Cisco Vendor-Specific RADIUS Attributes for the Delivery of Keying Material

Abstract

   This document defines a set of vendor-specific RADIUS Attributes
   designed to allow both the secure transmission of cryptographic
   keying material and strong authentication of any RADIUS message.
   These attributes have been allocated from the Cisco vendor-specific
   space and have been implemented by multiple vendors.

Status of This Memo

   This document is not an Internet Standards Track specification; it is
   published for informational purposes.

   This is a contribution to the RFC Series, independently of any other
   RFC stream. The RFC Editor has chosen to publish this document at
   its discretion and makes no statement about its value for
   implementation or deployment. Documents approved for publication by
   the RFC Editor are not a candidate for any level of Internet
   Standard; see Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6218.

IESG Note

   The IESG has concluded that this work is related to IETF work done in
   the RADEXT WG, but this relationship does not prevent publishing.
   The IESG recommends that the RADEXT WG proceed with the work for an
   interoperable modern key wrap solution using attributes from the
   standard space as part of its charter.

Copyright Notice

Table of Contents

1. Introduction

   This document defines a set of vendor-specific RADIUS Attributes,
   allocated from the Cisco vendor space, that can be used to securely
   transfer cryptographic keying material using standard techniques with
   well-understood security properties. In addition, the Message-
   Authentication-Code Attribute may be used to provide strong
   authentication for any RADIUS message, including those used for
   accounting and dynamic authorization.

   These attributes were designed to provide stronger protection and
   more flexibility than the currently defined Vendor-Specific
   MS-MPPE-Send-Key and MS-MPPE-Recv-Key Attributes in [RFC2548] and the
   Message-Authenticator Attribute in [RFC3579].

   Many remote access deployments (for example, deployments utilizing
   wireless LAN technology) require the secure transmission of
   cryptographic keying material from a RADIUS [RFC2865] server to a
   network access point. This material is usually produced as a
   by-product of an Extensible Authentication Protocol (EAP) [RFC3748]
   authentication and returned in the Access-Accept message following a

successful authentication process. The keying material is of a form
that may be used in virtually any cryptographic algorithm after
appropriate processing. These attributes may also be used in other
cases where an Authentication, Authorization, and Accounting (AAA)
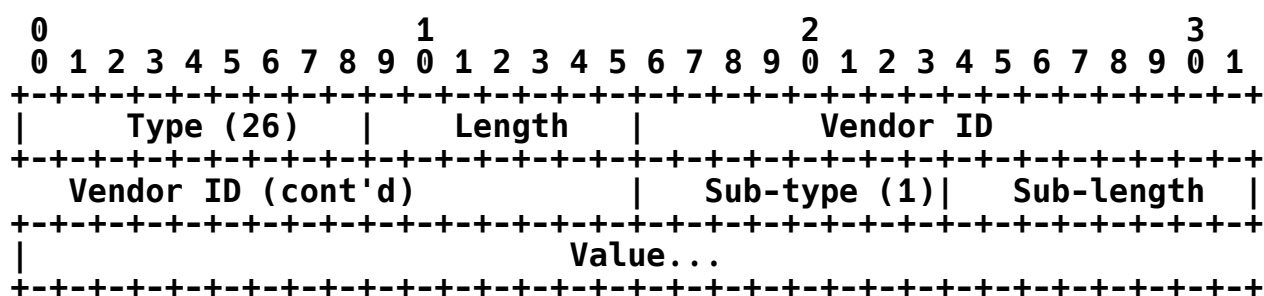server needs to deliver keying material to a network access point.

Discussion of this document may be directed to the authors.

2. 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 [RFC2119].

3. Attributes

The following subsections describe sub-attributes that are
transmitted in RADIUS Attributes of type Vendor-Specific [RFC2865].
The Vendor ID field of the Vendor-Specific Attribute(s) MUST be set
to decimal 9 (Cisco). The general format of the attributes is:

```
 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 (26)   |     Length    |           Vendor ID
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    Vendor ID (cont'd)            |   Sub-type (1)|  Sub-length  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Value...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

   26 for Vendor-Specific

Length

   Length of entire attribute including type and length fields

Vendor ID

   4 octets encoding the Cisco Vendor ID of 9

Sub-type

   Attribute sub-type of 1

   Sub-length

      Length of the sub-attribute including the sub-type and sub-length
      fields

   Value

      Value of the sub-attribute

   This specification concerns the following sub-attributes:

   o Keying-Material

   o MAC-Randomizer

   o Message-Authentication-Code

3.1. Keying-Material

   Description

      This Attribute MAY be used to transfer cryptographic keying
      material from a RADIUS server to a client.

      It MAY be sent in request messages (e.g., Access-Request, etc.),
      as well; if the Keying-Material (KM) Attribute is present in a
      request, it SHOULD be taken as a hint by the server that the
      client prefers this method of key delivery over others. The
      server is not obligated to honor the hint, however. When the
      Keying-Material Attribute is included in a request message, the KM
      ID, key-encrypting-key (KEK) ID, Lifetime, Initialization Vector
      (IV), and Key Material Data fields MAY be omitted.

      In environments where the Keying-Material Attribute is known to be
      supported or in cases where the client wants to avoid roll-back
      attacks, the client MAY be configured to require the use of the
      Keying-Material Attribute. If the client requires the use of the
      Keying-Material Attribute for keying material delivery and it is
      not present in the Access-Accept or Access-Challenge message, the
      client MAY ignore the message in question and end the user
      session.

Any packet that contains a Keying-Material Attribute MUST also include the Message-Authentication-Code Attribute.

Any packet that contains an instance of the Keying-Material Attribute MUST NOT contain an instance of any other attribute (e.g., MS-CHAP-MPPE-Keys [RFC2548], Tunnel-Password [RFC2868], etc.) encapsulating identical keying material.

The Keying-Material Attribute MUST NOT be used to transfer long-lived keys (i.e., passwords) between RADIUS servers and clients.

A summary of the Keying-Material 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 (26)    |     Length     |     Vendor ID
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      Vendor ID (cont'd)               |    Sub-type (1)|   Sub-length |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |              String ID  ("radius:app-key=")
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       String ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       String ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       String ID (cont'd)       |    Enc Type    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          App ID                             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          KEK ID
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KEK ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KEK ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KEK ID (cont'd)                            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          KM ID
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KM ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KM ID (cont'd)
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       KM ID (cont'd)                             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Lifetime                            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                            IV                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         IV (cont'd)                              |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                    Keying Material Data
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

   26 for Vendor-Specific

Length

   Length of entire attribute including type and length fields

Vendor ID

   4 octets encoding the Cisco Vendor ID of 9

Sub-type

   Attribute sub-type of 1

Sub-length

   Length of the sub-attribute including the sub-type and sub-length
   fields

String-ID

   The ASCII characters "radius:app-key=" without quotes or null
   termination

Enc Type

   The Enc Type field indicates the method used to encrypt the
   contents of the Data field. This document defines only one value
   (decimal) for this field:

      0 AES Key Wrap with 128-bit KEK [RFC3394]

   Implementations MUST support Enc Type 0 (AES Key Wrap with 128-bit
   KEK).

   Implementation Note

      A shared secret is used as the key-encrypting-key (KEK) for the
      AES key wrap algorithm. Implementations SHOULD provide a means
      to provision a key (cryptographically separate from the normal
      RADIUS shared secret) to be used exclusively as a KEK.

   App ID

      The App ID field is 4 octets in length and identifies the type of
      application for which the key material is to be used. This allows
      for multiple keys for different purposes to be present in the same
      message. This document defines two values for the App ID:

         0 Reserved

         1 EAP MSK

   KEK ID

      The KEK ID field is 16 octets in length. The combination of the
      KEK ID and the client and server IP addresses together uniquely
      identify a key shared between the RADIUS client and server. As a
      result, the KEK ID need not be globally unique. The KEK ID MUST
      refer to an encryption key of a type and length appropriate for
      use with the algorithm specified by the Enc Type field (see
      above). This key is used to protect the contents of the Data
      field (below). The KEK ID is a constant that is configured
      through an out-of-band mechanism. The same value is configured on
      both the RADIUS client and server. If no KEK ID is configured,
      then the field is set to 0. If only a single KEK is configured
      for use between a given RADIUS client and server, then 0 can be
      used as the default value.

   KM ID

      The KM ID field is 16 octets in length and contains an identifier
      for the contents of the Data field. The KM ID MAY be used by
      communicating parties to identify the material being transmitted.
      The combination of App ID and KM ID MUST uniquely identify the
      keying material between the parties utilizing it. The KM ID is
      assumed to be known to the parties that derived the keying
      material. If the KM ID is not used, it is set to 0. The KM ID
      for the EAP Master Session Key (MSK) application is set to 0.
      Another application that uses the KM ID field can be defined in
      the future.

   Lifetime

      The Lifetime field is an integer [RFC2865] representing the period
      of time (in seconds) for which the keying material is valid.

      Note: Applications using this value SHOULD consider the beginning
      of the lifetime to be the point in time when the keying material
      is first used.

IV

   The length of the IV field depends upon the value of the Enc Type
   field, but is fixed for any given value thereof. When the value
   of the Enc Type field is 0 (decimal), the IV field MUST be 8
   octets in length (as illustrated above), and the value of the IV
   field MUST be as specified in [RFC3394]. If the IV for Enc Type 0
   does not match [RFC3394], then the receiver MUST NOT use the key
   material from this attribute.

Keying Material Data

   The Keying Material Data field is of variable length and contains
   the actual encrypted keying material.

3.2. MAC-Randomizer

Description

   The MAC-Randomizer Attribute MUST be present in any message that
   includes an instance of the Message-Authentication-Code Attribute.
   The Random field MUST contain a 32-octet random number that SHOULD
   satisfy the requirements of [RFC4086].

   Implementation Note

      The Random field MUST be filled in before the Message
      Authentication Code (MAC) is computed. The MAC-Randomizer
      Attribute SHOULD be placed at the beginning of the RADIUS
      message if possible.

   A summary of the MAC-Randomizer 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 (26)  |     Length    |    Vendor ID
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        Vendor ID (cont'd)           |  Sub-type (1)|   Sub-length |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                 String ID  ("radius:random-nonce=")
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                           String ID (cont'd)
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                           String ID (cont'd)
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                           String ID (cont'd)
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                           String ID (cont'd)                       |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         Random...
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Type

      26 for Vendor-Specific

   Length

      Length of entire attribute including type and length fields

   Vendor ID

      4 octets encoding the Cisco Vendor ID of 9

   Sub-type

      Attribute sub-type of 1

   Sub-length

      Length of the sub-attribute including the sub-type and
      sub-length fields

   String-ID

      The ASCII characters "radius:random-nonce=" without quotes or
      null termination

   Random

      This field MUST contain a 32 octet random number that SHOULD
      satisfy the requirements of [RFC4086].

3.3. Message-Authentication-Code

   Description

      This Attribute MAY be used to "sign" messages to prevent spoofing.
      If it is present in a request, the receiver should take this as a
      hint that the sender prefers the use of this Attribute for message
      authentication; the receiver is not obligated to do so, however.

      The Message-Authentication-Code Attribute MUST be included in any
      message that contains a Keying-Material Attribute.

      If both the Message-Authentication-Code and Message-Authenticator
      Attributes are to be included in a message (e.g., for backward
      compatibility in a network containing both old and new clients),
      the value of the Message-Authentication-Code Attribute MUST be
      computed first.

      If any message is received containing an instance of the Message-
      Authentication-Code Attribute, the receiver MUST calculate the
      correct value of the Message-Authentication-Code and silently
      discard the packet if the computed value does not match the value
      received.

      If a received message contains an instance of the MAC-Randomizer
      Attribute (Section 3.2), the received MAC-Randomizer Attribute
      SHOULD be included in the computation of the Message-
      Authentication-Code Attribute sent in the response, as described
      below.

      A summary of the Message-Authentication-Code 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 (26)   |    Length     |              Vendor ID
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      Vendor ID (cont'd)           |  Sub-type (1)|   Sub-length  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |       String ID  ("radius:message-authenticator-code=")
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                         String ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
           String ID (cont'd)       |   MAC Type    |  MAC Key ID
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      MAC Key ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                        MAC Key ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                        MAC Key ID (cont'd)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                        MAC Key ID (cont'd)        |      MAC
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         MAC (cont'd) ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

   26 for Vendor-Specific

Length

   Length of entire attribute including type and length fields

Vendor ID

   4 octets encoding the Cisco Vendor ID of 9

   Sub-type

      Attribute sub-type of 1

   Sub-length

      Length of the sub-attribute including the sub-type and
      sub-length fields

   String-ID

      The ASCII characters "radius:message-authenticator-code="
      without quotes or null termination

   MAC Type

      The MAC Type field specifies the algorithm used to create the
      value in the MAC field. This document defines six values for
      the MAC Type field:

         0 HMAC-SHA-1 [FIPS] [RFC2104]

         1 HMAC-SHA-256 [FIPS] [RFC4231]

         2 HMAC-SHA-512 [FIPS] [RFC4231]

         3 CMAC-AES-128 [NIST]

         4 CMAC-AES-192 [NIST]

         5 CMAC-AES-256 [NIST]

      Implementations MUST support MAC Type 0 (HMAC-SHA-1).

   MAC Key ID

      The MAC Key ID field is 16 octets in length and contains an
      identifier for the key. The combination of the MAC Key ID and
      the client and server IP addresses together uniquely identify a
      key shared between the RADIUS client and server. As a result,
      the MAC Key ID need not be globally unique. The MAC Key ID
      MUST refer to a key of a type and length appropriate for use
      with the algorithm specified by the MAC Type field (see above).

The MAC Key ID is a constant that is configured through an out-of-band mechanism. The same value is configured on both the RADIUS client and server. If no MAC Key ID is configured, then the field is set to 0. If only a single MAC Key ID is configured for use between a given RADIUS client and server, then 0 can be used as the default value.

MAC

Both the length and value of the MAC field depend upon the algorithm specified by the value of the MAC Type field. If the algorithm specified is HMAC-SHA-1, HMAC-SHA-256, or HMAC-SHA-512, the MAC field MUST be 20, 32, or 64 octets in length, respectively. If the algorithm specified is CMAC-AES-128, CMAC-AES-192, or CMAC-AES-256, the MAC field SHOULD be 64 octets in length. The derivation of the MAC field value for all the algorithms specified in this document is identical, except for the algorithm used. There are differences, however, depending upon whether the MAC is being computed for a request message or a response. These differences are detailed below, with the free variable HASH-ALG representing the actual algorithm used.

Request Messages

For requests (e.g., CoA-Request [RFC5176], Accounting-Request [RFC2866], etc.), the value of the MAC field is a hash of the entire packet except the Request Authenticator in the header of the RADIUS packet, using a shared secret as the key, as follows.

MAC = MAC-ALG(Key, Type + Identifier + Length + Attributes)
    where '+' represents concatenation

The MAC-Randomizer Attribute (Section 3.2) MUST be included in any request in which the Message-Authentication-Code Attribute is used. The Random field of the MAC-Randomizer Attribute MUST be filled in before the value of the MAC field is computed.

If the Message-Authenticator-Code Attribute is included in a client request, the server SHOULD ignore the contents of the Request Authenticator.

Implementation Notes

> When the hash is calculated, both the MAC field of the
> Message-Authenticator-Code Attribute and the String field
> of the Message-Authenticator Attribute (if any) MUST be
> considered to be zero-filled.

> Implementations SHOULD provide a means to provision a key
> (cryptographically separate from the normal RADIUS shared
> secret) to be used exclusively in the generation of the
> Message-Authentication-Code.

Response Messages

For responses (e.g., CoA-ACK [RFC5176], Accounting-Response
[RFC2866], etc.), the value of the MAC field is a hash of
the entire packet except the Response Authenticator in the
header of the RADIUS packet using a shared secret as the
key, as follows.

MAC = HASH-ALG(Key, Type + Identifier + Length + Attributes)
   where '+' represents concatenation

If the request contained an instance of the MAC-Randomizer
Attribute and the responder wishes to include an instance of
the Message-Authentication-Code Attribute in the
corresponding response, then the MAC-Randomizer Attribute
from the request MUST be included in the response.

If the Message-Authenticator-Code Attribute is included in a
server response, the client SHOULD ignore the contents of
the Response Authenticator.

Implementation Notes

> When the hash is calculated, both the MAC field of the
> Message-Authenticator-Code Attribute and the String field
> of the Message-Authenticator Attribute (if any) MUST be
> considered to be zero-filled.

> The Message-Authentication-Code Attribute MUST be created
> and inserted in the packet before the Response
> Authenticator is calculated.

> Implementations SHOULD provide a means to provision a key
> (cryptographically separate from the normal RADIUS shared
> secret) to be used exclusively in the generation of the
> Message-Authentication-Code.

4. Security Considerations

   It is RECOMMENDED in this memo that two new keys, a key encrypting
   key and a message authentication key, be shared by the RADIUS client
   and server. If implemented, these two keys MUST be different from
   each other and SHOULD NOT be based on a password. These two keys
   MUST be cryptographically independent of the RADIUS shared secret
   used in calculating the Response Authenticator [RFC2865], Request
   Authenticator [RFC2866] [RFC5176], and Message-Authenticator
   Attribute [RFC3579]; otherwise, if the shared secret is broken, all
   is lost.

   To avoid the possibility of collisions, the same MAC key SHOULD NOT
   be used with more than $2^{(n/2)}$ messages, where 'n' is the length of
   the MAC value in octets.

   If a packet that contains an instance of the Keying-Material
   Attribute also contains an instance of another, weaker key transport
   attribute (e.g., MS-MPPE-Recv-Key [RFC2548]) encapsulating identical
   keying material, then breaking the weaker attribute might facilitate
   a known-plaintext attack against the KEK.

5. Contributors

   Hao Zhou, Nancy Cam-Winget, Alex Lam, Paul Funk, and John Fossaceca
   all contributed to this document.

6. Acknowledgements

   Thanks (in no particular order) to Keith McCloghrie, Kaushik Narayan,
   Murtaza Chiba, Bill Burr, Russ Housley, David McGrew, Pat Calhoun,
   Joel Halpern, Jim Schaad, Greg Weber, and Bernard Aboba for useful
   feedback.

7. References

7.1. Normative References

   [FIPS]   National Institute of Standards and Technology, "Secure
               Hash Standard (SHS)", FIPS PUB 180-3, October 2008.

   [NIST]   Dworkin, M., "Recommendation for Block Cipher Modes of
               Operation: The CMAC Mode for Authentication", NIST SP800-
               38B, May 2005.

   [RFC2104]   Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-
               Hashing for Message Authentication", RFC 2104,
               February 1997.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson,
              "Remote Authentication Dial In User Service (RADIUS)",
              RFC 2865, June 2000.

   [RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

   [RFC2868] Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege,
              M., and I. Goyret, "RADIUS Attributes for Tunnel Protocol
              Support", RFC 2868, June 2000.

   [RFC3394] Schaad, J. and R. Housley, "Advanced Encryption Standard
              (AES) Key Wrap Algorithm", RFC 3394, September 2002.

   [RFC3579] Aboba, B. and P. Calhoun, "RADIUS (Remote Authentication
              Dial In User Service) Support For Extensible
              Authentication Protocol (EAP)", RFC 3579, September 2003.

   [RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker,
              "Randomness Requirements for Security", BCP 106, RFC 4086,
              June 2005.

   [RFC4231] Nystrom, M., "Identifiers and Test Vectors for
              HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and
              HMAC-SHA-512", RFC 4231, December 2005.

   [RFC5176] Chiba, M., Dommety, G., Eklund, M., Mitton, D., and B.
              Aboba, "Dynamic Authorization Extensions to Remote
              Authentication Dial In User Service (RADIUS)", RFC 5176,
              January 2008.

7.2. Informative References

   [RFC2548] Zorn, G., "Microsoft Vendor-specific RADIUS Attributes",
              RFC 2548, March 1999.

   [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H.
              Levkowetz, Ed., "Extensible Authentication Protocol
              (EAP)", RFC 3748, June 2004.

Authors' Addresses

   Glen Zorn
   Network Zen
   227/358 Thanon Sanphawut
   Bang Na, Bangkok 10260
   Thailand

   Phone: +66 (0) 87 040 4617
   EMail: gwz@net-zen.net

   Tiebing Zhang
   Advista Technologies
   5252 Orange Ave., Suite 106
   Cypress, CA 90630
   US

   Phone: +1 (949) 242 0391
   EMail: tzhang@advistatech.com

   Jesse Walker
   Intel Corporation
   JF2-55
   2111 N.E. 25th Ave.
   Hillsboro, OR 97214-5961
   US

   Phone: +1 (503) 712-1849
   EMail: jesse.walker@intel.com

   Joseph Salowey
   Cisco Systems
   2901 Third Avenue
   SEA1/6/
   Seattle, WA 98121
   US

   Phone: +1 (206) 256-3380
   EMail: jsalowey@cisco.com