AAA Working Group
Internet-Draft
Category: Standards Track
<draft-calhoun-diameter-nasreq-06.txt>

Pat R. Calhoun
Sun Microsystems, Inc.
William Bulley
Merit Network, Inc.
Allan C. Rubens
Tut Systems, Inc.
Jeff Haag
Glen Zorn
Cisco Systems, Inc.
February 2001

Diameter NASREQ Extensions

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

The list of Internet-Draft Shadow Directories can be accessed at:

http://www.ietf.org/shadow.html.

This document is an individual contribution for consideration by the AAA Working Group of the Internet Engineering Task Force. Comments should be submitted to the diameter@diameter.org mailing list.

Distribution of this memo is unlimited.

Abstract

This document describes the Diameter extension that is used for AAA in a PPP/SLIP Dial-Up and Terminal Server Access environment. This extension, combined with the base protocol, satisfies the requirements defined in the NASREQ AAA criteria specification and the ROAMOPS AAA Criteria specification.

Given that it is expected that initial deployments of the Diameter protocol in a dial-up environment will include legacy systems, this extension was carefully designed to ease the burden of servers that must perform protocol conversion between RADIUS and Diameter. This is achieved by re-using the RADIUS address space, eliminating the need to perform attribute lookups.

Table of Contents

1.0 Introduction

This document describes the Diameter extension that is used for AAA in a PPP/SLIP Dial-Up and Terminal Server Access environment. This extension, combined with the base protocol [2], satisfies the requirements defined in the NASREQ AAA criteria specification [24] and the ROAMOPS AAA Criteria specification [4].

This document is divided into three main sections. The first section defines the Diameter Command-Codes and AVPs that are needed to support legacy authentication protocols, those that are typically supported by RADIUS [1] servers. The second section defines the Command-Codes and AVPs necessary for a Diameter node to support PPP's Extensible Authentication Protocol (EAP) [25]. The third section contains the Authorization AVPs that are needed for the various services offered by a NAS, such as PPP dial-in, terminal server and tunneling applications, such as L2TP [16].

Given that it is expected that initial deployments of the Diameter protocol in a dial-up environment will include legacy systems, this extension was carefully designed to ease the burden of servers that must perform protocol conversion between RADIUS and Diameter. This is achieved by re-using the RADIUS address space, eliminating the need to perform attribute lookups.

The value assigned for the Extension-Id [2] AVP is one (1).

1.1 Requirements language

In this document, the key words "MAY", "MUST", "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [12].

2.0 Supported AVPs

This section lists all of the Diameter AVPs and the legacy RADIUS attributes supported by this extension.

2.1 Diameter AVPs

This section will define all of the AVPs that are not backward compatible with the RADIUS protocol [1]. A Diameter message that includes one of these AVPs MAY cause interoperability issues should the request traverse a AAA node that only supports the RADIUS protocol. However, the Diameter protocol SHOULD NOT be hampered from future developments due to the existing installed base.

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

				AVP Flag rules				
Attribute Name	AVP Code	Section Defined	Value Type	MUST	MAY	SHLD NOT	MUST NOT	MAY Encr
EAP-Payload	402	4.3	OctetString	M	P		V	Y
Filter-Rule	400	2.1.2	OctetString	M	P		V	Y
Request-Type	401	2.1.1	Unsigned32	M	P		V	N

2.1.1 Request-Type AVP

The Request-Type AVP (AVP Code 401) is of type Unsigned32 and is used to determine the type of request being transmitted. Note that a request with this AVP set to a value other than AUTHORIZE_AUTHENTICATE MAY break backward RADIUS compatibility. The following values are defined:

AUTHENTICATE_ONLY 1
The request being sent is for authentication only, and MUST contain the relevant authentication AVPs that are needed by the

Diameter server to authenticate the user.

AUTHORIZE_ONLY 2
The request being sent is for authorization only, and MUST contain the authorization AVPs that are necessary to identify the service being requested/offered.

AUTHORIZE_AUTHENTICATE 3
The request contains a request for both authentication and authorization. The request MUST include both the relevant authentication information, and authorization information necessary to identify the service being requested/offered.

2.1.2 Filter-Rule AVP

The Filter-Rule AVP (AVP Code 400) is of type OctetString and provides filter rules that need to be configured on the NAS for the user. One or more such AVPs MAY be present in an authorization response.

Each packet can be filtered based on the following information that is associated with it:

```
Direction                           (in or out)
Source and destination IP address   (possibly masked)
Protocol
Source and destination port         (lists or ranges)
TCP flags
IP fragment flag
IP options
ICMP types
```

Rules for the appropriate direction are evaluated in order, with the first matched rule terminating the evaluation. Each packet is evaluated once. If no rule matches, the packet is dropped if the last rule evaluated was a permit, and passed if the last rule was a deny.

The filters in the Filter-Rule AVP MUST follow the format:

```
action dir proto from src to dst [options]
```

action: permit - Allow packets that match the rule.
deny - Drop packets that match the rule.

dir: "in" is from the terminal, "out" is to the terminal.

proto: An IP protocol specified by number. The "ip" keyword

```
                   means any protocol will match.

      src and dst  <address/mask> [ports]

                   The <address/mask> may be specified as:
                   ipno       An IPv4 or IPv6 number in dotted-quad or
                              canonical IPv6 form. Only this exact IP
                              number will match the rule.
                   ipno/bits  An IP number as above with a mask width of
                              the form 1.2.3.4/24.  In this case all IP
                              numbers from 1.2.3.0 to 1.2.3.255 will
                              match.  The bit width MUST be valid for
                              the IP version and the IP number MUST NOT
                              have bits set beyond the mask.

                   The sense of the match can be inverted by preceding
                   an address with the not modifier, causing all other
                   addresses to be matched instead.  This does not
                   affect the selection of port numbers.

                      The keyword "any" is 0.0.0.0/0 or the IPv6
                      equivalent.  The keyword "assigned" is the address
                      or set of addresses assigned to the terminal.  The
                      first rule SHOULD be "deny in ip !assigned".

                   With the TCP and UDP protocols, optional ports may be
                   specified as:

                      {port|port-port}[,port[,...]]

                   The `-' notation specifies a range of ports
                   (including boundaries).

                   Fragmented packets which have a non-zero offset (i.e.
                   not the first fragment) will never match a rule which
                   has one or more port specifications.  See the frag
                   option for details on matching fragmented packets.

      options:
         frag    Match if the packet is a fragment and this is not the
                 first fragment of the datagram.  frag may not be used
                 in conjunction with either tcpflags or TCP/UDP port
                 specifications.

         ipoptions spec
                 Match if the IP header contains the comma separated
                 list of options specified in spec. The supported IP
                 options are:
```

ssrr (strict source route), lsrr (loose source route), rr (record packet route) and ts (timestamp). The absence of a particular option may be denoted with a `!'.

tcpoptions spec
Match if the TCP header contains the comma separated list of options specified in spec. The supported TCP options are:

mss (maximum segment size), window (tcp window advertisement), sack (selective ack), ts (rfc1323 timestamp) and cc (rfc1644 t/tcp connection count). The absence of a particular option may be denoted with a `!'.

established
TCP packets only. Match packets that have the RST or ACK bits set.

setup
TCP packets only. Match packets that have the SYN bit set but no ACK bit.

tcpflags spec
TCP packets only. Match if the TCP header contains the comma separated list of flags specified in spec. The supported TCP flags are:

fin, syn, rst, psh, ack and urg. The absence of a particular flag may be denoted with a `!'. A rule which contains a tcpflags specification can never match a fragmented packet which has a non-zero offset. See the frag option for details on matching fragmented packets.

icmptypes types
ICMP packets only. Match if the ICMP type is in the list types. The list may be specified as any combination of ranges or individual types separated by commas. The supported ICMP types are:

echo reply (0), destination unreachable (3), source quench (4), redirect (5), echo request (8), router advertisement (9), router solicitation (10), time-to-live exceeded (11), IP header bad (12), timestamp request (13), timestamp reply (14), information request (15), information reply (16), address mask request (17) and address mask reply (18).

There is one kind of packet that the NAS MUST always discard, that is an IP fragment with a fragment offset of one. This is a valid packet, but it only has one use, to try to circumvent firewalls.

> A NAS that is unable to interpret or apply a deny rule MUST terminate the session. A NAS that is unable to interpret or apply a permit rule MAY apply a more restrictive rule. A NAS MAY apply deny rules of its own before the supplied rules, for example to protect the NAS owner's infrastructure.

The rule syntax is a modified subset of ipfw(8) from FreeBSD, and the ipfw.c code may provide a useful base for implementations.

2.2 Legacy RADIUS Attributes

The Diameter protocol reserves the first 255 AVP identifiers for "legacy RADIUS" support. The following table contains the RADIUS attributes supported by this Diameter extension, their AVP code values, types, possible flag values and whether the AVP MAY be encrypted. RADIUS attributes not listed are not supported by the Diameter protocol.

				AVP Flag rules				
Attribute Name	AVP Code	Section Defined	Value Type	MUST	MAY	SHLD NOT	MUST NOT	MAY Encr
ARAP-Features	71	7.2.8.1	OctetString	M	P		V	Y
ARAP-Password	70	3.1.1.4	OctetString	M	P		V	Y
ARAP-Security	73	7.2.8.3	Unsigned32	M	P		V	Y
ARAP-Security-Data	74	7.2.8.4	OctetString	M	P		V	Y
ARAP-Zone-Access	72	7.2.8.2	Unsigned32	M	P		V	Y
Callback-Id	20	6.4	OctetString	M	P		V	Y
Callback-Number	19	6.3	OctetString	M	P		V	Y
Called-Station-Id	30	6.6	OctetString	M	P		V	Y
Calling-Station-Id	31	6.7	OctetString	M	P		V	Y
CHAP-Challenge	60	3.1.1.3	OctetString	M	P		V	Y
CHAP-Password	3	3.1.1.2	OctetString	M	P		V	Y
Class	25	2.2.4	OctetString	M	P		V	Y
Connect-Info	77	6.10	OctetString	M	P		V	Y
Filter-Id	11	6.2	OctetString	M	P		V	Y

				AVP Flag rules				
Attribute Name	AVP Code	Section Defined	Value Type	MUST	MAY	SHLD NOT	MUST NOT	MAY Encr
Framed-Appletalk-Link	37	7.2.7.1	Unsigned32	M	P		V	Y
Framed-Appletalk-Network	38	7.2.7.2	Unsigned32	M	P		V	Y
Framed-Appletalk-Zone	39	7.2.7.3	OctetString	M	P		V	Y
Framed-Protocol	7	7.2.1	Unsigned32	M	P		V	Y
Framed-IP-Address	8	7.2.5.1	Address	M	P		V	Y
Framed-Compression	13	7.2.4	Unsigned32	M	P		V	Y
Framed-IP-Netmask	9	7.2.5.2	Address	M	P		V	Y
Framed-IP-Route	22	7.2.5.3	OctetString	M	P		V	Y
Framed-IPX-Route	23	7.2.6.1	OctetString	M	P		V	Y
Framed-MTU	12	7.2.3	Unsigned32	M	P		V	Y
Framed-Routing	10	7.2.2	Unsigned32	M	P		V	Y
Idle-Timeout	28	6.5	Unsigned32	M	P		V	Y
Login-IP-Host	14	7.3.1	Address	M	P		V	Y
Login-LAT-Group	36	7.3.4.3	OctetString	M	P		V	Y
Login-LAT-Node	35	7.3.4.2	OctetString	M	P		V	Y
Login-LAT-Port	63	7.3.4.4	OctetString	M	P		V	Y
Login-LAT-Service	34	7.3.4.1	Unsigned32	M	P		V	Y
Login-Service	15	7.3.2	Unsigned32	M	P		V	Y
Login-TCP-Port	16	7.3.3.1	Unsigned32	M	P		V	Y
User-Password	2	3.1.1.1	OctetString	M	P		V	Y
NAS-Identifier	32	2.2.2	OctetString	M	P		V	Y
NAS-IP-Address	4	2.2.1	Address	M	P		V	Y
NAS-Port	5	6.1.1	Unsigned32	M	P		V	Y
NAS-Port-Type	61	6.8	Unsigned32	M	P		V	Y
Password-Retry	75	3.1.2.2	Unsigned32	M	P		V	Y
Port-Limit	62	6.9	Unsigned32	M	P		V	Y
Prompt	76	3.1.3.1	Unsigned32	M	P		V	Y
Reply-Message	18	3.2	OctetString	M	P		V	Y
Service-Type	6	7.1	Unsigned32	M	P		V	Y
State	24	2.2.3	OctetString	M	P		V	Y
Tunnel-Assignment-Id	82	7.4.7	OctetString	M	P		V	Y
Tunnel-Client-Auth-ID	90	7.4.9	OctetString	M	P		V	Y
Tunnel-Client-Endpoint	66	7.4.3	OctetString	M	P		V	Y

Attribute Name	AVP Code	Section Defined	Value Type	AVP Flag rules MUST	MAY	SHLD NOT	MUST NOT	MAY Encr
Tunnel-Medium-Type	65	7.4.2	Unsigned32	M	P		V	Y
Tunnel-Password	69	7.4.5	OctetString	M	P		V	Y
Tunnel-Preference	83	7.4.8	Unsigned32	M	P		V	Y
Tunnel-Private-Group-ID	81	7.4.6	OctetString	M	P		V	Y
Tunnel-Server-Auth-ID	91	7.4.10	OctetString	M	P		V	Y
Tunnel-Server-Endpoint	67	7.4.4	OctetString	M	P		V	Y
Tunnel-Type	64	7.4.1	Unsigned32	M	P		V	Y

The AVPs defined in this section SHOULD only used when a Diameter/RADIUS gateway function is invoked, and are not used in the Diameter protocol.

2.2.1 NAS-IP-Address AVP

The NAS-IP-Address AVP (AVP Code 4) [1] is of type Address, and contains the IP Address of the NAS providing service to the user. When this AVP is present, the Host-Name AVP DOES NOT represent the NAS providing service to the user. Note that this AVP SHOULD only added by a RADIUS/Diameter protocol gateway [28].

2.2.2 NAS-Identifier AVP

The NAS-Identifier AVP (AVP Code 32) [1] is of type OctetString, and contains the Identity of the NAS providing service to the user. When this AVP is present, the Host-Name AVP DOES NOT represent the NAS providing service to the user. Note that this AVP SHOULD only added by a RADIUS/Diameter protocol gateway [28].

2.2.3 State AVP

The State AVP (AVP Code 24) is of type OctetString and is used to transmit the contents of the RADIUS State attribute, and no interpretation of the contents should be made. Note that this AVP SHOULD only added by a RADIUS/Diameter protocol gateway [28].

2.2.4 Class AVP

The Class AVP (AVP Code 25) is of type OctetString and is used to transmit the contents of the RADIUS Class attribute, and no interpretation of the contents should be made. Note that this AVP SHOULD only added by a RADIUS/Diameter protocol gateway [28].

3.0 Legacy RADIUS Authentication Support

This section defines the new Command-Code [2] values required to support the legacy authentication protocols (i.e. PAP, CHAP), as well as the AVPs that are necessary to carry the authentication information in the Diameter protocol. The functionality defined here provides a RADIUS-like AAA service, over a more reliable and secure transport, as defined in the base protocol [2].

Unlike the RADIUS protocol [1], the Diameter protocol does not require authentication information to be contained in a request from the client. Therefore, it is possible to send a request for authorization only. The type of service depends upon the Request-Type AVP. This difference MAY cause operational issues in environments that need RADIUS interoperability, and it MAY be necessary that protocol conversion gateways add some authentication information when transmitting to a RADIUS server.

The Diameter protocol allows for users to be periodically re-authenticated and/or re-authorized. In such instances, the Session-Id AVP in the AAR message MUST be the same as the one present in the original authentication/authorization message. A Diameter server informs the NAS of the authorized session lifetime via the Session-Timeout AVP [1].

A NAS MUST re-authenticate and/or authorize after the period provided by the server. Furthermore, it is possible for Diameter servers to issue an unsolicited re-authentication and/or re-authorization by issuing an AA-Challenge-Ind message to the NAS. Upon receipt of such a message, the NAS is instructed to issue a request to re-authenticate and/or re-authorize the client.

3.1 Command-Codes Values

This section defines new Command-Code [2] values that MUST be supported by all Diameter implementations that conform to this specification. The following Command Codes are defined in this section:

Command-Name	Abbrev.	Code	Reference
AA-Answer	AAA	266	3.1.2
AA-Challenge-Ind	ACI	267	3.1.3
AA-Request	AAR	265	3.1.1

3.1.1 AA-Request (AAR) Command

The AA-Request message (AAR), indicated by the Command-Code field set to 265, is used in order to request authentication and/or authorization for a given PPP user. The type of request is identified through the Request-Type AVP, and the default mode is both authentication and authorization.

If Authentication is requested the User-Name attribute SHOULD be present, as well as any additional authentication AVPs that would carry the password information. A request for authorization only SHOULD include the information from which the authorization will be performed, such as the User-Name, or DNIS and ANI AVPs. Certain networks MAY use different AVPs for authorization purposes. A request for authorization will include some AVPs defined in sections 2.0, 6.0 and 7.0.

It is possible for a single session to be authorized only first, then followed by an authentication request. However, the inverse SHOULD NOT be permitted.

If the AA-Request is a result of an AA-Challenge-Ind, the Session-Id MUST be identical as the one provided in the initial AA-Request for the same session. If the AA-Request is a result of an AA-Challenge-Ind that included a State AVP, the same AVP MUST be present in the following AA-Request.

Message Format

```
<AA-Request> ::= < Diameter Header: 265 >
                 { Session-Id }
                 { Host-Name }
                 [ NAS-Identifier ]
                 [ User-Name ]
                 [ User-Password ]
                 [ ARAP-Password ]
                 [ CHAP-Password ]
                 [ CHAP-Challenge ]
                 [ State ]
               * [ AVP ]
               * [ Proxy-State ]
               * [ Route-Record ]
               * [ Routing-Realm ]
              0*1< Integrity-Check-Value >
```

3.1.1.1 User-Password AVP

The User-Password AVP (AVP Code 2) is of type OctetString and contains the password of the user to be authenticated, or the user's input following an AA-Challenge-Ind.

This AVP MUST be encrypted using one of the methods described in [2] or [13]. Unless this AVP is used for one-time passwords, the User-Password AVP SHOULD NOT be used in non-trusted proxy environments.

The clear-text password (prior to encryption) MUST NOT be longer than 128 bytes in length.

3.1.1.2 CHAP-Password AVP

The CHAP-Password AVP (AVP Code 3) is of type Complex and contains the response value provided by a PPP Challenge-Handshake Authentication Protocol (CHAP) [6] user in response to the challenge.

If the CHAP-Password AVP is found in a message, the CHAP-Challenge AVP (see section 3.1.1.3) MUST be present as well.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                   AVP Header (AVP Code = 3)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  CHAP Ident   |    Data ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The CHAP Ident field contains the one octet CHAP Identifier from the user's CHAP response [6]. The Data field is 16 octets, and contains the CHAP Response from the user. The actual computation of the CHAP response can be found in [6].

3.1.1.3 CHAP-Challenge AVP

The CHAP-Challenge AVP (AVP Code 60) is of type OctetString and contains the CHAP Challenge sent by the NAS to a PPP Challenge-Handshake Authentication Protocol (CHAP) [6] user.

3.1.1.4 ARAP-Password AVP

The ARAP-Password AVP (AVP Code 70) is of type OctetString and is only present when the Framed-Protocol AVP (see Section 7.2.1) is included in the message and is set to ARAP. This AVP MUST NOT be present if the User-Password or CHAP-Password AVPs are present. See [32] for more information on the contents of this AVP.

3.1.2 AA-Answer (AAA) Command

The AA-Answer (AAA) message, indicated by the Command-Code field set to 266, is sent in response to the AA-Request message. If authorization was requested, a successful response will include the authorization AVPs appropriate for the service being provided, as defined in section 2.0, 6.0 and 7.0

Message Format

```
<AA-Answer> ::= < Diameter Header: 266 >
                { Session-Id }
                { Result-Code }
                { Host-Name }
                [ User-Name ]
              * [ AVP ]
              * [ Proxy-State ]
              * [ Route-Record ]
              * [ Routing-Realm ]
             0*1< Integrity-Check-Value >
```

3.1.2.1 ARAP-Challenge-Response AVP

The ARAP-Challenge-Response AVP (AVP Code 84) is of type OctetString and is only present when the Framed-Protocol AVP (see Section 7.2.1)

is included in the message and is set to ARAP. This AVP contains an 8 octet response to the dial-in client's challenge. The RADIUS server calculates this value by taking the dial-in client's challenge from the high order 8 octets of the ARAP-Password AVP and performing DES encryption on this value with the authenticating user's password as the key. If the user's password is less than 8 octets in length, the password is padded at the end with NULL octets to a length of 8 before using it as a key.

3.1.2.2 Password-Retry AVP

The Password-Retry AVP (AVP Code 75) is of type Unsigned32 and MAY be included in the AA-Answer if the Result-Code indicates an authentication failure. The value of this AVP indicates how many authentication attempts a user may be permitted before being disconnected. This AVP is primarily intended for use when the Framed-Protocol AVP (see Section 7.2.1) is set to ARAP.

3.1.3 AA-Challenge-Ind (ACI) Command

The AA-Challenge-Ind (ACI) message, indicated by the Command-Code field set to 267, is sent by a Diameter Home server to issue a challenge requiring a response to a dial-up user. The message MAY have one or more Reply-Message AVP, the User-Name AVP and it MAY have zero or one State AVP. No other AVPs are permitted in an AA-Challenge-Ind other than security related AVPs [2, 13].

On receipt of an AA-Challenge-Ind, the Identifier field is matched with a pending AA-Request. Invalid messages are silently discarded.

The receipt of a valid AA-Challenge-Ind indicates that a new AA-Request SHOULD be sent. The NAS MAY display the text message, if any, to the user, and then prompt the user for a response. It then sends its original AA-Request with a new request ID, with the User-Password AVP replaced by the user's response (encrypted), and including the State AVP from the AA-Challenge-Ind, if any.

A NAS that supports PAP MAY forward the Reply-Message to the dial-in client and accept a PAP response which it can use as though the user had entered the response. If the NAS cannot do so, it should treat the AA-Challenge-Ind as though it had received an AA-Answer with a Result-Code AVP set to a value other than DIAMETER_SUCCESS instead.

When possible, authentication mechanisms that include more than a single authentication round trip SHOULD use EAP (see section 4.0) instead of the AA-Challenge-Ind. This command has been maintained for

RADIUS backward compatibility.

```
   AA-Challenge-Ind ::= < Diameter Header: 267 >
                        { Session-Id }
                        { Result-Code }
                        { Host-Name }
                        [ User-Name ]
                        [ State ]
                      * [ AVP ]
                      * [ Reply-Message ]
                      * [ Proxy-State ]
                      * [ Route-Record ]
                      * [ Routing-Realm ]
                     0*1< Integrity-Check-Value >
```

3.1.3.1 Prompt AVP

The Prompt AVP (AVP Code 76) is of type Unsigned32, and MAY be present in the AA-Challenge-Ind message. When present, it is used by the NAS to determine whether the user's response, when entered, should be echoed. The following values are defined:

```
   0      No Echo
   1      Echo
```

3.2 Reply-Message AVP

The Reply-Message AVP (AVP Code 18) is of type OctetString and contains text which MAY be displayed to the user. When used in an AA-Answer message with a successful Result-Code AVP it indicates the success message. When found in the same message with a Result-Code other than Diameter-SUCCESS it contains the failure message.

The Reply-Message AVP MAY indicate a dialog message to prompt the user before another AA-Request attempt. When used in an AA-Challenge-Ind, it MAY indicate a dialog message to prompt the user for a response.

Multiple Reply-Message's MAY be included and if any are displayed, they MUST be displayed in the same order as they appear in the message.

4.0 Extensible Authentication Protocol Support

The Extensible Authentication Protocol (EAP), described in [25], provides a standard mechanism for support of additional

authentication methods within PPP. Through the use of EAP, support for a number of authentication schemes may be added, including smart and token cards, Kerberos, Public Key, One Time Passwords, and others.

This section describes the Command-Codes values and AVPs that are required for an EAP payload to be encapsulated within the Diameter protocol. Since authentication occurs between the PPP client and its home Diameter server, end-to-end authentication is achieved, reducing the possibility for fraudulent authentication, such as replay and man-in-the-middle attacks. End-to-end authentication also provides for mutual (bi-directional) authentication, which is not possible with PAP and CHAP in a roaming environment.

The Diameter/EAP extension allows a home Diameter server to initiate an unsolicited authentication request to the user. This allows the home server to periodically ensure that the user is still active, which is useful when a server requires re-authentication to extend the "life" of a session [26]. Server-initiated authentication can reduce the number of protocol exchanges over the Internet.

The EAP conversation between the authenticating peer and the NAS begins with the negotiation of EAP within LCP. Once EAP has been negotiated, the NAS will typically send to the Diameter server a Diameter-EAP-Request message with a NULL EAP-Payload AVP, signifying an EAP-Start. The Port number and NAS Identifier MUST be included in the AVPs issued by the NAS in the Diameter-EAP-Request packet.

If the Diameter home server supports EAP, it MUST respond with a Diameter-EAP-Ind message containing an EAP-Payload AVP that includes an encapsulated EAP payload [25]. The EAP payload is forwarded by the NAS to the PPP client. The initial Diameter-EAP-Ind normally includes an EAP-Request/Identity, requesting the PPP client to identify itself. Upon receipt of the PPP client's EAP-Response [25], the NAS will then issue a second Diameter-EAP-Request message, with the client's EAP payload encapsulated within the EAP-Payload AVP. The conversation continues until the Diameter server sends a Diameter-EAP-Answer with a Result-Code AVP indicating success or failure. A Result-Code AVP containing a failure indication SHOULD also include an EAP-Payload AVP containing an EAP-Failure [25] payload, and the NAS SHOULD disconnect the PPP client by issuing a LCP terminate. If the Result-Code AVP indicates success, the EAP-Payload AVP MUST encapsulate an EAP-Success [25] payload, and the NAS SHOULD successfully terminate the PPP authentication phase. If authorization was requested, a successful Diameter-EAP-Answer MUST also include the appropriate authorization AVPs required for the service requested (see sections 2.0, 6.0 and 7.0).

The above scenario creates a situation in which the NAS never needs to manipulate an EAP packet. An alternative may be used in situations where an EAP-Request/Identity message will always be sent by the NAS to the authenticating peer. This involves having the NAS send an EAP-Request/Identity message to the PPP client, and forwarding the EAP-Response/Identity packet to the Diameter server in the EAP-Payload AVP of a Diameter-EAP-Request packet. While this approach will save a round-trip, it cannot be universally employed. There are circumstances in which the user's identity may not be needed (such as when authentication and accounting is handled based on the calling or called phone number), and therefore an EAP-Request/Identity packet may not necessarily be issued by the NAS to the authenticating peer.

Unless the NAS interprets the EAP-Response/Identity packet returned by the authenticating peer, it will not have access to the user's identity. Therefore, the Diameter Server SHOULD return the user's identity by inserting it in the User-Name attribute of subsequent Diameter-EAP-Answer packets. Without the user's identity, the Session-Id AVP MAY be used for accounting and billing, however operationally this MAY be very difficult to manage.

The Diameter-EAP-Ind message MAY be sent by a Diameter server in order to initiate an unsolicited authentication of the PPP user, as described in [26]. This functionality allows a home Diameter server to easily extend the "life" of a session for a particular service, while reducing the total number of authentication round-trips, should the NAS initiate the periodic authentication.

Should an EAP authentication session be interrupted due to a home server failure, the session MAY be directed to an alternate server, but the authentication session will have to be restarted from the beginning.

When Diameter is used in a roaming environment, the NAS SHOULD issue the EAP-Request/Identity request to the PPP client, and forward the EAP-Response in the initial Diameter-EAP-Request message. This allows any Diameter proxies or brokers to identify the user, and forward the message to the appropriate home server. If a response is received with the Result-Code set to DIAMETER_COMMAND_UNSUPPORTED [2], it is an indication that a Diameter server in the proxy chain does not support EAP. The NAS MAY re-open LCP and attempt to negotiate another PPP authentication protocol, such as PAP or CHAP. A NAS SHOULD be cautious when determining whether a less secure authentication protocol will be used, since this could be a result of a bidding down attack. See [28] for additional information.

4.1 Alternative uses

Currently the conversation between the backend authentication server and the Diameter server is proprietary because of lack of standardization. In order to increase standardization and provide interoperability between Diameter vendors and backend security vendors, it is recommended that Diameter-encapsulated EAP be used for this conversation.

This has the advantage of allowing the Diameter server to support EAP without the need for authentication-specific code within the Diameter server. Authentication-specific code can then reside on a backend authentication server instead.

In the case where Diameter-encapsulated EAP is used in a conversation between a Diameter server and a backend authentication server, the latter will typically return an Diameter-EAP-Answer/EAP-Payload/EAP-Success message without inclusion of the expected authorization AVPs required in a successful response. This means that the Diameter server MUST add these attributes prior to sending an Diameter-EAP-Answer/EAP-Payload/EAP-Success message to the NAS.

4.2 Command-Codes Values

This section defines new Command-Code [2] values that MUST be supported by all Diameter implementations conforming to this specification. The following Command Codes are defined in this section:

Command-Name	Abbrev.	Code	Reference
Diameter-EAP-Answer	DEA	269	4.2.2
Diameter-EAP-Ind	DEI	270	4.2.3
Diameter-EAP-Request	DER	268	4.2.1

4.2.1 Diameter-EAP-Request (DER) Command

The Diameter-EAP-Request (DER) command, indicated by the Command-Code field set to 268, is sent by a Diameter client to a Diameter server and conveys an EAP-Response [25] from the dial-up PPP client. The Diameter-EAP-Request MUST contain one EAP-Payload AVP, which contains the actual EAP payload. An EAP-Payload AVP with no data MAY be sent to the Diameter server to initiate an EAP authentication session.

Upon receipt of a Diameter-EAP-Request, a Diameter server MUST issue a reply. The reply may be either:

1) a Diameter-EAP-Ind containing an EAP-Request encapsulated within an EAP-Payload attribute
2) a Diameter-EAP-Answer containing an EAP-Success encapsulated within an EAP-Payload and a Result-Code indicating success.
3) a Diameter-EAP-Answer containing an EAP-Failure encapsulated within an EAP-Payload and a Result-Code indicating failure.
4) A Message-Reject-Ind packet with a Result-Code set to DIAMETER_COMMAND_UNSUPPORTED if a Diameter server does not support the EAP extension.

Message Format

```
<Diameter-EAP-Request> ::= < Diameter Header: 268 >
                           { Session-Id }
                           { User-Name }
                           { Host-Name }
                           { EAP-Payload }
                           [ NAS-IP-Address ]
                           [ NAS-Identifier ]
                           [ State ]
                         * [ AVP ]
                         * [ Proxy-State ]
                         * [ Route-Record ]
                         * [ Routing-Realm ]
                        0*1< Integrity-Check-Value >
```

4.2.2 Diameter-EAP-Answer (DEA) Command

The Diameter-EAP-Answer (DEA) message, indicated by the Command-Code field set to 269, is sent by the Diameter server to the client to indicate either a successful or failed authentication. The Diameter-EAP-Answer message SHOULD include an EAP payload of type EAP-Success or EAP-Failure encapsulated within an EAP-Payload AVP. The Result-Code AVP MUST indicate a failure if the EAP-Failure payload is present, while the AVP MUST indicate success if the EAP-Success payload is present.

If the message from the Diameter client included a request for authorization, a successful response MUST include the authorization AVPs that are relevant to the service being provided.

Message Format

```
<Diameter-EAP-Answer> ::= < Diameter Header: 269 >
                          { Session-Id }
                          { Result-Code }
                          { Host-Name }
                          [ EAP-Payload ]
                          [ User-Name ]
                        * [ AVP ]
                        * [ Proxy-State ]
                        * [ Route-Record ]
                        * [ Routing-Realm ]
                       0*1< Integrity-Check-Value >
```

4.2.3 Diameter-EAP-Ind (DEI) Command

The Diameter-EAP-Ind (DEI) command, indicated by the Command-Code set to 270, has two uses. This message MAY be sent in response to a Diameter-EAP-Request message, and MUST contain an EAP-Response payload [25] encapsulated within an EAP-Payload AVP.

Alternatively, this message MAY also be sent unsolicited from a Diameter server to a client to request re-authentication of a PPP client. For re-authentication, it is recommended that the Identity request be skipped in order to reduce the number of authentication round trips. This is only possible when the user's identity is already known by the home Diameter server.

Upon receipt of the message, the NAS MUST issue the EAP payload to the PPP Client, and SHOULD respond with a Diameter-EAP-Request containing the EAP-Response [25] packet.

Message Format

```
<Diameter-EAP-Ind> ::= <Diameter Header: 270>
                        { Session-Id }
                        { Host-Name }
                        { EAP-Payload }
                        { User-Name }
                      * [ AVP ]
                      * [ Proxy-State ]
                      * [ Route-Record ]
                      * [ Routing-Realm ]
                     0*1< Integrity-Check-Value >
```

4.3 EAP-Payload AVP

The EAP-Payload AVP (AVP Code 402) is of type OctetString and is used

to encapsulate the actual EAP payload [25] that is being exchanged between the dial-up PPP client and the home Diameter server.

5.0 Diameter Session Termination

When a Network Access Server (NAS) receives an indication that a user's session is being disconnected (e.g. LCP Terminate is received), the NAS MUST issue a Session-Termination-Request (STR) [2] to its Diameter Server. This will ensure that any resources maintained on the servers is freed appropriately.

Further, a NAS that receives a Session-Termination-Ind (STI) [2] MUST disconnect the PPP (or tunneling) session and respond with an STR message.

6.0 Call and Session Information

This section contains the authorization AVPs that are needed to identify call and session information, and allows the server to set constraints on a session.

6.1 NAS-Port AVP

The NAS-Port AVP (AVP Code 5) is of type Unsigned32 and contains the physical port number of the NAS which is authenticating the user, and is normally only present in an authentication and/or authorization request. Note that this is using "port" in its sense of a physical connection on the NAS, not in the sense of a TCP or UDP port number. Either NAS-Port or NAS-Port-Type (AVP Code 61) or both SHOULD be present in the request, if the NAS differentiates among its ports.

6.2 Filter-Id AVP

The Filter-Id AVP (AVP Code 11) is of type OctetString and contains the name of the filter list for this user. Zero or more Filter-Id AVPs MAY be sent in an authorization response.

Identifying a filter list by name allows the filter to be used on different NASes without regard to filter-list implementation details. However, this AVP is not roaming friendly since filter naming differs from one service provider to another.

In non-RADIUS environments, it is strongly recommended that the Filter-Rule AVP be used instead.

6.3 Callback-Number AVP

The Callback-Number AVP (AVP Code 19) is of type OctetString and contains a dialing string to be used for callback. It MAY be used in an authentication and/or authorization request as a hint to the server that a Callback service is desired, but the server is not required to honor the hint in the corresponding response.

The codification of the range of allowed usage of this field is outside the scope of this specification.

6.4 Callback-Id AVP

The Callback-Id AVP (AVP Code 20) is of type OctetString and contains the name of a place to be called, to be interpreted by the NAS. This AVP MAY be present in an authentication and/or authorization response.

This AVP is not roaming friendly since it assumes that the Callback-Id is configured on the NAS. It is therefore preferable to use the Callback-Number AVP instead.

6.5 Idle-Timeout AVP

The Idle-Timeout AVP (AVP Code 28) is of type Unsigned32 and sets the maximum number of consecutive seconds of idle connection allowed to the user before termination of the session or prompt. It MAY be used in an authentication and/or authorization request (or challenge) as a hint to the server that an idle timeout is desired, but the server is not required to honor the hint in the corresponding response.

6.6 Called-Station-Id AVP

The Called-Station-Id AVP (AVP Code 30) is of type OctetString and allows the NAS to send in the request the phone number that the user called, using Dialed Number Identification (DNIS) or a similar technology. Note that this may be different from the phone number the call comes in on. It SHOULD only be present in authentication and/or authorization requests.

If the Request-Type AVP is set to authorization-only and the User-Name AVP is absent, the Diameter Server MAY perform authorization based on this field. This can be used by a NAS to request whether a call should be answered based on the DNIS.

The codification of the range of allowed usage of this field is outside the scope of this specification.

6.7 Calling-Station-Id AVP

The Calling-Station-Id AVP (AVP Code 31) is of type OctetString and allows the NAS to send in the request the phone number that the call came from, using Automatic Number Identification (ANI) or a similar technology. It SHOULD only be present in authentication and/or authorization requests.

If the Request-Type AVP is set to authorization-only and the User-Name AVP is absent, the Diameter Server MAY perform authorization based on this field. This can be used by a NAS to request whether a call should be answered based on the ANI.

The codification of the range of allowed usage of this field is outside the scope of this specification.

6.8 NAS-Port-Type AVP

The NAS-Port-Type AVP (AVP Code 61) is of type Unsigned32 and contains the type of the physical port of the NAS which is authenticating the user. It can be used instead of or in addition to the NAS-Port (5) AVP. This AVP SHOULD only be used in authentication and/or authorization requests. This AVP MAY be combined with the NAS-Port AVP to assist in differentiating its ports.

The following values are defined:

```
0       Async
1       Sync
2       ISDN Sync
3       ISDN Async V.120
4       ISDN Async V.110
5       Virtual
6       PIAFS
7       HDLC Clear Channel
8       X.25
9       X.75
10      G.3 Fax
11      SDSL - Symmetric DSL
12      ADSL-CAP - Asymmetric DSL, Carrierless Amplitude Phase Modulation
13      ADSL-DMT - Asymmetric DSL, Discrete Multi-Tone
14      IDSL - ISDN Digital Subscriber Line
15      Ethernet
```

```
16      xDSL
17      Cable
18      Wireless - Other
19      Wireless - IEEE 802.11
```

"Virtual" refers to a connection to the NAS via some transport protocol, instead of through a physical port. For example, if a user telnetted into a NAS to authenticate himself as an Outbound-User, the request might include NAS-Port-Type = Virtual as a hint to the Diameter server that the user was not on a physical port.

6.9 Port-Limit AVP

The Port-Limit AVP (AVP Code 62) is of type Unsigned32 and sets the maximum number of ports to be provided to the user by the NAS. It MAY be used in an authentication and/or authorization request as a hint to the server that multilink PPP [9] service is desired, but the server is not required to honor the hint in the corresponding response.

6.10 Connect-Info AVP

The Connect-Info AVP (AVP Code 77) is of type OctetString and is sent in the AA-Request message, and indicates the nature of the user's connection. The value consists of UTF-8 encoded 10646 characters. The connection speed SHOULD be included at the beginning of the first Connect-Info AVP in the message. If the transmit and receive connection speeds differ, they may both be included in the first AVP with the transmit speed first (the speed the NAS modem transmits at), a slash (/), the receive speed, then optionally other information.

7.0 Service Specific Authorization AVPs

This section contains the RADIUS authorization AVPs that are supported in the Diameter protocol. The Service-Type AVP MUST be present in all messages, and based on the value of the Service-Type AVP, additional AVPs defined in sections 7.2, 7.3 and 7.4 MAY be present.

7.1 Service-Type AVP

The Service-Type AVP (AVP Code 6) is of type Unsigned32 and contains the type of service the user has requested, or the type of service to be provided. One such AVP MAY be present in an authentication and/or

authorization request or response. A NAS is not required to implement all of these service types, and MUST treat unknown or unsupported Service-Types as though a response with a Result-Code other than Diameter-SUCCESS had been received instead.

When used in a request, the Service-Type AVP SHOULD be considered to be a hint to the server that the NAS has reason to believe the user would prefer the kind of service indicated, but the server is not required to honor the hint. The following values have been defined for the Service-Type AVP:

Login   1
   The user should be connected to a host. The message MAY include additional AVPs defined in section 7.3.

Framed   2
   A Framed Protocol should be started for the User, such as PPP or SLIP. The message MAY include additional AVPs defined in section 7.2, or 7.4 for tunneling services.

Callback Login   3
   The user should be disconnected and called back, then connected to a host. The message MAY include additional AVPs defined in section 7.3.

Callback Framed   4
   The user should be disconnected and called back, then a Framed Protocol should be started for the User, such as PPP or SLIP. The message MAY include additional AVPs defined in section 7.2, or 7.4 for tunneling services.

Outbound   5
   The user should be granted access to outgoing devices.

Administrative   6
   The user should be granted access to the administrative interface to the NAS from which privileged commands can be executed.

NAS Prompt   7
   The user should be provided a command prompt on the NAS from which non-privileged commands can be executed.

Authenticate Only   8
   Only Authentication is requested, and no authorization information needs to be returned in the response.

Callback NAS Prompt 9
   The user should be disconnected and called back, then provided a

command prompt on the NAS from which non-privileged commands can be executed.

7.2 Framed Access Authorization AVPs

This section contains the authorization AVPs that are necessary to support framed access, such as PPP, SLIP, etc. AVPs defined in this section MAY be present in a message if the Service-Type AVP was set to "Framed" or "Callback Framed".

7.2.1 Framed-Protocol AVP

The Framed-Protocol AVP (AVP Code 7) is of type Unsigned32 and contains the framing to be used for framed access. This AVP MAY be present in both requests and responses. The following values are currently supported:

1 PPP
2 SLIP
3 AppleTalk Remote Access Protocol (ARAP)
4 Gandalf proprietary SingleLink/MultiLink protocol
5 Xylogics proprietary IPX/SLIP
6 X.75 Synchronous

7.2.2 Framed-Routing AVP

The Framed-Routing AVP (AVP Code 10) is of type Unsigned32 and contains the routing method for the user, when the user is a router to a network. This AVP SHOULD only be present in authorization responses. The following values are defined for this AVP:

0 None
1 Send routing packets
2 Listen for routing packets
3 Send and Listen

7.2.3 Framed-MTU AVP

The Framed-MTU AVP (AVP Code 12) is of type Unsigned32 and contains the Maximum Transmission Unit to be configured for the user, when it is not negotiated by some other means (such as PPP). This AVP SHOULD only be present in authorization responses. The MTU value MUST be between the range of 64 and 65535.

7.2.4 Framed-Compression AVP

The Framed-Compression AVP (AVP Code 13) is of type Unsigned32 and contains the compression protocol to be used for the link. It MAY be used in an authorization request as a hint to the server that a specific compression type is desired, but the server is not required to honor the hint in the corresponding response.

More than one compression protocol AVP MAY be sent. It is the responsibility of the NAS to apply the proper compression protocol to appropriate link traffic.

The following values are defined:

```
   0      None
   1      VJ TCP/IP header compression [7]
   2      IPX header compression
   3      Stac-LZS compression
```

7.2.5 IP Access

The AVPs defined in this section are used when the user requests, or is being granted, access to IP. They are only present if the Framed-Protocol AVP (see Section 7.2.1) is set to PPP, SLIP, Gandalf proprietarySingleLink/MultiLink protocol, or X.75 Synchronous.

7.2.5.1 Framed-IP-Address AVP

The Framed-IP-Address AVP (AVP Code 8) is of type Address and contains the address to be configured for the user. It MAY be used in an authorization request as a hint to the server that a specific address is desired, but the server is not required to honor the hint in the corresponding response.

Two addresses have special significance; 0xFFFFFFFF and 0xFFFFFFFE. The value 0xFFFFFFFF indicates that the NAS should allow the user to select an address (e.g. Negotiated). The value 0xFFFFFFFE indicates that the NAS should select an address for the user (e.g. Assigned from a pool of addresses kept by the NAS).

7.2.5.2 Framed-IP-Netmask AVP

The Framed-IP-Netmask AVP (AVP Code 9) is of type Address and contains the IP netmask to be configured for the user when the user is a router to a network. It MAY be used in an authorization request as a hint to the server that a specific netmask is desired, but the

server is not required to honor the hint in the corresponding response. This AVP MUST be present in a response if the request included this AVP with a value of 0xFFFFFFFF.

7.2.5.3 Framed-IP-Route AVP

The Framed-IP-Route AVP (AVP Code 22) is of type OctetString and contains the routing information to be configured for the user on the NAS. Zero or more such AVPs MAY be present in an authorization response.

The string MUST contain a destination prefix in dotted quad form optionally followed by a slash and a decimal length specifier stating how many high order bits of the prefix should be used. That is followed by a space, a gateway address in dotted quad form, a space, and one or more metrics separated by spaces. For example, "192.168.1.0/24 192.168.1.1 1".

The length specifier may be omitted in which case it should default to 8 bits for class A prefixes, 16 bits for class B prefixes, and 24 bits for class C prefixes. For example, "192.168.1.0 192.168.1.1 1".

Whenever the gateway address is specified as "0.0.0.0" the IP address of the user SHOULD be used as the gateway address.

7.2.6 IPX Access

The AVPs defined in this section are used when the user requests, or is being granted, access to IPX. They are only present if the Framed-Protocol AVP (see Section 7.2.1) is set to PPP, Xylogics proprietary IPX/SLIP, Gandalf proprietarySingleLink/MultiLink protocol, or X.75 Synchronous.

7.2.6.1 Framed-IPX-Network AVP

The Framed-IPX-Network AVP (AVP Code 23) is of type OctetString and contains the IPX Network number to be configured for the user. It MAY be used in an authorization request as a hint to the server that a specific address is desired, but the server is not required to honor the hint in the corresponding response.

Two addresses have special significance; 0xFFFFFFFF and 0xFFFFFFFE. The value 0xFFFFFFFF indicates that the NAS should allow the user to select an address (e.g. Negotiated). The value 0xFFFFFFFE indicates that the NAS should select an address for the user (e.g. assigned

from a pool of one or more IPX networks kept by the NAS).

7.2.7 Appletalk Access

The AVPs defined in this section are used when the user requests, or is being granted, access to Appletalk. They are only present if the Framed-Protocol AVP (see Section 7.2.1) is set to PPP, Gandalf proprietary SingleLink/MultiLink protocol, or X.75 Synchronous.

7.2.7.1 Framed-AppleTalk-Link AVP

The Framed-AppleTalk-Link AVP (AVP Code 37) is of type Unsigned32 and contains the AppleTalk network number which should be used for the serial link to the user, which is another AppleTalk router. This AVP MUST only be present in an authorization response and is never used when the user is not another router.

Despite the size of the field, values range from zero to 65535. The special value of zero indicates that this is an unnumbered serial link. A value of one to 65535 means that the serial line between the NAS and the user should be assigned that value as an AppleTalk network number.

7.2.7.2 Framed-AppleTalk-Network AVP

The Framed-AppleTalk-Network AVP (AVP Code 38) is of type Unsigned32 and contains the AppleTalk Network number which the NAS should probe to allocate an AppleTalk node for the user. This AVP MUST only be present in an authorization response and is never used when the user is not another router. Multiple instances of this AVP indicate that the NAS may probe using any of the network numbers specified.

Despite the size of the field, values range from zero to 65535. The special value zero indicates that the NAS should assign a network for the user, using its default cable range. A value between one and 65535 (inclusive) indicates the AppleTalk Network the NAS should probe to find an address for the user.

7.2.7.3 Framed-AppleTalk-Zone AVP

The Framed-AppleTalk-Zone AVP (AVP Code 39) is of type OctetString and contains the AppleTalk Default Zone to be used for this user. This AVP MUST only be present in an authorization response. Multiple instances of this AVP in the same message are not allowed.

The codification of the range of allowed usage of this field is outside the scope of this specification.

7.2.8 ARAP Access

The AVPs defined in this section are used when the user requests, or is being granted, access to ARAP. They are only present if the Framed-Protocol AVP (see Section 7.2.1) is set to AppleTalk Remote Access Protocol (ARAP).

7.2.8.1 ARAP-Features AVP

The ARAP-Features AVP (AVP Code 71) is of type OctetString, and MAY be present in the AA-Accept message if the Framed-Protocol AVP is set to the value of ARAP. See [32] for more information of the format of this AVP.

7.2.8.2 ARAP-Zone-Access AVP

The ARAP-Zone-Access AVP (AVP Code 72) is of type Unsigned32, and MAY be present in the AA-Accept message if the Framed-Protocol AVP is set to the value of ARAP. The following values are supported:

```
1       Only allow access to default zone
2       Use zone filter inclusively
4       Use zone filter exclusively
```

The value 3 is skipped, not because these are bit flags, but because 3 in some ARAP implementations means "all zones" which is the same as not specifying a list at all under RADIUS.

If this attribute is present and the value is 2 or 4 then a Filter-Id must also be present to name a zone list filter to apply the access flag to.

7.2.8.3 ARAP-Security AVP

The ARAP-Security AVP (AVP Code 73) is of type Unsigned32, and MAY be present in the AA-Challenge-Ind message if the Framed-Protocol AVP is set to the value of ARAP. See [32] for more information of the format of this AVP.

7.2.8.4 ARAP-Security-Data AVP

The ARAP-Security AVP (AVP Code 74) is of type OctetString, and MAY be present in the AA-Request or AA-Challenge-Ind message if the Framed-Protocol AVP is set to the value of ARAP. This AVP contains the security module challenge or response associated with the ARAP Security Module specified in ARAP-Security.

7.3 Non-Framed Access Authorization AVPs

This section contains the authorization AVPs that are needed to support terminal server functionality. AVPs defined in this section MAY be present in a message if the Service-Type AVP was set to "Login" or "Callback Login".

7.3.1 Login-IP-Host AVP

The Login-IP-Host AVP (AVP Code 14) is of type Address and contains the system with which to connect the user, when the Login-Service AVP is included. It MAY be used in an authorization request as a hint to the server that a specific host is desired, but the server is not required to honor the hint in the corresponding response.

Two addresses have special significance; 0xFFFFFFFF and 0xFFFFFFFE. The value 0xFFFFFFFF indicates that the NAS SHOULD allow the user to select an address. The value zero indicates that the NAS SHOULD select a host to connect the user to.

7.3.2 Login-Service AVP

The Login-Service AVP (AVP Code 15) is of type Unsigned32 and contains the service which should be used to connect the user to the login host. This AVP SHOULD only be present in authorization responses.

The following values are defined:

```
0      Telnet
1      Rlogin
2      TCP Clear
3      PortMaster (proprietary)
4      LAT
5      X25-PAD
6      X25-T3POS
8      TCP Clear Quiet (supresses any NAS-generated connect string)
```

7.3.3 TCP Services

The AVP described in this section MAY be present if the Login-Service AVP is set to Telnet, Rlogin, TCP Clear or TCP Clear Quiet.

7.3.3.1 Login-TCP-Port AVP

The Login-TCP-Port AVP (AVP Code 16) is of type Unsigned32 and contains the TCP port with which the user is to be connected, when the Login-Service AVP is also present. This AVP SHOULD only be present in authorization responses. The value MUST NOT be greater than 65535.

7.3.4 LAT Services

The AVP described in this section MAY be present if the Login-Service AVP is set to LAT.

7.3.4.1 Login-LAT-Service AVP

The Login-LAT-Service AVP (AVP Code 34) is of type OctetString and contains the system with which the user is to be connected by LAT. It MAY be used in an authorization request as a hint to the server that a specific service is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST only be present in the response if the Login-Service AVP states that LAT is desired.

Administrators use the service attribute when dealing with clustered systems, such as a VAX or Alpha cluster. In such an environment several different time sharing hosts share the same resources (disks, printers, etc.), and administrators often configure each to offer access (service) to each of the shared resources. In this case, each host in the cluster advertises its services through LAT broadcasts.

Sophisticated users often know which service providers (machines) are faster and tend to use a node name when initiating a LAT connection. Alternately, some administrators want particular users to use certain machines as a primitive form of load balancing (although LAT knows how to do load balancing itself).

The String field contains the identity of the LAT service to use. The LAT Architecture allows this string to contain $ (dollar), - (hyphen), . (period), _ (underscore), numerics, upper and lower case alphabetics, and the ISO Latin-1 character set extension [8]. All LAT string comparisons are case insensitive.

7.3.4.2 Login-LAT-Node AVP

The Login-LAT-Node AVP (AVP Code 35) is of type OctetString and contains the Node with which the user is to be automatically connected by LAT. It MAY be used in an authorization request as a hint to the server that a specific LAT node is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST only be present in a response if the Service-Type AVP is set to LAT.

The String field contains the identity of the LAT service to use. The LAT Architecture allows this string to contain $ (dollar), - (hyphen), . (period), _ (underscore), numerics, upper and lower case alphabetics, and the ISO Latin-1 character set extension [8]. All LAT string comparisons are case insensitive.

7.3.4.3 Login-LAT-Group AVP

The Login-LAT-Group AVP (AVP Code 36) is of type OctetString and contains a string identifying the LAT group codes which this user is authorized to use. It MAY be used in an authorization request as a hint to the server that a specific group is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST only be present in a response if the Service-Type AVP is set to LAT.

LAT supports 256 different group codes, which LAT uses as a form of access rights. LAT encodes the group codes as a 256 bit bitmap.

Administrators can assign one or more of the group code bits at the LAT service provider; it will only accept LAT connections that have these group codes set in the bit map. The administrators assign a bitmap of authorized group codes to each user; LAT gets these from the operating system, and uses these in its requests to the service providers.

The codification of the range of allowed usage of this field is outside the scope of this specification.

7.3.4.4 Login-LAT-Port AVP

The Login-LAT-Port AVP (AVP Code 63) is of type OctetString and contains the Port with which the user is to be connected by LAT. It MAY be used in an authorization request as a hint to the server that a specific port is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST only be present

in a response if the Service-Type AVP is set to LAT.

The String field contains the identity of the LAT service to use. The LAT Architecture allows this string to contain $ (dollar), - (hyphen), . (period), _ (underscore), numerics, upper and lower case alphabetics, and the ISO Latin-1 character set extension [8]. All LAT string comparisons are case insensitive.

7.4 Tunneling AVPs

This section contains the authorization AVPs that are needed for a NAS to support tunneling users.

7.4.1 Tunnel-Type AVP

The Tunnel-Type AVP (AVP Code 64) is of type Unsigned32 and contains the tunneling protocol(s) to be used (in the case of a tunnel initiator) or the the tunneling protocol in use (in the case of a tunnel terminator). It MAY be used in an authorization request as a hint to the server that a specific tunnel type is desired, but the server is not required to honor the hint in the corresponding response.

The Tunnel-Type SHOULD also be present in the corresponding ADIF Record within the Accounting-Request.

A tunnel initiator is not required to implement any of these tunnel types; if a tunnel initiator receives a response that contains only unknown or unsupported Tunnel-Types, the tunnel initiator MUST behave as though a response was received with the Result-Code indicating a failure.

The following values have been defined:

```
1       Point-to-Point Tunneling Protocol (PPTP) [14]
2       Layer Two Forwarding (L2F) [15]
3       Layer Two Tunneling Protocol (L2TP) [16]
4       Ascend Tunnel Management Protocol (ATMP) [17]
5       Virtual Tunneling Protocol (VTP)
6       IP Authentication Header in the Tunnel-mode (AH) [18]
7       IP-in-IP Encapsulation (IP-IP) [19]
8       Minimal IP-in-IP Encapsulation (MIN-IP-IP) [20]
9       IP Encapsulating Security Payload in the Tunnel-mode (ESP) [21]
10      Generic Route Encapsulation (GRE) [22]
11      Bay Dial Virtual Services (DVS)
12      IP-in-IP Tunneling [23]
```

7.4.2 Tunnel-Medium-Type AVP

The Tunnel-Medium-Type AVP (AVP Code 65) is of type Unsigned32 and contains the transport medium to use when creating a tunnel for those protocols (such as L2TP) that can operate over multiple transports. It MAY be used in an authorization request as a hint to the server that a specific medium is desired, but the server is not required to honor the hint in the corresponding response.

The Value field is three octets and contains one of the values listed under "Address Family Numbers" in [10]. The value of most importance is (1) for IPv4 and (2) for IPv6.

7.4.3 Tunnel-Client-Endpoint AVP

The Tunnel-Client-Endpoint AVP (AVP Code 66) is of type OctetString and contains the address of the initiator end of the tunnel. It MAY be used in an authorization request as a hint to the server that a specific endpoint is desired, but the server is not required to honor the hint in the corresponding response.

This AVP SHOULD be included in the ADIF Record of the corresponding Accounting-Request messages, in which case it indicates the address from which the tunnel was initiated. This AVP, along with the Tunnel-Server-Endpoint and Session-Id AVP [2], MAY be used to provide a globally unique means to identify a tunnel for accounting and auditing purposes.

If Tunnel-Medium-Type is IPv4 (1), then this string is either the fully qualified domain name (FQDN) of the tunnel client machine, or it is a "dotted-decimal" IP address. Conformant implementations MUST support the dotted-decimal format and SHOULD support the FQDN format for IP addresses.

If Tunnel-Medium-Type is IPv6 (2), then this string is either the FQDN of the tunnel client machine, or it is a text representation of the address in either the preferred or alternate form [5]. Conformant implementations MUST support the preferred form and SHOULD support both the alternate text form and the FQDN format for IPv6 addresses.

If Tunnel-Medium-Type is neither IPv4 nor IPv6, this string is a tag referring to configuration data local to the Diameter client that describes the interface and medium-specific address to use.

7.4.4 Tunnel-Server-Endpoint AVP

The Tunnel-Server-Endpoint AVP (AVP Code 67) is of OctetString and contains the address of the server end of the tunnel. It MAY be used in an authorization request as a hint to the server that a specific endpoint is desired, but the server is not required to honor the hint in the corresponding response.

This AVP SHOULD be included in the ADIF Record of the corresponding Accounting-Request messages, in which case it indicates the address from which the tunnel was initiated. This AVP, along with the Tunnel-Client-Endpoint and Session-Id AVP [2], MAY be used to provide a globally unique means to identify a tunnel for accounting and auditing purposes.

If Tunnel-Medium-Type is IPv4 (1), then this string is either the fully qualified domain name (FQDN) of the tunnel client machine, or it is a "dotted-decimal" IP address. Conformant implementations MUST support the dotted-decimal format and SHOULD support the FQDN format for IP addresses.

If Tunnel-Medium-Type is IPv6 (2), then this string is either the FQDN of the tunnel client machine, or it is a text representation of the address in either the preferred or alternate form [5]. Conformant implementations MUST support the preferred form and SHOULD support both the alternate text form and the FQDN format for IPv6 addresses.

If Tunnel-Medium-Type is not IPv4 or IPv6, this string is a tag referring to configuration data local to the Diameter client that describes the interface and medium-specific address to use.

7.4.5 Tunnel-Password AVP

The Tunnel-Password AVP (AVP Code 69) is of type OctetString and may contain a password to be used to authenticate to a remote server. This AVP MUST only be present in authorization responses in an encrypted form, using one of the methods described in [2] and [13].

7.4.6 Tunnel-Private-Group-ID AVP

The Tunnel-Private-Group-ID AVP (AVP Code 81) is of type OctetString and contains the group ID for a particular tunneled session. The Tunnel-Private-Group-ID AVP MAY be included in an authorization request if the tunnel initiator can pre-determine the group resulting from a particular connection and SHOULD be included in the authorization response if this tunnel session is to be treated as belonging to a particular private group. Private groups may be used

to associate a tunneled session with a particular group of users. For example, it MAY be used to facilitate routing of unregistered IP addresses through a particular interface. This value SHOULD be included the corresponding ADIF-Record in the Accounting-Request which pertain to a tunneled session.

7.4.7 Tunnel-Assignment-ID AVP

The Tunnel-Assignment-ID AVP (AVP Code 82) is of type OctetString and is used to indicate to the tunnel initiator the particular tunnel to which a session is to be assigned. Some tunneling protocols, such as PPTP and L2TP, allow for sessions between the same two tunnel endpoints to be multiplexed over the same tunnel and also for a given session to utilize its own dedicated tunnel. This attribute provides a mechanism for Diameter to be used to inform the tunnel initiator (e.g. PAC, LAC) whether to assign the session to a multiplexed tunnel or to a separate tunnel. Furthermore, it allows for sessions sharing multiplexed tunnels to be assigned to different multiplexed tunnels.

A particular tunneling implementation may assign differing characteristics to particular tunnels. For example, different tunnels may be assigned different QOS parameters. Such tunnels may be used to carry either individual or multiple sessions. The Tunnel-Assignment-ID attribute thus allows the Diameter server to indicate that a particular session is to be assigned to a tunnel that provides an appropriate level of service. It is expected that any QOS-related Diameter tunneling attributes defined in the future that accompany this attribute will be associated by the tunnel initiator with the ID given by this attribute. In the meantime, any semantic given to a particular ID string is a matter left to local configuration in the tunnel initiator.

The Tunnel-Assignment-ID AVP is of significance only to Diameter and the tunnel initiator. The ID it specifies is intended to be of only local use to Diameter and the tunnel initiator. The ID assigned by the tunnel initiator is not conveyed to the tunnel peer.

This attribute MAY be included in authorization responses. The tunnel initiator receiving this attribute MAY choose to ignore it and assign the session to an arbitrary multiplexed or non-multiplexed tunnel between the desired endpoints. This attribute SHOULD also be included in the corresponding ADIF-Record in the Accounting-Request messages which pertain to a tunneled session.

If a tunnel initiator supports the Tunnel-Assignment-ID AVP, then it should assign a session to a tunnel in the following manner:

- If this AVP is present and a tunnel exists between the specified endpoints with the specified ID, then the session should be assigned to that tunnel.

- If this AVP is present and no tunnel exists between the specified endpoints with the specified ID, then a new tunnel should be established for the session and the specified ID should be associated with the new tunnel.

- If this AVP is not present, then the session is assigned to an unnamed tunnel. If an unnamed tunnel does not yet exist between the specified endpoints then it is established and used for this and subsequent sessions established without the Tunnel-Assignment-ID attribute. A tunnel initiator MUST NOT assign a session for which a Tunnel-Assignment-ID AVP was not specified to a named tunnel (i.e. one that was initiated by a session specifying this AVP).

Note that the same ID may be used to name different tunnels if such tunnels are between different endpoints.

7.4.8 Tunnel-Preference AVP

The Tunnel-Preference AVP (AVP Code 83) is of type Unsigned32 and is used to identify the relative preference assigned to each tunnel when more than one set of tunneling AVPs is returned within separete Grouped-AVP AVPs. It MAY be used in an authorization request as a hint to the server that a specific preference is desired, but the server is not required to honor the hint in the corresponding response.

For example, suppose that AVPs describing two tunnels are returned by the server, one with a Tunnel-Type of PPTP and the other with a Tunnel-Type of L2TP. If the tunnel initiator supports only one of the Tunnel-Types returned, it will initiate a tunnel of that type. If, however, it supports both tunnel protocols, it SHOULD use the value of the Tunnel-Preference AVP to decide which tunnel should be started. The tunnel having the numerically lowest value in the Value field of this AVP SHOULD be given the highest preference. The values assigned to two or more instances of the Tunnel-Preference AVP within a given authorization response MAY be identical. In this case, the tunnel initiator SHOULD use locally configured metrics to decide which set of AVPs to use.

7.4.9 Tunnel-Client-Auth-ID AVP

The Tunnel-Client-Auth-ID AVP (AVP Code 90) is of type Unsigned32 and specifies the name used by the tunnel initiator during the authentication phase of tunnel establishment. It MAY be used in an authorization request as a hint to the server that a specific preference is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST be present in the authorization response if an authentication name other than the default is desired. This AVP SHOULD be included in the corresponding ADIF-Record of the Accounting-Request messages which pertain to a tunneled session.

7.4.10 Tunnel-Server-Auth-ID AVP

The Tunnel-Server-Auth-ID AVP (AVP Code 91) is of type OctetString and specifies the name used by the tunnel terminator during the authentication phase of tunnel establishment. It MAY be used in an authorization request as a hint to the server that a specific preference is desired, but the server is not required to honor the hint in the corresponding response. This AVP MUST be present in the authorization response if an authentication name other than the default is desired. This AVP SHOULD be included in the corresponding ADIF-Record of the Accounting-Request messages which pertain to a tunneled session.

8.0 Accounting Considerations

This section contains a description of the AVPs defined in this document that are to be included in Diameter accounting messages [29].

8.1 Framed Access

This section contains the AVPs defined in this extension that are to be present in the Accounting-Request and optionally in the Accounting-Answer messages, defined in [29], when the Service-Type AVP specifies Framed service.

```
<Service-Specific AVPs> ::= { Service-Type }
                            { Framed-Protocol }
                            [ Framed-IP-Address ]
                            [ Framed-IP-Netmask ]
                            [ Framed-Routing ]
                            [ Framed-MTU ]
                            [ Framed-Compression ]
                            [ Framed-Route ]
                            [ Framed-IPX-Network ]
                            [ Framed-AppleTalk-Link ]
                            [ Framed-AppleTalk-Network ]
                            [ Framed-AppleTalk-Zone ]
```

8.2 Non-Framed Access

This section contains the AVPs defined in this extension that are to be present in the Accounting-Request and optionally in the Accounting-Answer messages, defined in [29], when the Service-Type AVP specifies non-Framed service.

```
<Service-Specific AVPs> ::= { Service-Type }
                            { Login-Service }
                            [ Login-IP-Host ]
                            [ Login-TCP-Port ]
                            [ Login-LAT-Service ]
                            [ Login-LAT-Node ]
                            [ Login-LAT-Group ]
                            [ Login-LAT-Port ]
```

8.3 Tunneling

Additional work is required to identify how to integrate tunneling in the Accounting extension. One method is as defined in [34], which would require new Accounting-Type messages (e.g. tunnel and link start/stop).

9.0 Interactions with Resource Management

The Resource Management extension [31] provides the ability for a Diameter node to query a peer for session state information. The document states that service-specific extensions are responsible for specifying what AVPs are to be present in the Resource-Token [31] AVP.

In addition to the AVPs listed in [31], the Resource-Token with the

Extension-Id AVP set to one (1) MUST include the Service-Type AVP. In the event of a framed (PPP) user, the Framed-IP-Address and Framed-IPX-Network MUST be present if the corresponding network is being used. For Login users, the Login-IP-Host AVP and Login-Service AVP MUST be present. For tunneling users, the Tunnel-Type, Tunnel-Medium-Type, Tunnel-Client-Endpoint, and the Tunnel-Server-Endpoint AVPs MUST be present.

10.0 IANA Considerations

The command codes defined in Sections 3.1 and 4.2 are values taken from the Command-Code [2] address space and extended in [13], [29] and [30]. IANA should record the values as defined in Sections 2.1 and 4.2.

The AVPs defined in section 2.1 were alllocated from from the AVP numbering space defined in [2], and extended in [13], [29], and [30]. IANA should record the values as defined in Sections 2.1.

The Diameter base protocol [2] reserves the first 255 AVPs for legacy RADIUS support [1]. The AVPs defined in section 2.2 are defined in [1] and [32], and no number assignment is necessary.

10.1 Request-Type AVP Values

The Request-Type AVP (section 2.1.1) has a set of values that MUST be maintained by IANA. Values 1 through 3 are defined in this document. The remaining values are available for assignment via Designated Expert [27].

11.0 Security Considerations

This document does not contain any security protocol, but does discuss how PPP authentication protocols can be carried within the Diameter protocol. The PPP authentication protocols that are described are PAP, CHAP and EAP.

The use of PAP SHOULD be discouraged, since it exposes user's passwords to possibly non-trusted entities. PAP is also frequently used for use with One-Time Passwords (OTP), which does not expose any security risks. However, it is highly recommended that OTP be supported through the EAP protocol.

This document also describes how CHAP can be carried within the Diameter protocol, which is required for backward RADIUS

compatibility. The CHAP protocol, as used in a RADIUS environment, facilitates authentication replay attacks, and therefore SHOULD NOT be used when EAP is available.

12.0 References

[1] Rigney, et alia, "RADIUS", RFC-2138, Livingston, April 1997

[2] Calhoun, Rubens, Akhtar, Guttman, "Diameter Base Protocol", draft-calhoun-diameter-18.txt, IETF work in progress, January 2001.

[3] Aboba, Beadles, "The Network Access Identifier." RFC 2486. January 1999.

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

[5] Hinden, R., Deering, S., "IP Version 6 Addressing Architecture", RFC 2373, July 1998

[6] W. Simpson, "PPP Challenge Handshake Authentication Protocol (CHAP)", RFC 1994, August 1996.

[7] Jacobson, "Compressing TCP/IP headers for low-speed serial links", RFC 1144, February 1990.

[8] ISO 8859. International Standard -- Information Processing -- 8-bit Single-Byte Coded Graphic Character Sets -- Part 1: Latin Alphabet No. 1, ISO 8859-1:1987. <URL:http://www.iso.ch/cate/d16338.html>

[9] Sklower, Lloyd, McGregor, Carr, "The PPP Multilink Protocol (MP)", RFC 1717, November 1994.

[10] Reynolds, J., Postel, J., "Assigned Numbers", STD 2, RFC 1700, October 1994

[11] Calhoun, Zorn, Pan, Akhtar, "Diameter Framework", draft-calhoun-diameter-framework-09.txt, IETF work in progress, January 2001.

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

[13] P. Calhoun, W. Bulley, S. Farrell, "Diameter Strong Security

Extension", draft-calhoun-diameter-strong-crypto-06.txt, IETF work in progress, January 2001.

[14] Hamzeh, K., Pall, G., Verthein, W., Taarud, J., Little, W., Zorn, G., "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637, July 1999

[15] Valencia, A., Littlewood, M., Kolar, T., "Cisco Layer Two Forwarding (Protocol) 'L2F'", RFC 2341, May 1998

[16] Townsley, W. M., Valencia, A., Rubens, A., Pall, G. S., Zorn, G., Palter, B., "Layer Two Tunneling Protocol (L2TP)", RFC 2661, August 1999

[17] Hamzeh, K., "Ascend Tunnel Management Protocol - ATMP", RFC 2107, February 1997

[18] Kent, S., Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, November 1998

[19] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996

[20] Perkins, C., "Minimal Encapsulation within IP", RFC 2004, October 1996

[21] Atkinson, R., "IP Encapsulating Security Payload (ESP)", RFC 1827, August 1995

[22] Hanks, S., Li, T., Farinacci, D., Traina, P., "Generic Routing Encapsulation (GRE)", RFC 1701, October 1994

[23] Simpson, W., "IP in IP Tunneling", RFC 1853, October 1995

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

[25] L. J. Blunk, J. R. Vollbrecht, "PPP Extensible Authentication Protocol (EAP)." RFC 2284, March 1998.

[26] G. Zorn, P. R. Calhoun, "Limiting Fraud in Roaming", draft-ietf-roamops-fraud-limit-00.txt, IETF work in progress, May 1999.

[27] Narten, Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998

[28] P. Calhoun, A. Rubens, H. Akhtar, E. Guttman, W. Bulley, J. Haag, "Diameter Implementation Guidelines", draft-calhoun-diameter-impl-guide-04.txt, IETF work in progress, June 2000.

[29] J. Arkko, P. Calhoun, P. Patel, G. Zorn, "Diameter Accounting Extension", draft-calhoun-diameter-accounting-09.txt, IETF work in progress, January 2001.

[30] P. Calhoun, C. Perkins, "Diameter Mobile IP Extensions", draft-calhoun-diameter-mobileip-12.txt, IETF work in progress, January 2001.

[31] P. Calhoun, "Diameter Resource Management", draft-calhoun-diameter-res-mgmt-06.txt, IETF Work in Progress, January 2001.

[32] C. Rigney, W. Willats, P. Calhoun, "RADIUS Extensions", RFC 2869, June 2000.

[33] G. Zorn, D. Leifer, A. Rubens, J. Shriver, M. Holdrege, I. Goyret, "RADIUS Attributes for Tunnel Protocol Support", RFC 2868, June 2000.

[24] G. Zorn, B. Aboba, D. Mitton, "RADIUS Accounting Modifications for Tunnel Protocol Support", RFC 2867, June 2000.

13.0 Acknowledgements

The authors would like to thank Carl Rigney, Allan C. Rubens, William Allen Simpson, and Steve Willens for their work on the original RADIUS, from which much of the concepts in this specification were derived from. Also Carl Rigney and Ward Willats for [32], and Glen Zorn, Dory Leifer, Allan C. Rubens, John Shriver, Matt Holdrege and Ignacio Goyret for their work on [33]. This document stole text and concepts from both [32] and [33].

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

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

14.0 Authors' Addresses

Questions about this memo can be directed to:

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

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

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

Phone: +1 734-764-9993
Fax: +1 734-647-3185
E-mail: web@merit.edu

Allan C. Rubens
Tut Systems, Inc.
220 E. Huron, Suite 260
Ann Arbor, MI 48104
USA

Phone: +1 734-995-1697
E-Mail: arubens@tutsys.com

Jeff Haag
Cisco Systems
7025 Kit Creek Road
PO Box 14987
Research Triangle Park, NC 27709

Phone: 1-919-392-2353
E-Mail: haag@cisco.com

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

USA

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

15.0 Full Copyright Statement

16.0 Expiration Date

This memo is filed as <draft-calhoun-diameter-nasreq-06.txt> and expires in July 2001.