Network Working Group
Request for Comments: 2869
Category: Informational

C. Rigney
Livingston
W. Willats
Cyno Technologies
P. Calhoun
Sun Microsystems

June 2000

RADIUS Extensions

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document describes additional attributes for carrying authentication, authorization and accounting information between a Network Access Server (NAS) and a shared Accounting Server using the Remote Authentication Dial In User Service (RADIUS) protocol described in RFC 2865 [1] and RFC 2866 [2].

Table of Contents

1. Introduction

RFC 2865 [1] describes the RADIUS Protocol as it is implemented and deployed today, and RFC 2866 [2] describes how Accounting can be performed with RADIUS.

This memo suggests several additional Attributes that can be added to RADIUS to perform various useful functions. These Attributes do not have extensive field experience yet and should therefore be considered experimental.

The Extensible Authentication Protocol (EAP) [3] is a PPP extension that provides support for additional authentication methods within PPP. This memo describes how the EAP-Message and Message-Authenticator attributes may be used for providing EAP support within RADIUS.

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

1.1. Specification of Requirements

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [4].

An implementation is not compliant if it fails to satisfy one or more of the must or must not requirements for the protocols it implements. An implementation that satisfies all the must, must not, should and should not requirements for its protocols is said to be "unconditionally compliant"; one that satisfies all the must and must not requirements but not all the should or should not requirements for its protocols is said to be "conditionally compliant."

A NAS that does not implement a given service MUST NOT implement the RADIUS attributes for that service. For example, a NAS that is unable to offer ARAP service MUST NOT implement the RADIUS attributes for ARAP. A NAS MUST treat a RADIUS access-request requesting an unavailable service as an access-reject instead.

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.

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

Operation is identical to that defined in RFC 2865 [1] and RFC 2866 [2].

2.1. RADIUS support for Interim Accounting Updates

When a user is authenticated, a RADIUS server issues an Access-Accept in response to a successful Access-Request. If the server wishes to receive interim accounting messages for the given user it must include the Acct-Interim-Interval RADIUS attribute in the message, which indicates the interval in seconds between interim messages.

It is also possible to statically configure an interim value on the NAS itself. Note that a locally configured value on the NAS MUST override the value found in an Access-Accept.

This scheme does not break backward interoperability since a RADIUS server not supporting this extension will simply not add the new Attribute. NASes not supporting this extension will ignore the Attribute.

Note that all information in an interim message is cumulative (i.e. number of packets sent is the total since the beginning of the session, not since the last interim message).

It is envisioned that an Interim Accounting record (with Acct-Status-Type = Interim-Update (3)) would contain all of the attributes normally found in an Accounting Stop message with the exception of the Acct-Term-Cause attribute.

Since all the information is cumulative, a NAS MUST ensure that only a single generation of an interim Accounting message for a given session is present in the retransmission queue at any given time.

A NAS MAY use a fudge factor to add a random delay between Interim Accounting messages for separate sessions. This will ensure that a cycle where all messages are sent at once is prevented, such as might otherwise occur if a primary link was recently restored and many dial-up users were directed to the same NAS at once.

The Network and NAS CPU load of using Interim Updates should be carefully considered, and appropriate values of Acct-Interim-Interval chosen.

2.2. RADIUS support for Apple Remote Access Protocol

The RADIUS (Remote Authentication Dial-In User Service) protocol provides a method that allows multiple dial-in Network Access Server (NAS) devices to share a common authentication database.

The Apple Remote Access Protocol (ARAP) provides a method for sending AppleTalk network traffic over point-to-point links, typically, but not exclusively, asynchronous and ISDN switched-circuit connections. Though Apple is moving toward ATCP on PPP for future remote access services, ARAP is still a common way for the installed base of Macintosh users to make remote network connections, and is likely to remain so for some time.

ARAP is supported by several NAS vendors who also support PPP, IPX and other protocols in the same NAS. ARAP connections in these multi-protocol devices are often not authenticated with RADIUS, or if they are, each vendor creates an individual solution to the problem.

This section describes the use of additional RADIUS attributes to support ARAP. RADIUS client and server implementations that implement this specification should be able to authenticate ARAP connections in an interoperable manner.

This section assumes prior knowledge of RADIUS, and will go into some detail on the operation of ARAP before entering a detailed discussion of the proposed ARAP RADIUS attributes.

There are two features of ARAP this document does not address:

1. User initiated password changing. This is not part of RADIUS, but can be implemented through a software process other than RADIUS.

2. Out-of-Band messages. At any time, the NAS can send messages to an ARA client which appear in a dialog box on the dial-in user’s screen. These are not part of authentication and do not belong here. However, we note that a Reply-Message attribute in

an Access-Accept may be sent down to the user as a sign-on message of the day string using the out-of-band channel.

We have tried to respect the spirit of the existing RADIUS protocol as much as possible, making design decisions compatible with prior art. Further, we have tried to strike a balance between flooding the RADIUS world with new attributes, and hiding all of ARAP operation within a single multiplexed ARAP attribute string or within Extended Authentication Protocol (EAP) [3] machinery.

However, we feel ARAP is enough of a departure from PPP to warrant a small set of similarly named attributes of its own.

We have assumed that an ARAP-aware RADIUS server will be able to do DES encryption and generate security module challenges. This is in keeping with the general RADIUS goal of smart server / simple NAS.

ARAP authenticates a connection in two phases. The first is a "Two-Way DES" random number exchange, using the user’s password as a key. We say "Two-Way" because the ARAP NAS challenges the dial-in client to authenticate itself, and the dial-in client challenges the ARAP NAS to authenticate itself.

Specifically, ARAP does the following:

1. The NAS sends two 32-bit random numbers to the dial-in client in an ARAP msg_auth_challenge packet.

2. The dial-in client uses the user’s password to DES encrypt the two random numbers sent to it by the NAS. The dial-in client then sends this result, the user’s name and two 32-bit random numbers of its own back to the NAS in an ARAP msg_auth_request packet.

3. The NAS verifies the encrypted random numbers sent by the dial-in client are what it expected. If so, it encrypts the dial-in client’s challenge using the password and sends it back to the dial-in client in an ARAP msg_auth_response packet.

Note that if the dial-in client’s response was wrong, meaning the user has the wrong password, the server can initiate a retry sequence up to the maximum amount of retries allowed by the NAS. In this case, when the dial-in client receives the ARAP msg_auth_response packet it will acknowledge it with an ARAP msg_auth_again packet.

After this first "DES Phase" the ARAP NAS MAY initiate a secondary authentication phase using what Apple calls "Add-In Security Modules." Security Modules are small pieces of code which run on both the client and server and are allowed to read and write arbitrary data across the communications link to perform additional authentication functions. Various security token vendors use this mechanism to authenticate ARA callers.

Although ARAP allows security modules to read and write anything they like, all existing security modules use simple challenge and response cycles, with perhaps some overall control information. This document assumes all existing security modules can be supported with one or more challenge/response cycles.

To complicate RADIUS and ARAP integration, ARAP sends down some profile information after the DES Phase and before the Security Module phase. This means that besides the responses to challenges, this profile information must also be present, at somewhat unusual times. Fortunately the information is only a few pieces of numeric data related to passwords, which this document packs into a single new attribute.

Presenting an Access-Request to RADIUS on behalf of an ARAP connection is straightforward. The ARAP NAS generates the random number challenge, and then receives the dial-in client’s response, the dial-in client’s challenge, and the user’s name. Assuming the user is not a guest, the following information is forwarded in an Access-Request packet: User-Name (up to 31 characters long), Framed-Protocol (set to 3, ARAP), ARAP-Password, and any additional attributes desired, such as Service-Type, NAS-IP-Address, NAS-Id, NAS-Port-Type, NAS-Port, NAS-Port-Id, Connect-Info, etc.

The Request Authenticator is a NAS-generated 16 octet random number. The low-order 8 octets of this number are sent to the dial-in user as the two 4 octet random numbers required in the ARAP msg_auth_challenge packet. Octets 0-3 are the first random number and Octets 4-7 are the second random number.

The ARAP-Password in the Access-Request contains a 16 octet random number field, and is used to carry the dial-in user’s response to the NAS challenge and the client’s own challenge to the NAS. The high-order octets contain the dial-in user’s challenge to the NAS (2 32-bit numbers, 8 octets) and the low-order octets contain the dial-in user’s response to the NAS challenge (2 32-bit numbers, 8 octets).

Only one of User-Password, CHAP-Password, or ARAP-Password needs to be present in an Access-Request, or one or more EAP-Messages.

If the RADIUS server does not support ARAP it SHOULD return an Access-Reject to the NAS.

If the RADIUS server does support ARAP, it should verify the user’s response using the Challenge (from the lower order 8 octets of the Request Authenticator) and the user’s response (from the low order 8 octets of the ARAP-Password).

If that authentication fails, the RADIUS server should return an Access-Reject packet to the NAS, with optional Password-Retry and Reply-Messages attributes. The presence of Password-Retry indicates the ARAP NAS MAY choose to initiate another challenge-response cycle, up to a total number of times equal to the integer value of the Password-Retry attribute.

If the user is authenticated, the RADIUS server should return an Access-Accept packet (Code 2) to the NAS, with ID and Response Authenticator as usual, and attributes as follows:

Service-Type of Framed-Protocol.

Framed-Protocol of ARAP (3).

Session-Timeout with the maximum connect time for the user in seconds. If the user is to be given unlimited time, Session-Timeout should not be included in the Access-Accept packet, and ARAP will treat that as an unlimited timeout (-1).

ARAP-Challenge-Response, containing 8 octets with the 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 attribute 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. If the user’s password is greater than 8 octets in length, an Access-Reject MUST be sent instead.

ARAP-Features, containing information that the NAS should send to the user in an ARAP "feature flags" packet.

Octet 0: If zero, user cannot change their password. If non-zero user can. (RADIUS does not handle the password changing, just the attribute which indicates whether ARAP indicates they can.)

Octet 1: Minimum acceptable password length (0-8).

Octet 2-5: Password creation date in Macintosh format, defined as 32 bits unsigned representing seconds since Midnight GMT January 1, 1904.

Octet 6-9 Password Expiration Delta from create date in seconds.

Octet 10-13: Current RADIUS time in Macintosh format

Optionally, a single Reply-Message with a text string up to 253 characters long which MAY be sent down to the user to be displayed in a sign-on/message of the day dialog.

Framed-AppleTalk-Network may be included.

Framed-AppleTalk-Zone, up to 32 characters in length, may be included.

ARAP defines the notion of a list of zones for a user. Along with a list of zone names, a Zone Access Flag is defined (and used by the NAS) which says how to use the list of zone names. That is, the dial-in user may only be allowed to see the Default Zone, or only the zones in the zone list (inclusive) or any zone except those in the zone list (exclusive).

The ARAP NAS handles this by having a named filter which contains (at least) zone names. This solves the problem where a single RADIUS server is managing disparate NAS clients who may not be able to "see" all of the zone names in a user zone list. Zone names only have meaning "at the NAS." The disadvantage of this approach is that zone filters must be set up on the NAS somehow, then referenced by the RADIUS Filter-Id.

ARAP-Zone-Access contains an integer which specifies how the "zone list" for this user should be used. 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.

The inclusion of a Callback-Number or Callback-Id attribute in the Access-Accept MAY cause the ARAP NAS to disconnect after sending the Feature Flags to begin callback processing in an ARAP specific way.

Other attributes may be present in the Access-Accept packet as well.

An ARAP NAS will need other information to finish bringing up the connection to the dial in client, but this information can be provided by the ARAP NAS without any help from RADIUS, either through configuration by SNMP, a NAS administration program, or deduced by the AppleTalk stack in the NAS. Specifically:

1. AppearAsNet and AppearAsNode values, sent to the client to tell it what network and node numbers it should use in its datagram packets. AppearAsNet can be taken from the Framed-AppleTalk-Network attribute or from the configuration or AppleTalk stack onthe NAS.

2. The "default" zone - that is the name of the AppleTalk zone in which the dial-in client will appear. (Or can be specified with the Framed-AppleTalk-Zone attribute.)

3. Other very NAS specific stuff such as the name of the NAS, and smartbuffering information. (Smartbuffering is an ARAP mechanism for replacing common AppleTalk datagrams with small tokens, to improve slow link performance in a few common traffic situations.)

4. "Zone List" information for this user. The ARAP specification defines a "zone count" field which is actually unused.

RADIUS supports ARAP Security Modules in the following manner.

After DES authentication has been completed, the RADIUS server may instruct the ARAP NAS to run one or more security modules for the dial-in user. Although the underlying protocol supports executing multiple security modules in series, in practice all current implementations only allow executing one. Through the use of multiple Access-Challenge requests, multiple modules can be supported, but this facility will probably never be used.

We also assume that, even though ARAP allows a free-form dialog between security modules on each end of the point-to-point link, in actual practice all security modules can be reduced to a simple challenge/response cycle.

If the RADIUS server wishes to instruct the ARAP NAS to run a security module, it should send an Access-Challenge packet to the NAS with (optionally) the State attribute, plus the ARAP-Challenge-Response, ARAP-Features, and two more attributes:

ARAP-Security: a four octet security module signature, containing a Macintosh OSType.

ARAP-Security-Data, a string to carry the actual security module challenge and response.

When the security module finishes executing, the security module response is passed in an ARAP-Security-Data attribute from the NAS to the RADIUS server in a second Access-Request, also including the State from the Access-Challenge. The authenticator field contains no special information in this case, and this can be discerned by the presence of the State attribute.

2.3. RADIUS Support for Extensible Authentication Protocol (EAP)

The Extensible Authentication Protocol (EAP), described in [3], 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 cards, Kerberos, Public Key, One Time Passwords, and others. In order to provide for support of EAP within RADIUS, two new attributes, EAP-Message and Message-Authenticator, are introduced in this document. This section describes how these new attributes may be used for providing EAP support within RADIUS.

In the proposed scheme, the RADIUS server is used to shuttle RADIUS-encapsulated EAP Packets between the NAS and a backend security server. While the conversation between the RADIUS server and the backend security server will typically occur using a proprietary protocol developed by the backend security server vendor, it is also possible to use RADIUS-encapsulated EAP via the EAP-Message attribute. This has the advantage of allowing the RADIUS server to support EAP without the need for authentication-specific code, which can instead reside on the backend security server.

2.3.1. Protocol Overview

The EAP conversation between the authenticating peer (dial-in user) and the NAS begins with the negotiation of EAP within LCP. Once EAP has been negotiated, the NAS MUST send an EAP-Request/Identity message to the authenticating peer, unless identity is determined via some other means such as Called-Station-Id or Calling-Station-Id. The peer will then respond with an EAP-Response/Identity which the the NAS will then forward to the RADIUS server in the EAP-Message attribute of a RADIUS Access-Request packet. The RADIUS Server will typically use the EAP-Response/Identity to determine which EAP type is to be applied to the user.

In order to permit non-EAP aware RADIUS proxies to forward the Access-Request packet, if the NAS sends the EAP-Request/Identity, the NAS MUST copy the contents of the EAP-Response/Identity into the User-Name attribute and MUST include the EAP-Response/Identity in the User-Name attribute in every subsequent Access-Request. NAS-Port or NAS-Port-Id SHOULD be included in the attributes issued by the NAS in the Access-Request packet, and either NAS-Identifier or NAS-IP-Address MUST be included. In order to permit forwarding of the Access-Reply by EAP-unaware proxies, if a User-Name attribute was included in an Access-Request, the RADIUS Server MUST include the User-Name attribute in subsequent Access-Accept packets. Without the User-Name attribute, accounting and billing becomes very difficult to manage.

If identity is determined via another means such as Called-Station-Id or Calling-Station-Id, the NAS MUST include these identifying attributes in every Access-Request.

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 Called-Station-Id or Calling-Station-Id), and therefore an EAP-Request/Identity packet may not necessarily be issued by the NAS to the authenticating peer. In cases where an EAP-Request/Identity packet will not be sent, the NAS will send to the RADIUS server a RADIUS Access-Request packet containing an EAP-Message attribute signifying EAP-Start. EAP-Start is indicated by sending an EAP-Message attribute with a length of 2 (no data). However, it should be noted that since no User-Name attribute is included in the Access-Request, this approach is not compatible with RADIUS as specified in [1], nor can it easily be applied in situations where proxies are deployed, such as roaming or shared use networks.

If the RADIUS server supports EAP, it MUST respond with an Access-Challenge packet containing an EAP-Message attribute. If the RADIUS server does not support EAP, it MUST respond with an Access-Reject. The EAP-Message attribute includes an encapsulated EAP packet which is then passed on to the authenticating peer. In the case where the NAS does not initially send an EAP-Request/Identity message to the peer, the Access-Challenge typically will contain an EAP-Message attribute encapsulating an EAP-Request/Identity message, requesting the dial-in user to identify themself. The NAS will then respond with a RADIUS Access-Request packet containing an EAP-Message attribute encapsulating an EAP-Response. The conversation continues until either a RADIUS Access-Reject or Access-Accept packet is received.

Reception of a RADIUS Access-Reject packet, with or without an EAP-Message attribute encapsulating EAP-Failure, MUST result in the NAS issuing an LCP Terminate Request to the authenticating peer. A RADIUS Access-Accept packet with an EAP-Message attribute encapsulating EAP-Success successfully ends the authentication phase. The RADIUS Access-Accept/EAP-Message/EAP-Success packet MUST contain all of the expected attributes which are currently returned in an Access-Accept packet.

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.

For proxied RADIUS requests there are two methods of processing. If the domain is determined based on the Called-Station-Id, the RADIUS Server may proxy the initial RADIUS Access-Request/EAP-Start. If the domain is determined based on the user’s identity, the local RADIUS Server MUST respond with a RADIUS Access-Challenge/EAP-Identity packet. The response from the authenticating peer MUST be proxied to the final authentication server.

For proxied RADIUS requests, the NAS may receive an Access-Reject packet in response to its Access-Request/EAP-Identity packet. This would occur if the message was proxied to a RADIUS Server which does not support the EAP-Message extension. On receiving an Access-Reject, the NAS MUST send an LCP Terminate Request to the authenticating peer, and disconnect.

2.3.2. Retransmission

As noted in [3], the EAP authenticator (NAS) is responsible for retransmission of packets between the authenticating peer and the NAS. Thus if an EAP packet is lost in transit between the authenticating peer and the NAS (or vice versa), the NAS will retransmit. As in RADIUS [1], the RADIUS client is responsible for retransmission of packets between the RADIUS client and the RADIUS server.

Note that it may be necessary to adjust retransmission strategies and authentication timeouts in certain cases. For example, when a token card is used additional time may be required to allow the user to find the card and enter the token. Since the NAS will typically not have knowledge of the required parameters, these need to be provided by the RADIUS server. This can be accomplished by inclusion of Session-Timeout and Password-Retry attributes within the Access-Challenge packet.

If Session-Timeout is present in an Access-Challenge packet that also contains an EAP-Message, the value of the Session-Timeout provides the NAS with the maximum number of seconds the NAS should wait for an EAP-Response before retransmitting the EAP-Message to the dial-in user.

2.3.3. Fragmentation

Using the EAP-Message attribute, it is possible for the RADIUS server to encapsulate an EAP packet that is larger than the MTU on the link between the NAS and the peer. Since it is not possible for the RADIUS server to use MTU discovery to ascertain the link MTU, the Framed-MTU attribute may be included in an Access-Request packet containing an EAP-Message attribute so as to provide the RADIUS server with this information.

2.3.4. Examples

The example below shows the conversation between the authenticating peer, NAS, and RADIUS server, for the case of a One Time Password (OTP) authentication. OTP is used only for illustrative purposes; other authentication protocols could also have been used, although they might show somewhat different behavior.

```
Authenticating Peer     NAS                    RADIUS Server
-------------------     ---                    -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP ACK-EAP
auth ->
                        <- PPP EAP-Request/
                        Identity
PPP EAP-Response/
Identity (MyID) ->
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        (MyID) ->
                                                <- RADIUS
                                                Access-Challenge/
                                                EAP-Message/EAP-Request
                                                OTP/OTP Challenge
                        <- PPP EAP-Request/
                        OTP/OTP Challenge
PPP EAP-Response/
OTP, OTPpw ->
```

```
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        OTP, OTPpw ->
                                                 <- RADIUS
                                                 Access-Accept/
                                                 EAP-Message/EAP-Success
                                                 (other attributes)
                        <- PPP EAP-Success
PPP Authentication
Phase complete,
NCP Phase starts
```

In the case where the NAS first sends an EAP-Start packet to the RADIUS server, the conversation would appear as follows:

```
Authenticating Peer     NAS                    RADIUS Server
-------------------     ---                    -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP ACK-EAP
auth ->
                        RADIUS
                        Access-Request/
                       EAP-Message/Start ->
                                               <- RADIUS
                                               Access-Challenge/
                                               EAP-Message/Identity
                        <- PPP EA-Request/
                        Identity
PPP EAP-Response/
Identity (MyID) ->
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        (MyID) ->
                                                <- RADIUS
                                                Access-Challenge/
                                                EAP-Message/EAP-Request
                                                OTP/OTP Challenge
                        <- PPP EAP-Request/
                        OTP/OTP Challenge
PPP EAP-Response/
OTP, OTPpw ->
```

```
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        OTP, OTPpw ->
                                                 <- RADIUS
                                                 Access-Accept/
                                                 EAP-Message/EAP-Success
                                                 (other attributes)
                        <- PPP EAP-Success
PPP Authentication
Phase complete,
NCP Phase starts
```

In the case where the client fails EAP authentication, the conversation would appear as follows:

```
Authenticating Peer     NAS                    RADIUS Server
-------------------     ---                    -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP ACK-EAP
auth ->
                        Access-Request/
                        EAP-Message/Start ->
                                               <- RADIUS
                                               Access-Challenge/
                                               EAP-Message/Identity
                        <- PPP EAP-Request/
                        Identity
PPP EAP-Response/
Identity (MyID) ->
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        (MyID) ->
                                                <- RADIUS
                                                Access-Challenge/
                                                EAP-Message/EAP-Request
                                                OTP/OTP Challenge
                        <- PPP EAP-Request/
                        OTP/OTP Challenge
PPP EAP-Response/
OTP, OTPpw ->
                        RADIUS
                        Access-Request/
                        EAP-Message/
                        EAP-Response/
                        OTP, OTPpw ->
                                                 <- RADIUS
                                                 Access-Reject/
                                                 EAP-Message/EAP-Failure

                        <- PPP EAP-Failure
                        (client disconnected)
```

In the case that the RADIUS server or proxy does not support EAP-Message, the conversation would appear as follows:

```
Authenticating Peer     NAS                       RADIUS Server
-------------------     ---                       -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP ACK-EAP
auth ->
                        RADIUS
                        Access-Request/
                        EAP-Message/Start ->
                                                  <- RADIUS
                                                  Access-Reject
                        <- PPP LCP Terminate
                        (User Disconnected)
```

In the case where the local RADIUS Server does support EAP-Message, but the remote RADIUS Server does not, the conversation would appear as follows:

```
Authenticating Peer     NAS                       RADIUS Server
-------------------     ---                       -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP ACK-EAP
auth ->
                        RADIUS
                        Access-Request/
                        EAP-Message/Start ->
                                                  <- RADIUS
                                                  Access-Challenge/
                                                  EAP-Message/Identity
                        <- PPP EAP-Request/
                        Identity
```

```
PPP EAP-Response/
Identity
(MyID) ->
                        RADIUS
                        Access-Request/
                        EAP-Message/EAP-Response/
                        (MyID) ->
                                                  <- RADIUS
                                                  Access-Reject
                                                  (proxied from remote
                                                   RADIUS Server)
                        <- PPP LCP Terminate
                        (User Disconnected)
```

In the case where the authenticating peer does not support EAP, but where EAP is required for that user, the conversation would appear as follows:

```
Authenticating Peer     NAS                       RADIUS Server
-------------------     ---                       -------------

                        <- PPP LCP Request-EAP
                        auth
PPP LCP NAK-EAP
auth ->
                        <- PPP LCP Request-CHAP
                        auth
PPP LCP ACK-CHAP
auth ->
                        <- PPP CHAP Challenge
PPP CHAP Response ->
                        RADIUS
                        Access-Request/
                        User-Name,
                        CHAP-Password ->
                                                  <- RADIUS
                                                  Access-Reject
                        <-  PPP LCP Terminate
                        (User Disconnected)
```

In the case where the NAS does not support EAP, but where EAP is required for that user, the conversation would appear as follows:

```
Authenticating Peer     NAS                       RADIUS Server
-------------------     ---                       -------------

                        <- PPP LCP Request-CHAP
                        auth
```

```
PP LCP ACK-CHAP
auth ->
                        <- PPP CHAP Challenge
PPP CHAP Response ->
                        RADIUS
                        Access-Request/
                        User-Name,
                        CHAP-Password ->

                                                 <- RADIUS
                                                 Access-Reject
                        <-  PPP LCP Terminate
                        (User Disconnected)
```

2.3.5. Alternative uses

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

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

In the case where RADIUS-encapsulated EAP is used in a conversation between a RADIUS server and a backend security server, the security server will typically return an Access-Accept/EAP-Success message without inclusion of the expected attributes currently returned in an Access-Accept. This means that the RADIUS server MUST add these attributes prior to sending an Access-Accept/EAP-Success message to the NAS.

3. Packet Format

Packet Format is identical to that defined in RFC 2865 [1] and 2866 [2].

4. Packet Types

Packet types are identical to those defined in RFC 2865 [1] and 2866 [2].

See "Table of Attributes" below to determine which types of packets can contain which attributes defined here.

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 order of attributes of the same type SHOULD be preserved. The order of attributes of different types is not required to be preserved.

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

A summary of the attribute format is the same as in RFC 2865 [1] but is included here for ease of reference. The fields are transmitted from left to right.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |  Value ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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 [5]. 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 RFC 2865 [1], "RADIUS")
40-51	(refer to RFC 2866 [2], "RADIUS Accounting")
52	Acct-Input-Gigawords
53	Acct-Output-Gigawords
54	Unused
55	Event-Timestamp
56-59	Unused
60-63	(refer to RFC 2865 [1], "RADIUS")
64-67	(refer to [6])
68	(refer to [7])
69	(refer to [6])
70	ARAP-Password
71	ARAP-Features
72	ARAP-Zone-Access

73	ARAP-Security
74	ARAP-Security-Data
75	Password-Retry
76	Prompt
77	Connect-Info
78	Configuration-Token
79	EAP-Message
80	Message-Authenticator
81-83	(refer to [6])
84	ARAP-Challenge-Response
85	Acct-Interim-Interval
86	(refer to [7])
87	NAS-Port-Id
88	Framed-Pool
89	Unused
90-91	(refer to [6])
92-191	Unused

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 a packet 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.

Note that none of the types in RADIUS terminate with a NUL (hex 00). In particular, types "text" and "string" in RADIUS do not terminate with a NUL (hex 00). The Attribute has a length field and does not use a terminator. Text contains UTF-8 encoded 10646 [8] characters and String contains 8-bit binary data. Servers and servers and clients MUST be able to deal with embedded nulls. RADIUS implementers using C are cautioned not to use strcpy() when handling strings.

The format of the value field is one of five data types. Note that type "text" is a subset of type "string."

text   1-253 octets containing UTF-8 encoded 10646 [8] characters. Text of length zero (0) MUST NOT be sent; omit the entire attribute instead.

string 1-253 octets containing binary data (values 0 through 255 decimal, inclusive). Strings of length zero (0) MUST NOT be sent; omit the entire attribute instead.

address 32 bit unsigned value, most significant octet first.

integer 32 bit unsigned value, most significant octet first.

time 32 bit unsigned value, most significant octet first -- seconds since 00:00:00 UTC, January 1, 1970.

5.1. Acct-Input-Gigawords

Description

This attribute indicates how many times the Acct-Input-Octets counter has wrapped around 2^32 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 or Interim-Update.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

52 for Acct-Input-Gigawords.

Length

6

Value

The Value field is four octets.

5.2. Acct-Output-Gigawords

Description

This attribute indicates how many times the Acct-Output-Octets counter has wrapped around 2^32 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 or Interim-Update.

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

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |             Value
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
              Value (cont)         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

53 for Acct-Output-Gigawords.

Length

6

Value

The Value field is four octets.

5.3. Event-Timestamp

Description

This attribute is included in an Accounting-Request packet to record the time that this event occurred on the NAS, in seconds since January 1, 1970 00:00 UTC.

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

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |    Length     |             Value
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
               Value (cont)         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

55 for Event-Timestamp

Length

6

Value

The Value field is four octets encoding an unsigned integer with the number of seconds since January 1, 1970 00:00 UTC.

5.4. ARAP-Password

Description

This attribute is only present in an Access-Request packet containing a Framed-Protocol of ARAP.

Only one of User-Password, CHAP-Password, or ARAP-Password needs to be present in an Access-Request, or one or more EAP-Messages.

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

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |    Length     |             Value1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                    |             Value2
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                    |             Value3
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                    |             Value4
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

70 for ARAP-Password.

Length

18

Value

This attribute contains a 16 octet string, used to carry the dial-in user’s response to the NAS challenge and the client’s own challenge to the NAS. The high-order octets (Value1 and Value2) contain the dial-in user’s challenge to the NAS (2 32-bit numbers, 8 octets) and the low-order octets (Value3 and Value4) contain the dial-in user’s response to the NAS challenge (2 32-bit numbers, 8 octets).

5.5. ARAP-Features

Description

This attribute is sent in an Access-Accept packet with Framed-Protocol of ARAP, and includes password information that the NAS should sent to the user in an ARAP "feature flags" packet.

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

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |     Value1    |    Value2     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Value3                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Value4                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Value5                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

71 for ARAP-Features.

Length

16

Value

The Value field is a compound string containing information the NAS should send to the user in the ARAP "feature flags" packet.

Value1: If zero, user cannot change their password. If non-zero user can. (RADIUS does not handle the password changing, just the attribute which indicates whether ARAP indicates they can.)

Value2: Minimum acceptable password length, from 0 to 8.

Value3: Password creation date in Macintosh format, defined as 32 unsigned bits representing seconds since Midnight GMT January 1, 1904.

Value4: Password Expiration Delta from create date in seconds.

Value5: Current RADIUS time in Macintosh format.

5.6. ARAP-Zone-Access

Description

This attribute is included in an Access-Accept packet with Framed-Protocol of ARAP to indicate how the ARAP zone list for the user should be used.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

72 for ARAP-Zone-Access.

Length

6

Value

The Value field is four octets encoding an integer with one of the following values:

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.

5.7. ARAP-Security

Description

This attribute identifies the ARAP Security Module to be used in an Access-Challenge packet.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

73 for ARAP-Security.

Length

6

Value

The Value field is four octets, containing an integer specifying the security module signature, which is a Macintosh OSType. (Macintosh OSTypes are 4 ascii characters cast as a 32-bit integer)

5.8. ARAP-Security-Data

Description

This attribute contains the actual security module challenge or response, and can be found in Access-Challenge and Access-Request packets.

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

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |     String...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

74 for ARAP-Security-Data.

Length

>=3

String

The String field contains the security module challenge or response associated with the ARAP Security Module specified in ARAP-Security.

5.9. Password-Retry

Description

This attribute MAY be included in an Access-Reject to indicate how many authentication attempts a user may be allowed to attempt before being disconnected.

It is primarily intended for use with ARAP authentication.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

75 for Password-Retry.

Length

6

Value

The Value field is four octets, containing an integer specifying the number of password retry attempts to permit the user.

5.10. Prompt

Description

This attribute is used only in Access-Challenge packets, and indicates to the NAS whether it should echo the user’s response as it is entered, or not echo it.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

76 for Prompt.

Length

6

Value

The Value field is four octets.

```
 0      No Echo
 1      Echo
```

5.11. Connect-Info

Description

This attribute is sent from the NAS to indicate the nature of the user’s connection.

The NAS MAY send this attribute in an Access-Request or Accounting-Request to indicate the nature of the user’s connection.

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

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |     Text...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

77 for Connect-Info.

Length

>= 3

Text

The Text field consists of UTF-8 encoded 10646 [8] characters. The connection speed SHOULD be included at the beginning of the first Connect-Info attribute in the packet. If the transmit and receive connection speeds differ, they may both be included in the first attribute with the transmit speed first (the speed the NAS modem transmits at), a slash (/), the receive speed, then optionally other information.

For example, "28800 V42BIS/LAPM" or "52000/31200 V90"

More than one Connect-Info attribute may be present in an Accounting-Request packet to accommodate expected efforts by ITU to have modems report more connection information in a standard format that might exceed 252 octets.

5.12. Configuration-Token

Description

This attribute is for use in large distributed authentication networks based on proxy. It is sent from a RADIUS Proxy Server to a RADIUS Proxy Client in an Access-Accept to indicate a type of user profile to be used. It should not be sent to a NAS.

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

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |  String ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

78 for Configuration-Token.

Length

>= 3

String

The String field is one or more octets. The actual format of the information is site or application specific, and a robust implementation SHOULD support the field as undistinguished octets.

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

5.13. EAP-Message

Description

This attribute encapsulates Extended Access Protocol [3] packets so as to allow the NAS to authenticate dial-in users via EAP without having to understand the EAP protocol.

The NAS places any EAP messages received from the user into one or more EAP attributes and forwards them to the RADIUS Server as part of the Access-Request, which can return EAP messages in Access-Challenge, Access-Accept and Access-Reject packets.

A RADIUS Server receiving EAP messages that it does not understand SHOULD return an Access-Reject.

The NAS places EAP messages received from the authenticating peer into one or more EAP-Message attributes and forwards them to the RADIUS Server within an Access-Request message. If multiple EAP-Messages are contained within an Access-Request or Access-Challenge packet, they MUST be in order and they MUST be consecutive attributes in the Access-Request or Access-Challenge packet. Access-Accept and Access-Reject packets SHOULD only have ONE EAP-Message attribute in them, containing EAP-Success or EAP-Failure.

It is expected that EAP will be used to implement a variety of authentication methods, including methods involving strong cryptography. In order to prevent attackers from subverting EAP by attacking RADIUS/EAP, (for example, by modifying the EAP-Success or EAP-Failure packets) it is necessary that RADIUS/EAP provide integrity protection at least as strong as those used in the EAP methods themselves.

Therefore the Message-Authenticator attribute MUST be used to protect all Access-Request, Access-Challenge, Access-Accept, and Access-Reject packets containing an EAP-Message attribute.

Access-Request packets including an EAP-Message attribute without a Message-Authenticator attribute SHOULD be silently discarded by the RADIUS server. A RADIUS Server supporting EAP-Message MUST calculate the correct value of the Message-Authenticator and silently discard the packet if it does not match the value sent. A RADIUS Server not supporting EAP-Message MUST return an Access-Reject if it receives an Access-Request containing an EAP-Message attribute. A RADIUS Server receiving an EAP-Message attribute that it does not understand MUST return an Access-Reject.

Access-Challenge, Access-Accept, or Access-Reject packets including an EAP-Message attribute without a Message-Authenticator attribute SHOULD be silently discarded by the NAS. A NAS supporting EAP-Message MUST calculate the correct value of the Message-Authenticator and silently discard the packet if it does not match the value sent.

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

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |     String...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

79 for EAP-Message.

Length

>= 3

String

The String field contains EAP packets, as defined in [3]. If multiple EAP-Message attributes are present in a packet their values should be concatenated; this allows EAP packets longer than 253 octets to be passed by RADIUS.

5.14. Message-Authenticator

Description

This attribute MAY be used to sign Access-Requests to prevent spoofing Access-Requests using CHAP, ARAP or EAP authentication methods. It MAY be used in any Access-Request. It MUST be used in any Access-Request, Access-Accept, Access-Reject or Access-Challenge that includes an EAP-Message attribute.

A RADIUS Server receiving an Access-Request with a Message-Authenticator Attribute present MUST calculate the correct value of the Message-Authenticator and silently discard the packet if it does not match the value sent.

A RADIUS Client receiving an Access-Accept, Access-Reject or Access-Challenge with a Message-Authenticator Attribute present MUST calculate the correct value of the Message-Authenticator and silently discard the packet if it does not match the value sent.

Earlier drafts of this memo used "Signature" as the name of this attribute, but Message-Authenticator is more precise. Its operation has not changed, just the name.

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

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |     String...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

80 for Message-Authenticator

Length

18

String

When present in an Access-Request packet, Message-Authenticator is an HMAC-MD5 [9] checksum of the entire Access-Request packet, including Type, ID, Length and authenticator, using the shared secret as the key, as follows.

Message-Authenticator = HMAC-MD5 (Type, Identifier, Length, Request Authenticator, Attributes)

When the checksum is calculated the signature string should be considered to be sixteen octets of zero.

For Access-Challenge, Access-Accept, and Access-Reject packets, the Message-Authenticator is calculated as follows, using the Request-Authenticator from the Access-Request this packet is in reply to:

Message-Authenticator = HMAC-MD5 (Type, Identifier, Length, Request Authenticator, Attributes)

When the checksum is calculated the signature string should be considered to be sixteen octets of zero. The shared secret is used as the key for the HMAC-MD5 hash. The is calculated and inserted in the packet before the Response Authenticator is calculated.

This attribute is not needed if the User-Password attribute is present, but is useful for preventing attacks on other types of authentication. This attribute is intended to thwart attempts by an attacker to setup a "rogue" NAS, and perform online dictionary attacks against the RADIUS server. It does not afford protection against "offline" attacks where the attacker intercepts packets containing (for example) CHAP challenge and response, and performs a dictionary attack against those packets offline.

IP Security will eventually make this attribute unnecessary, so it should be considered an interim measure.

5.15. ARAP-Challenge-Response

Description

This attribute is sent in an Access-Accept packet with Framed-Protocol of ARAP, and contains the response to the dial-in client’s challenge.

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

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |     Value...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

84 for ARAP-Challenge-Response.

Length

10

Value

The Value field 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 attribute 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.

5.16. Acct-Interim-Interval

Description

This attribute indicates the number of seconds between each interim update in seconds for this specific session. This value can only appear in the Access-Accept message.

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

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |             Value
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Value (cont)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

85 for Acct-Interim-Interval.

Length

6

Value

The Value field contains the number of seconds between each interim update to be sent from the NAS for this session. The value MUST NOT be smaller than 60. The value SHOULD NOT be smaller than 600, and careful consideration should be given to its impact on network traffic.

5.17. NAS-Port-Id

Description

This Attribute contains a text string which identifies the port of the NAS which is authenticating the user. It is only used in Access-Request and Accounting-Request packets. 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-Id SHOULD be present in an Access-Request packet, if the NAS differentiates among its ports. NAS-Port-Id is intended for use by NASes which cannot conveniently number their ports.

A summary of the NAS-Port-Id Attribute format is shown below. The fields are transmitted from left to right.

```
    0                   1                   2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |    Length     |     Text...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

87 for NAS-Port-Id.

Length

>= 3

Text

The Text field contains the name of the port using UTF-8 encoded 10646 [8] characters.

5.18. Framed-Pool

Description

This Attribute contains the name of an assigned address pool that SHOULD be used to assign an address for the user. If a NAS does not support multiple address pools, the NAS should ignore this Attribute. Address pools are usually used for IP addresses, but can be used for other protocols if the NAS supports pools for those protocols.

A summary of the Framed-Pool Attribute format is shown below. The fields are transmitted from left to right.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |     String...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

88 for Framed-Pool

Length

>= 3

String

The string field contains the name of an assigned address pool configured on the NAS.

5.19. Table of Attributes

The following table provides a guide to which attributes may be found in which kind of packets. Acct-Input-Gigawords, Acct-Output-Gigawords, Event-Timestamp, and NAS-Port-Id may have 0-1 instances in an Accounting-Request packet. Connect-Info may have 0+ instances in an Accounting-Request packet. The other attributes added in this document must not be present in an Accounting-Request.

Request	Accept	Reject	Challenge	#	Attribute
0-1	0	0	0	70	ARAP-Password [Note 1]
0	0-1	0	0-1	71	ARAP-Features
0	0-1	0	0	72	ARAP-Zone-Access
0-1	0	0	0-1	73	ARAP-Security
0+	0	0	0+	74	ARAP-Security-Data
0	0	0-1	0	75	Password-Retry
0	0	0	0-1	76	Prompt
0-1	0	0	0	77	Connect-Info
0	0+	0	0	78	Configuration-Token
0+	0+	0+	0+	79	EAP-Message [Note 1]
0-1	0-1	0-1	0-1	80	Message-Authenticator [Note 1]
0	0-1	0	0-1	84	ARAP-Challenge-Response
0	0-1	0	0	85	Acct-Interim-Interval
0-1	0	0	0	87	NAS-Port-Id
0	0-1	0	0	88	Framed-Pool
Request	Accept	Reject	Challenge	#	Attribute

[Note 1] An Access-Request that contains either a User-Password or CHAP-Password or ARAP-Password or one or more EAP-Message attributes MUST NOT contain more than one type of those four attributes. If it does not contain any of those four attributes, it SHOULD contain a Message-Authenticator. If any packet type contains an EAP-Message attribute it MUST also contain a Message-Authenticator.

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.

6. IANA Considerations

The Packet Type Codes, Attribute Types, and Attribute Values defined in this document are registered by the Internet Assigned Numbers Authority (IANA) from the RADIUS name spaces as described in the "IANA Considerations" section of [1], in accordance with BCP 26 [10].

7. Security Considerations

The attributes other than Message-Authenticator and EAP-Message in this document have no additional security considerations beyond those already identified in [1].

7.1. Message-Authenticator Security

Access-Request packets with a User-Password establish the identity of both the user and the NAS sending the Access-Request, because of the way the shared secret between NAS and RADIUS server is used. Access-Request packets with CHAP-Password or EAP-Message do not have a User-Password attribute, so the Message-Authenticator attribute should be used in access-request packets that do not have a User-Password, in order to establish the identity of the NAS sending the request.

7.2. EAP Security

Since the purpose of EAP is to provide enhanced security for PPP authentication, it is critical that RADIUS support for EAP be secure. In particular, the following issues must be addressed:

Separation of EAP server and PPP authenticator
Connection hijacking
Man in the middle attacks
Multiple databases

Negotiation attacks

7.2.1. Separation of EAP server and PPP authenticator

It is possible for the EAP endpoints to mutually authenticate, negotiate a ciphersuite, and derive a session key for subsequent use in PPP encryption.

This does not present an issue on the peer, since the peer and EAP client reside on the same machine; all that is required is for the EAP client module to pass the session key to the PPP encryption module.

The situation is more complex when EAP is used with RADIUS, since the PPP authenticator will typically not reside on the same machine as the EAP server. For example, the EAP server may be a backend security server, or a module residing on the RADIUS server.

In the case where the EAP server and PPP authenticator reside on different machines, there are several implications for security. Firstly, mutual authentication will occur between the peer and the EAP server, not between the peer and the authenticator. This means that it is not possible for the peer to validate the identity of the NAS or tunnel server that it is speaking to.

As described earlier, when EAP/RADIUS is used to encapsulate EAP packets, the Message-Authenticator attribute is required in EAP/RADIUS Access-Requests sent from the NAS or tunnel server to the RADIUS server. Since the Message-Authenticator attribute involves a HMAC-MD5 hash, it is possible for the RADIUS server to verify the integrity of the Access-Request as well as the NAS or tunnel server’s identity. Similarly, Access-Challenge packets sent from the RADIUS server to the NAS are also authenticated and integrity protected using an HMAC-MD5 hash, enabling the NAS or tunnel server to determine the integrity of the packet and verify the identity of the RADIUS server. Moreover, EAP packets sent via methods that contain their own integrity protection cannot be successfully modified by a rogue NAS or tunnel server.

The second issue that arises in the case of an EAP server and PPP authenticator residing on different machines is that the session key negotiated between the peer and EAP server will need to be transmitted to the authenticator. Therefore a mechanism needs to be provided to transmit the session key from the EAP server to the authenticator or tunnel server that needs to use the key. The specification of this transit mechanism is outside the scope of this document.

7.2.2. Connection hijacking

In this form of attack, the attacker attempts to inject packets into the conversation between the NAS and the RADIUS server, or between the RADIUS server and the backend security server. RADIUS does not support encryption, and as described in [1], only Access-Reply and Access-Challenge packets are integrity protected. Moreover, the integrity protection mechanism described in [1] is weaker than that likely to be used by some EAP methods, making it possible to subvert those methods by attacking EAP/RADIUS.

In order to provide for authentication of all packets in the EAP exchange, all EAP/RADIUS packets MUST be authenticated using the Message-Authenticator attribute, as described previously.

7.2.3. Man in the middle attacks

Since RADIUS security is based on shared secrets, end-to-end security is not provided in the case where authentication or accounting packets are forwarded along a proxy chain. As a result, attackers gaining control of a RADIUS proxy will be able to modify EAP packets in transit.

7.2.4. Multiple databases

In many cases a backend security server will be deployed along with a RADIUS server in order to provide EAP services. Unless the backend security server also functions as a RADIUS server, two separate user databases will exist, each containing information about the security requirements for the user. This represents a weakness, since security may be compromised by a successful attack on either of the servers, or their backend databases. With multiple user databases, adding a new user may require multiple operations, increasing the chances for error. The problems are further magnified in the case where user information is also being kept in an LDAP server. In this case, three stores of user information may exist.

In order to address these threats, consolidation of databases is recommended. This can be achieved by having both the RADIUS server and backend security server store information in the same backend database; by having the backend security server provide a full RADIUS implementation; or by consolidating both the backend security server and the RADIUS server onto the same machine.

7.2.5. Negotiation attacks

In a negotiation attack, a rogue NAS, tunnel server, RADIUS proxy or RADIUS server causes the authenticating peer to choose a less secure authentication method so as to make it easier to obtain the user’s password. For example, a session that would normally be authenticated with EAP would instead authenticated via CHAP or PAP; alternatively, a connection that would normally be authenticated via one EAP type occurs via a less secure EAP type, such as MD5. The threat posed by rogue devices, once thought to be remote, has gained currency given compromises of telephone company switching systems, such as those described in [11].

Protection against negotiation attacks requires the elimination of downward negotiations. This can be achieved via implementation of per-connection policy on the part of the authenticating peer, and per-user policy on the part of the RADIUS server.

For the authenticating peer, authentication policy should be set on a per-connection basis. Per-connection policy allows an authenticating peer to negotiate EAP when calling one service, while negotiating CHAP for another service, even if both services are accessible via the same phone number.

With per-connection policy, an authenticating peer will only attempt to negotiate EAP for a session in which EAP support is expected. As a result, there is a presumption that an authenticating peer selecting EAP requires that level of security. If it cannot be provided, it is likely that there is some kind of misconfiguration, or even that the authenticating peer is contacting the wrong server. Should the NAS not be able to negotiate EAP, or should the EAP-Request sent by the NAS be of a different EAP type than what is expected, the authenticating peer MUST disconnect. An authenticating peer expecting EAP to be negotiated for a session MUST NOT negotiate CHAP or PAP.

For a NAS, it may not be possible to determine whether a user is required to authenticate with EAP until the user’s identity is known. For example, for shared-uses NASes it is possible for one reseller to implement EAP while another does not. In such cases, if any users of the NAS MUST do EAP, then the NAS MUST attempt to negotiate EAP for every call. This avoids forcing an EAP-capable client to do more than one authentication, which weakens security.

If CHAP is negotiated, the NAS will pass the User-Name and CHAP-Password attributes to the RADIUS Server in an Access-Request packet. If the user is not required to use EAP, then the RADIUS Server will respond with an Access-Accept or Access-Reject packet as appropriate. However, if CHAP has been negotiated but EAP is required, the RADIUS server MUST respond with an Access-Reject, rather than an Access-Challenge/EAP-Message/EAP-Request packet. The authenticating peer MUST refuse to renegotiate authentication, even if the renegotiation is from CHAP to EAP.

If EAP is negotiated but is not supported by the RADIUS proxy or server, then the server or proxy MUST respond with an Access-Reject. In these cases, the NAS MUST send an LCP-Terminate and disconnect the user. This is the correct behavior since the authenticating peer is expecting EAP to be negotiated, and that expectation cannot be fulfilled. An EAP-capable authenticating peer MUST refuse to renegotiate the authentication protocol if EAP had initially been negotiated. Note that problems with a non-EAP capable RADIUS proxy could prove difficult to diagnose, since a user dialing in from one location (with an EAP-capable proxy) might be able to successfully authenticate via EAP, while the same user dialing into another location (and encountering an EAP-incapable proxy) might be consistently disconnected.

8. References

[1] Rigney, C., Willens, S., Rubens, A. and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[2] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

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

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

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

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

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

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

[9] Krawczyk, H., Bellare, M. and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[10] Alvestrand, H. and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[11] Slatalla, M., and Quittner, J., "Masters of Deception." HarperCollins, New York, 1995.

9. Acknowledgements

RADIUS and RADIUS Accounting were originally developed by Livingston Enterprises (now part of Lucent Technologies) for their PortMaster series of Network Access Servers.

The section on ARAP is adopted with permission from "Using RADIUS to Authenticate Apple Remote Access Connections" by Ward Willats of Cyno Technologies (ward@cyno.com).

The section on Acct-Interim-Interval is adopted with permission from an earlier work in progress by Pat Calhoun of Sun Microsystems, Mark Beadles of Compuserve, and Alex Ratcliffe of UUNET Technologies.

The section on EAP is adopted with permission from an earlier work in progress by Pat Calhoun of Sun Microsystems, Allan Rubens of Merit Network, and Bernard Aboba of Microsoft. Thanks also to Dave Dawson and Karl Fox of Ascend, and Glen Zorn and Narendra Gidwani of Microsoft for useful discussions of this problem space.

10. Chair’s Address

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

Carl Rigney
Livingston Enterprises
4464 Willow Road
Pleasanton, California 94588

Phone: +1 925 737 2100
EMail: cdr@telemancy.com

11. Authors' Addresses

Questions about this memo can also be directed to:

Carl Rigney
Livingston Enterprises
4464 Willow Road
Pleasanton, California 94588

EMail: cdr@telemancy.com

Questions on ARAP and RADIUS may be directed to:

Ward Willats
Cyno Technologies
1082 Glen Echo Ave
San Jose, CA 95125

Phone: +1 408 297 7766
EMail: ward@cyno.com

Questions on EAP and RADIUS may be directed to any of the following:

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

Phone: +1 650 786 7733
EMail: pcalhoun@eng.sun.com

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

Phone: +1 734 995 1697
EMail: arubens@tutsys.com

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: +1 425 936 6605
EMail: bernarda@microsoft.com

12. Full Copyright Statement

Acknowledgement

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