fashion. Retry and fallback algorithms are the topic of current research and are not specified in detail in this document.

The RADIUS accounting server MAY make requests of other servers in order to satisfy the request, in which case it acts as a client.

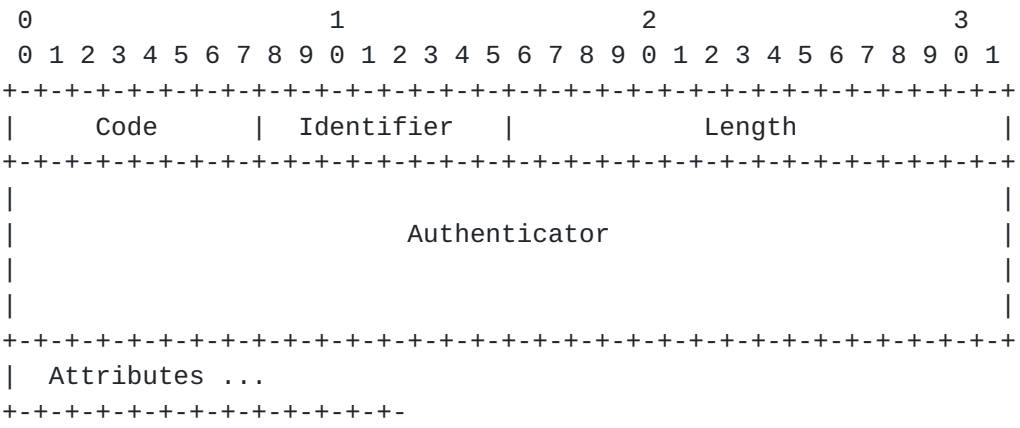
If the RADIUS accounting server is unable to successfully record the accounting packet it MUST NOT send an Accounting-Response acknowledgment to the client.

3. Packet Format

Exactly one RADIUS Accounting packet is encapsulated in the UDP Data field [1], where the UDP Destination Port field indicates 1646 (decimal).

When a reply is generated, the source and destination ports are reversed.

A summary of the RADIUS data format is shown below. The fields are transmitted from left to right.



Code

The Code field is one octet, and identifies the type of RADIUS packet. When a packet is received with an invalid Code field, it is silently discarded.

RADIUS Accounting Codes (decimal) are assigned as follows:

4	Accounting-Request
5	Accounting-Response

Identifier

The Identifier field is one octet, and aids in matching requests and replies.

Length

The Length field is two octets. It indicates the length of the packet including the Code, Identifier, Length, Authenticator and Attribute fields. Octets outside the range of the Length field

should be treated as padding and should be ignored on reception. If the packet is shorter than the Length field indicates, it should be silently discarded. The minimum length is 20 and maximum length is 4096.

Authenticator

The Authenticator field is sixteen (16) octets. The most significant octet is transmitted first. This value is used to authenticate the messages between the client and RADIUS accounting server.

Request Authenticator

In Accounting-Request Packets, the Authenticator value is a 16 octet MD5 [3] checksum, called the Request Authenticator.

The NAS and RADIUS accounting server share a secret. The Request Authenticator field in Accounting-Request packets contains a one-way MD5 hash calculated over a stream of octets consisting of the Code + Identifier + Length + 16 zero octets + request attributes + shared secret (where + indicates concatenation). The 16 octet MD5 hash value is stored in the Authenticator field of the Accounting-Request packet.

Note that the Request Authenticator of an Accounting-Request can not be done the same way as the Request Authenticator of a RADIUS Access-Request, because there is no User-Password attribute in an Accounting-Request.

Response Authenticator

The Authenticator field in an Accounting-Response packet is called the Response Authenticator, and contains a one-way MD5 hash calculated over a stream of octets consisting of the Accounting-Response Code, Identifier, Length, the Request Authenticator field from the Accounting-Request packet being replied to, and the response attributes if any, followed by the shared secret. The resulting 16 octet MD5 hash value is stored in the Authenticator field of the Accounting-Response packet.

Attributes

Attributes may have multiple instances, in such a case the order of attributes of the same type SHOULD be preserved. The order of attributes of different types is not required to be preserved.

4. Packet Types

The RADIUS packet type is determined by the Code field in the first octet of the packet.

4.1. Accounting-Request

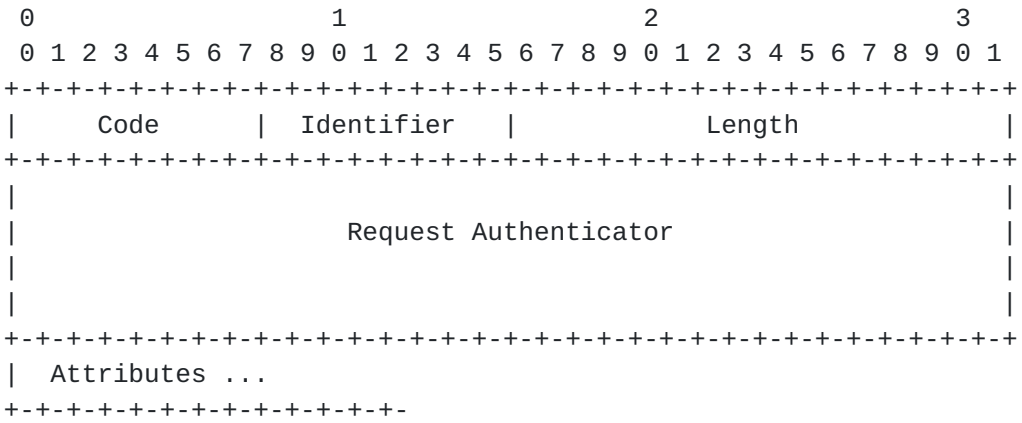
Description

Accounting-Request packets are sent from a client (typically a Network Access Server or its proxy) to a RADIUS accounting server, and convey information used to provide accounting for a service provided to a user. The client transmits a RADIUS packet with the Code field set to 4 (Accounting-Request).

Upon receipt of an Accounting-Request, the server MUST transmit an Accounting-Response reply if it successfully records the accounting packet, and MUST NOT transmit any reply if it fails to record the accounting packet.

Any attribute valid in a RADIUS Access-Request or Access-Accept packet is valid in a RADIUS Accounting-Request packet, except that the following attributes MUST NOT be present in an Accounting-Request: User-Password, CHAP-Password, Reply-Message, State. Either NAS-IP-Address or NAS-Identifier MUST be present in a RADIUS Accounting-Request. It SHOULD contain a NAS-Port or NAS-Port-Type attribute or both unless the service does not involve a port or the NAS does not distinguish among its ports.

A summary of the Accounting-Request packet format is shown below. The fields are transmitted from left to right.



Code

4 for Accounting-Request.

Identifier

The Identifier field MUST be changed whenever the content of the Attributes field changes, and whenever a valid reply has been received for a previous request. For retransmissions where the contents are identical, the Identifier MUST remain unchanged.

Note that if Acct-Delay-Time is included in the attributes of an Accounting-Request then the Acct-Delay-Time value will be updated when the packet is retransmitted, changing the content of the Attributes field and requiring a new Identifier and Request Authenticator.

Request Authenticator

The Request Authenticator of an Accounting-Request contains a 16-octet MD5 hash value calculated according to the method described in "Request Authenticator" above.

Attributes

The Attributes field is variable in length, and contains a list of Attributes.

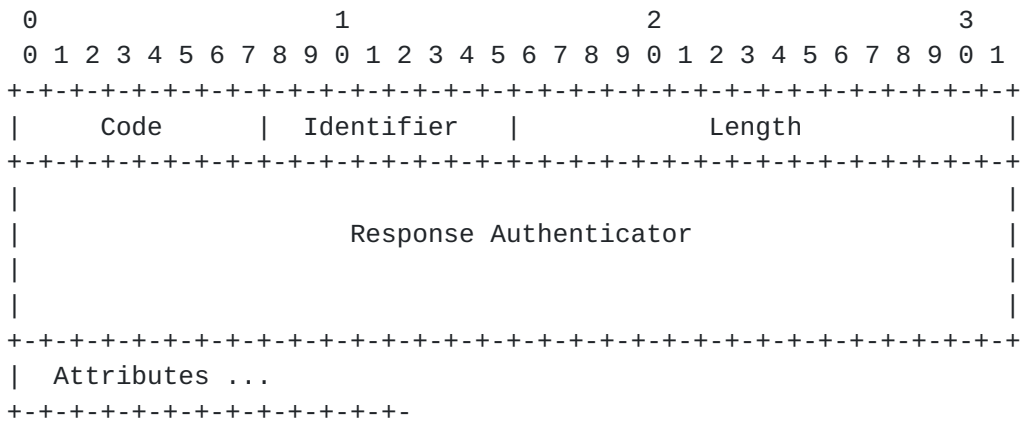
4.2. Accounting-Response

Description

Accounting-Response packets are sent by the RADIUS accounting server to the client to acknowledge that the Accounting-Request has been received and recorded successfully. If the Accounting-Request was recorded successfully then the RADIUS accounting server MUST transmit a packet with the Code field set to 5 (Accounting-Response). On reception of an Accounting-Response by the client, the Identifier field is matched with a pending Accounting-Request. Invalid packets are silently discarded.

A RADIUS Accounting-Response is not required to have any attributes in it.

A summary of the Accounting-Response packet format is shown below. The fields are transmitted from left to right.



Code

5 for Accounting-Response.

Identifier

The Identifier field is a copy of the Identifier field of the Accounting-Request which caused this Accounting-Response.

Response Authenticator

The Response Authenticator of an Accounting-Response contains a 16-octet MD5 hash value calculated according to the method described in "Response Authenticator" above.

Attributes

The Attributes field is variable in length, and contains a list of zero or more Attributes.

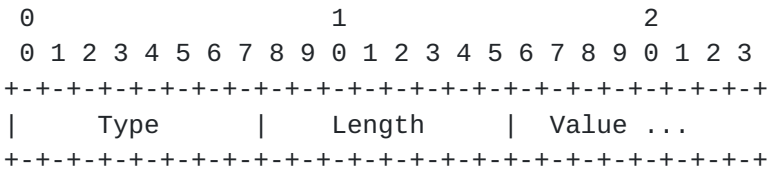
5. Attributes

RADIUS Attributes carry the specific authentication, authorization and accounting details for the request and response.

Some attributes MAY be included more than once. The effect of this is attribute specific, and is specified in each attribute description.

The end of the list of attributes is indicated by the Length of the RADIUS packet.

A summary of the attribute format is shown below. The fields are transmitted from left to right.



Type

The Type field is one octet. Up-to-date values of the RADIUS Type field are specified in the most recent "Assigned Numbers" RFC [2]. Values 192-223 are reserved for experimental use, values 224-240 are reserved for implementation-specific use, and values 241-255 are reserved and should not be used. This specification concerns the following values:

1-39	(refer to RADIUS Internet-Draft)
40	Acct-Status-Type
41	Acct-Delay-Time
42	Acct-Input-Octets
43	Acct-Output-Octets
44	Acct-Session-Id
45	Acct-Authentic
46	Acct-Session-Time
47	Acct-Input-Packets
48	Acct-Output-Packets
49	Acct-Terminate-Cause
50	Acct-Multi-Session-Id
51	Acct-Link-Count
60+	(refer to RADIUS Internet-Draft)

Length

The Length field is one octet, and indicates the length of this attribute including the Type, Length and Value fields. If an attribute is received in an Accounting-Request with an invalid Length, the entire request should be silently discarded.

Value

The Value field is zero or more octets and contains information specific to the attribute. The format and length of the Value field is determined by the Type and Length fields.

The format of the value field is one of four data types.

string	0-253 octets
address	32 bit value, most significant octet first.
integer	32 bit value, most significant octet first.
time	32 bit value, most significant octet first -- seconds since 00:00:00 GMT, January 1, 1970. The standard Attributes do not use this data type but it is presented here for possible use within Vendor-Specific attributes.

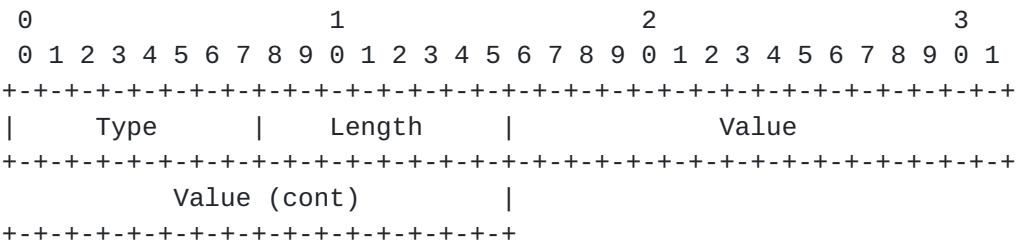
5.1. Acct-Status-Type

Description

This attribute indicates whether this Accounting-Request marks the beginning of the user service (Start) or the end (Stop).

It MAY be used by the client to mark the start of accounting (for example, upon booting) by specifying Accounting-On and to mark the end of accounting (for example, just before a scheduled reboot) by specifying Accounting-Off.

A summary of the Acct-Status-Type attribute format is shown below. The fields are transmitted from left to right.



Type

40 for Acct-Status-Type.

Length

6

Value

The Value field is four octets.

1	Start
2	Stop
7	Accounting-On
8	Accounting-Off

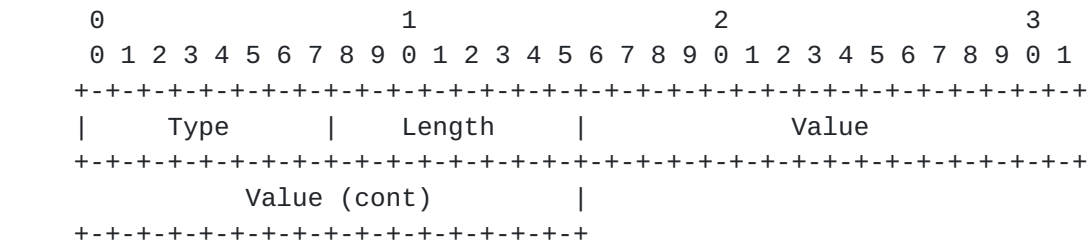
5.2. Acct-Delay-Time

Description

This attribute indicates how many seconds the client has been trying to send this record for, and can be subtracted from the time of arrival on the server to find the approximate time of the event generating this Accounting-Request. (Network transit time is ignored.)

Note that changing the Acct-Delay-Time causes the Identifier to change; see the discussion under Identifier above.

A summary of the Acct-Delay-Time attribute format is shown below. The fields are transmitted from left to right.



Type

41 for Acct-Delay-Time.

Length

6

Value

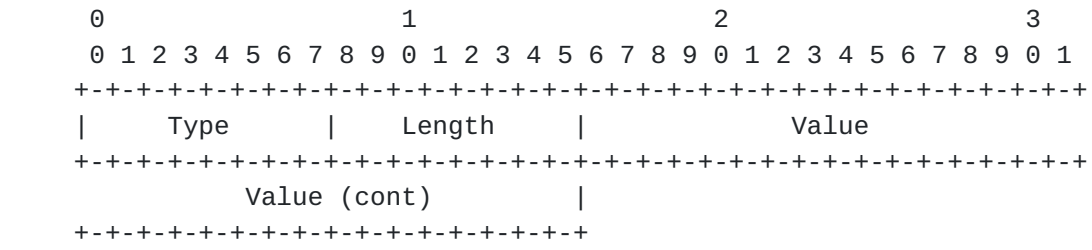
The Value field is four octets.

5.3. Acct-Input-Octets

Description

This attribute indicates how many octets have been received from the port over the course of this service being provided, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Input-Octets attribute format is shown below. The fields are transmitted from left to right.



Type

42 for Acct-Input-Octets.

Length

6

Value

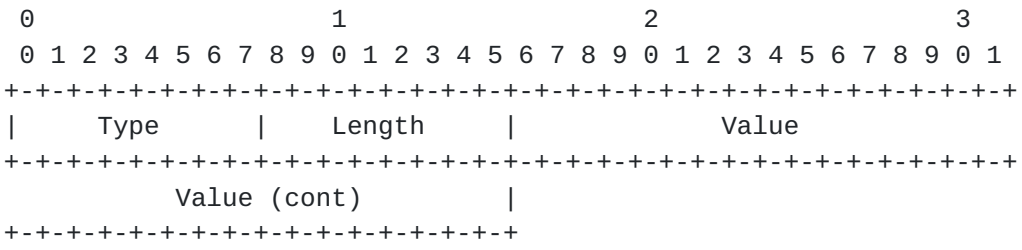
The Value field is four octets.

5.4. Acct-Output-Octets

Description

This attribute indicates how many octets have been sent to the port in the course of delivering this service, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Output-Octets attribute format is shown below. The fields are transmitted from left to right.



Type

43 for Acct-Output-Octets.

Length

6

Value

The Value field is four octets.

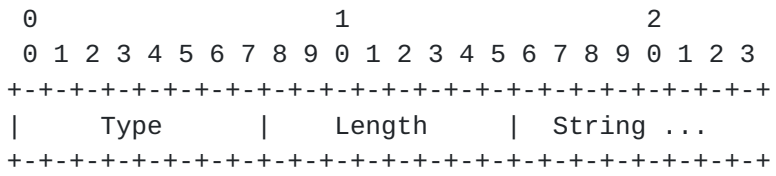
5.5. Acct-Session-Id

Description

This attribute is a unique Accounting ID to make it easy to match start and stop records in a log file. The start and stop records for a given session MUST have the same Acct-Session-Id. It is strongly recommended that the Acct-Session-Id be a printable ASCII string.

For example, one implementation uses a string with an 8-digit upper case hexadecimal number, the first two digits increment on each reboot (wrapping every 256 reboots) and the next 6 digits counting from 0 for the first person logging in after a reboot up to 2^24-1, about 16 million. Other encodings are possible.

A summary of the Acct-Session-Id attribute format is shown below. The fields are transmitted from left to right.



Type

44 for Acct-Session-Id.

Length

>= 3

String

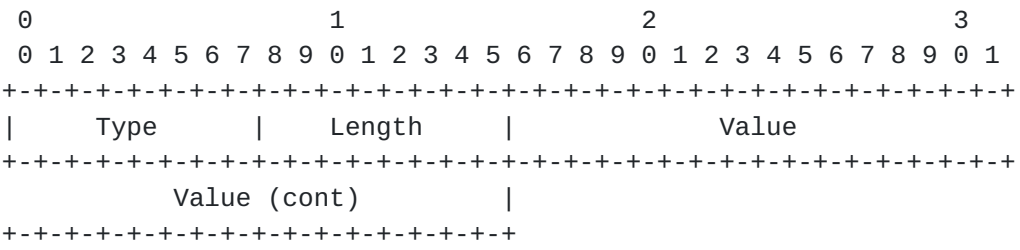
The String field SHOULD be a string of printable ASCII characters.

5.6. Acct-Authentic

Description

This attribute MAY be included in an Accounting-Request to indicate how the user was authenticated, whether by RADIUS, the NAS itself, or another remote authentication protocol. Users who are delivered service without being authenticated SHOULD NOT generate Accounting records.

A summary of the Acct-Authentic attribute format is shown below. The fields are transmitted from left to right.



Type

45 for Acct-Authentic.

Length

6

Value

The Value field is four octets.

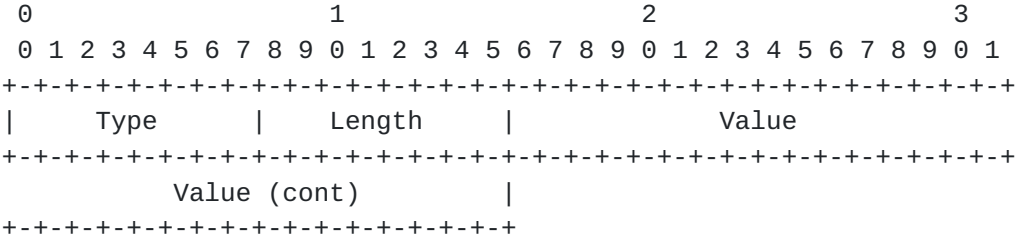
1 RADIUS
2 Local
3 Remote

5.7. Acct-Session-Time

Description

This attribute indicates how many seconds the user has received service for, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Session-Time attribute format is shown below. The fields are transmitted from left to right.



Type

46 for Acct-Session-Time.

Length

6

Value

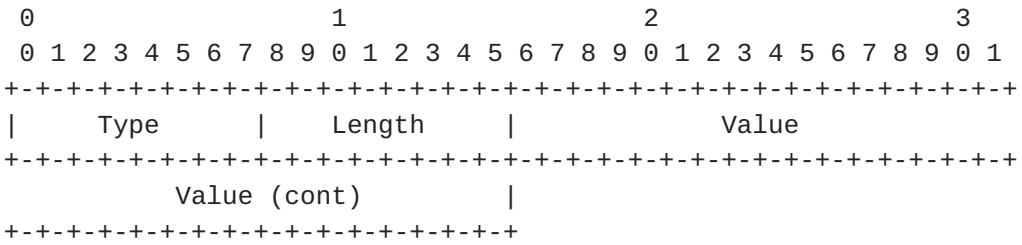
The Value field is four octets.

5.8. Acct-Input-Packets

Description

This attribute indicates how many packets have been received from the port over the course of this service being provided to a Framed User, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Input-packets attribute format is shown below. The fields are transmitted from left to right.



Type

47 for Acct-Input-Packets.

Length

6

Value

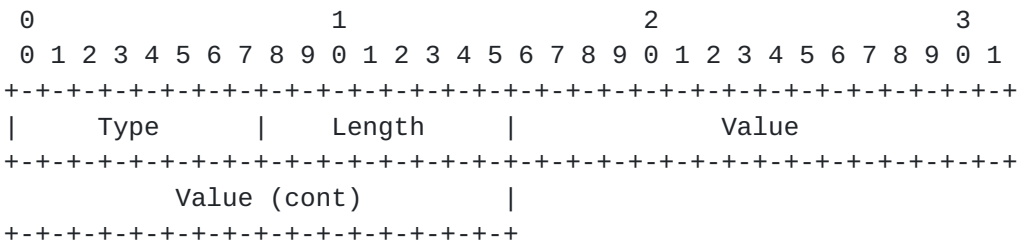
The Value field is four octets.

5.9. Acct-Output-Packets

Description

This attribute indicates how many packets have been sent to the port in the course of delivering this service to a Framed User, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Output-Packets attribute format is shown below. The fields are transmitted from left to right.



Type

48 for Acct-Output-Packets.

Length

6

Value

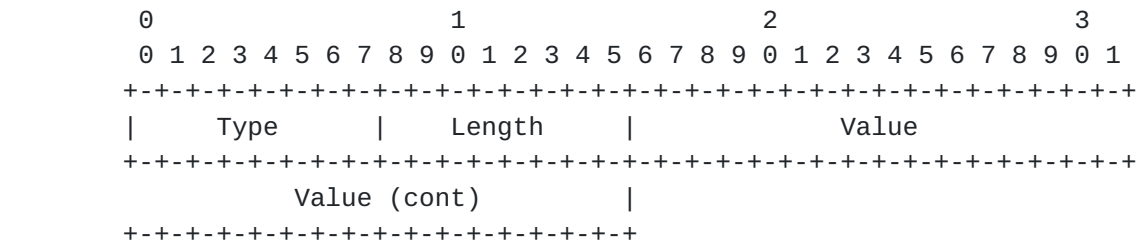
The Value field is four octets.

5.10. Acct-Terminate-Cause

Description

This attribute indicates how the session was terminated, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.

A summary of the Acct-Terminate-Cause attribute format is shown below. The fields are transmitted from left to right.



Type

49 for Acct-Terminate-Cause

Length

6

Value

The Value field is four octets, containing an integer specifying the cause of session termination, as follows:

1	User Request
2	Lost Carrier
3	Lost Service
4	Idle Timeout
5	Session Timeout
6	Admin Reset
7	Admin Reboot
8	Port Error
9	NAS Error
10	NAS Request
11	NAS Reboot
12	Port Unneeded
13	Port Preempted
14	Port Suspended
15	Service Unavailable
16	Callback
17	User Error
18	Host Request

The termination causes are as follows:

User Request	User requested termination of service, for example with LCP Terminate or by logging out.

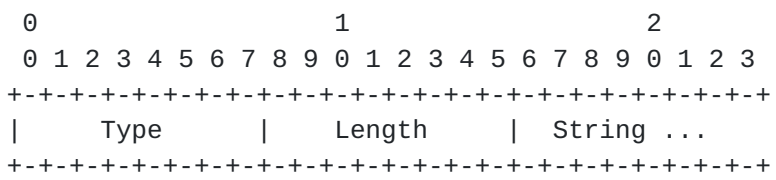
Lost Carrier	DCD was dropped on the port.
Lost Service	Service can no longer be provided; for example, user's connection to a host was interrupted.
Idle Timeout	Idle timer expired.
Session Timeout	Maximum session length timer expired.
Admin Reset	Administrator reset the port or session.
Admin Reboot	Administrator is ending service on the NAS, for example prior to rebooting the NAS.
Port Error	NAS detected an error on the port which required ending the session.
NAS Error	NAS detected some error (other than on the port) which required ending the session.
NAS Request	NAS ended session for a non-error reason not otherwise listed here.
NAS Reboot	The NAS ended the session in order to reboot non-administratively ("crash").
Port Unneeded	NAS ended session because resource usage fell below low-water mark (for example, if a bandwidth-on-demand algorithm decided that the port was no longer needed).
Port Preempted	NAS ended session in order to allocate the port to a higher priority use.
Port Suspended	NAS ended session to suspend a virtual session.
Service Unavailable	NAS was unable to provide requested service.
Callback	NAS is terminating current session in order to perform callback for a new session.
User Error	Input from user is in error, causing termination of session.
Host Request	Login Host terminated session normally.

5.11. Acct-Multi-Session-Id

Description

This attribute is a unique Accounting ID to make it easy to link together multiple related sessions in a log file. Each session linked together would have a unique Acct-Session-Id but the same Acct-Multi-Session-Id. It is strongly recommended that the Acct-Multi-Session-Id be a printable ASCII string.

A summary of the Acct-Session-Id attribute format is shown below. The fields are transmitted from left to right.



Type

50 for Acct-Multi-Session-Id.

Length

>= 3

String

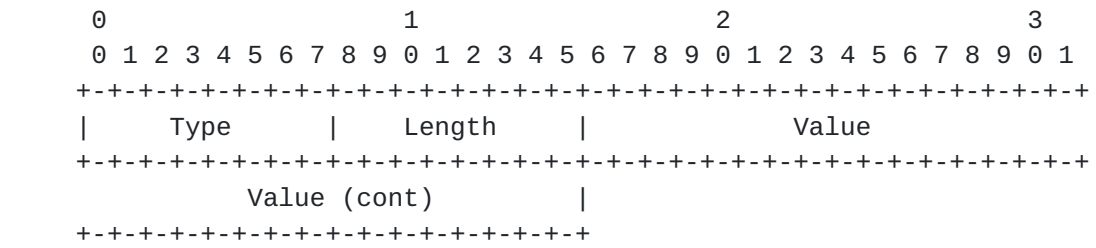
The String field SHOULD be a string of printable ASCII characters.

5.12. Acct-Link-Count

Description

This attribute gives the count of links which are known to have been in a given multilink session at the time the accounting record is generated. The NAS MAY include the Acct-Link-Count attribute in any Accounting-Request which might have multiple links.

A summary of the Acct-Link-Count attribute format is show below. The fields are transmitted from left to right.



Type

51 for Acct-Link-Count.

Length

6

Value

The Value field is four octets, and contains the number of links seen so far in this Multilink Session.

It may be used to make it easier for an accounting server to know when it has all the records for a given Multilink session. When the number of Accounting-Requests received with Acct-Status-Type = Stop and the same Acct-Multi-Session-Id and unique Acct-Session-Id's equals the largest value of Acct-Link-Count seen in those Accounting-Requests, all Stop Accounting-Requests for that Multilink Session have been received.

An example showing 8 Accounting-Requests should make things clearer. For clarity only the relevant attributes are shown, but additional attributes containing accounting information will also be present in the Accounting-Request.

Multi-Session-Id	Session-Id	Status-Type	Link-Count
"10"	"10"	Start	1
"10"	"11"	Start	2
"10"	"11"	Stop	2
"10"	"12"	Start	3
"10"	"13"	Start	4
"10"	"12"	Stop	4
"10"	"13"	Stop	4
"10"	"10"	Stop	4

5.13. Table of Attributes

The following table provides a guide to which attributes may be found in Accounting-Request packets. No attributes should be found in Accounting-Response packets (except possibly for Vendor-Specific).

#	Attribute
0-1	User-Name
0	User-Password
0	CHAP-Password
0-1	NAS-IP-Address [4]
0-1	NAS-Port
0-1	Service-Type
0-1	Framed-Protocol
0-1	Framed-IP-Address
0-1	Framed-IP-Netmask
0-1	Framed-Routing
0+	Filter-Id
0-1	Framed-MTU
0+	Framed-Compression
0+	Login-IP-Host
0-1	Login-Service
0-1	Login-TCP-Port
0	Reply-Message
0-1	Callback-Number
0-1	Callback-Id
0+	Framed-Route
0-1	Framed-IPX-Network
0	State
0+	Class
0+	Vendor-Specific
0-1	Session-Timeout
0-1	Idle-Timeout
0-1	Termination-Action
0-1	Called-Station-Id
0-1	Calling-Station-Id
0-1	NAS-Identifier [4]
0+	Proxy-State
0-1	Login-LAT-Service
0-1	Login-LAT-Node
0-1	Login-LAT-Group
0-1	Framed-AppleTalk-Link
0-1	Framed-AppleTalk-Network
0-1	Framed-AppleTalk-Zone
1	Acct-Status-Type
0-1	Acct-Delay-Time
0-1	Acct-Input-Octets

```
0-1   Acct-Output-Octets
1     Acct-Session-Id
0-1   Acct-Authentic
0-1   Acct-Session-Time
0-1   Acct-Input-Packets
0-1   Acct-Output-Packets
0-1   Acct-Terminate-Cause
0+    Acct-Multi-Session-Id
0+    Acct-Link-Count
0     CHAP-Challenge
0-1   NAS-Port-Type
0-1   Port-Limit
0-1   Login-LAT-Port
```

[4] An Accounting-Request MUST contain either a NAS-IP-Address or a NAS-Identifier, and it is permitted (but not recommended) for it to contain both.

The following table defines the above table entries.

```
0     This attribute MUST NOT be present
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.
```

Security Considerations

Security issues are briefly discussed in sections concerning the authenticator included in accounting requests and responses, using a shared secret which is never sent over the network.

References

[1] Postel, J., "User Datagram Protocol", STD 6, RFC 768, USC/Information Sciences Institute, August 1980.

[2] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700, USC/Information Sciences Institute, October 1994.

[3] Rivest, R., and S. Dusse, "The MD5 Message-Digest Algorithm", RFC 1321, MIT Laboratory for Computer Science, RSA Data Security Inc., April 1992.

Acknowledgments

RADIUS and RADIUS Accounting were originally developed by Livingston Enterprises for their PortMaster series of Network Access Servers.

Chair's Address

The RADIUS working group can be contacted via the current chair:

Carl Rigney
Livingston Enterprises
6920 Koll Center Parkway, Suite 220
Pleasanton, California 94566

Phone: +1 510 426 0770
E-Mail: cdr@livingston.com

Author's Address

Questions about this memo can also be directed to:

Carl Rigney
Livingston Enterprises
6920 Koll Center Parkway, Suite 220
Pleasanton, California 94566

E-Mail: cdr@livingston.com