Internet Engineering Task Force
AAA Working Group
Internet Draft
draft-sterman-aaa-sip-00.txt
Nov 12, 2001
Expires: Apr 12 2002

B. Sterman
D. Sadolevsky
D. Schwartz
deltathree, Inc.
D. Williams
Cisco Systems

RADIUS Extension for Digest Authentication

STATUS OF THIS MEMO

Abstract

Basic and Digest authentication schemes (RFC2617 [1]) are widely used in protocols such as SIP (RFC2543 [2]) and HTTP (RFC2616 [3]). RADIUS (RFC2865 [4]) is a protocol for back end authentication. RADIUS supports Basic authentication natively, as well as several other authentication schemes, such as CHAP, but does not support Digest authentication scheme. This document describes an extension to RADIUS for Digest authentication and provides a scenario of Digest user authentication.

1 Introduction

1.1 Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALLNOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119.

1.2 Scenario

Figure 1 depicts the scenario that is relevant for this document. It shows a generic case where entities A and B communicate in the front-end using protocols such as HTTP/SIP, while entities B and C communicate in the back-end using RADIUS.

```
       HTTP/SIP           RADIUS

 +-----+    (1)    +-----+           +-----+
 |     |==========>|     |           |     |
 |     |    (2)    |     |           |     |
 |     |<==========|     |           |     |
 |     |    (3)    |     |           |     |
 |     |==========>|     |           |     |
 |  A  |           |  B  |    (4)    |  C  |
 |     |           |     |---------->|     |
 |     |           |     |    (5)    |     |
 |     |           |     |<----------|     |
 |     |    (6)    |     |           |     |
 |     |<==========|     |           |     |
 +-----+           +-----+           +-----+

 ====> HTTP/SIP
 ----> RADIUS
```

Figure 1: Scenario relevant to document

The roles played by the entities in this scenario are as follows:

A: HTTP client / SIP UA

B: {HTTP server / HTTP proxy server / SIP proxy server / SIP UAS} acting also as a RADIUS NAS

C: RADIUS server

The relevant order of messages sent in this scenario is as follows:

A sends B an HTTP/SIP request without authorization header (step 1). B challenges A sending an HTTP/SIP "(Proxy) Authorization required" response containing a locally generated nonce (step 2). A sends B an HTTP/SIP request with authorization header (step 3). B sends C a RADIUS Access-Request with attributes described in this document (step 4). C responds to B with a RADIUS Access-Accept/Access-Reject response (step 5).

If credentials were accepted B receives an Access-Accept response and the message sent from A is considered authentic. If B receives an Access-Reject response, however, B then responds to A with a "(Proxy) Authorization required" response (step 6).

1.3 Motivation

Basic and Digest authentication are used within protocols such as HTTP and SIP. Recently, there have been efforts towards the use of an Extensible Authentication Protocol (EAP) within protocols such as HTTP and SIP. [5] is one such effort. The advantage here is that, new authentication schemes may be used without any modification to the SIP/HTTP protocol itself. This is because the EAP packet for the particular authentication scheme is carried transparently by the SIP/HTTP protocol.

However, the use of Basic and Digest authentication is likely to continue to be used directly within protocols such as SIP/HTTP in the near future, and hence their interoperability with a back-end authentication protocol such as RADIUS is needed.

There is also an ongoing effort to accomplish the same thing as this document does in relation to DIAMETER [6], but DIAMETER itself has not reached the RFC status as of the time of writing this. When it happens and when [6] reaches the RFC status too, implementers are encouraged to switch to [6].

1.4 Approach

The approach taken here is to extend RADIUS to support Digest authentication by mimicking its native support for CHAP authentication. According to [4], the RADIUS server distinguishes between different authentication schemes by looking at the presence of an attribute specific for that scheme. For the three natively supported authentication schemes, these attributes are: User-Password for PAP (or any other clear-text password scheme), CHAP-Password for CHAP, and State + User-Password for challenge-response scheme. This document adds another attribute to be used in this role: Digest-Response. Also according to [4], "An Access-Request packet MUST contain either a User-Password or a CHAP-Password or a State. It MUST NOT contain both a User-Password and a CHAP-Password. If future extensions allow other kinds of authentication information to be conveyed, the attribute for that can be used instead of User-Password or CHAP-Password." The Digest-Response introduced here therefore can be used instead of User-Password or CHAP-Password.

The HTTP Authentication parameters found in the Proxy-Authorization or Authorization request header are mapped into two newly defined experimental RADIUS attributes. The Digest-Response attribute and the Digest-Attributes attribute carrying multiple HTTP Digest parameters as subattributes. These 2 new RADIUS attributes are defined in the document together with some other information required for calculating the correct digest response on the RADIUS server with exception of the password, which the RADIUS server is assumed to be able to retrieve from a data store given the username. The structure of Digest-Response, the structure of Digest-Attributes and the mapping/meaning of its subattributes are described in the next chapter.

2 New RADIUS attributes

2.1 Digest-Response attribute

Description

This attribute contains the request-digest response value contained in a Digest (Proxy)Authorization header. It is only used in Access-Request packets. If this attribute is present, the RADIUS server SHOULD view the Access-Request as a Digest one.

A summary of the Digest-Attributes 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |  Length       | String...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

206(Experimental) for Digest-Response.

Length

34

String

String which proves the user knows a password. The String field is 32 octets long and contains hexadecimal representation of 16 octet digest value as it was calculated by the authenticated client. The String field SHOULD be copied from request-digest of digest-response ([1]).

2.2 Digest-Attributes attribute

Description

This attribute contains subattributes which indicate the values contained in a Digest (Proxy)Authorization header together with other information necessary to calculate the correct digest response value. It is only used in Access-Request packets. There can be multiple Digest-Attributes attributes contained in one Access-Request packet. In this case RADIUS server MUST interpret a concatenation of their values as if it came in one attribute.

A summary of the Digest-Attributes 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

207(Experimental) for Digest-Attributes.

Length

>= 5

String

The String field is 3 or more octets and contains one or more subattributes. Format of a subattribute is shown be¡ low. 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|     Sub-Type  |  Sub-Length   | Sub-Value...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Sub-Type

Subattribute type. Meanings of the following defined types can be found in section 2.3

1	Realm
2	Nonce
3	Method
4	URI

5	QOP
6	Algorithm
7	Body-Digest
8	CNonce
9	Nonce-Count
10	User-Name

Sub-Length

>= 3

Sub-Value

Subattribute-specific value

2.3.1 Realm

Sub-Type

1

Sub-Length

>= 3

Sub-Value

String, copied from realm-value of digest-response ([1])

2.3.2 Nonce

Sub-Type

2

Sub-Length

>= 3

Sub-Value

String, copied from nonce-value of digest-response ([1])

2.3.3 Method

Sub-Type

3

Sub-Length

>= 3

Sub-Value

String, copied from digest-response. Method is taken from request-URI of message ([2/3])

2.3.4 URI

Sub-Type

4

Sub-Length

>= 3

Sub-Value

String, copied from digest-uri-value of digest-response ([1])

2.3.5 QOP

Sub-Type

5

Sub-Length

>= 3

Sub-Value

String, copied from qop-value of digest-response ([1])

2.3.6 Algorithm

Sub-Type

6

Sub-Length

>= 3

Sub-Value

String, "MD5" | "MD5-sess" | token, copied from algorithm of digest-response ([1])

2.3.7 Body-Digest

Sub-Type

7

Sub-Length

34

Sub-Value

String, hexadecimal representation of a digest calculated over entity-body of HTTP/SIP request ([1/2]). Computed by entity B in figure 1. This attribute is not part of the HTTP Digest response.

2.3.8 CNonce

Sub-Type

8

Sub-Length

>= 3

Sub-Value

String copied from cnonce-value of digest-response ([1])

2.3.9 Nonce-Count

Sub-Type

9

Sub-Length

= 10

Sub-Value

String, 8LHEX, copied from nc-value of digest-response ([1])

2.3.10 User-Name

Sub-Type

10

Sub-Length

>= 3

Sub-Value

String copied from username-value of digest-response ([1]) the RADIUS server SHOULD NOT use this value for password finding, but only for digest calculation purpose. In order to find the user record containing password, the RADIUS server SHOULD use the value of the User-Name _attribute_

3 Example

This is an example sniffed from the traffic between HearMe softphone (A), Cisco Systems Proxy Server (B) and deltathree RADIUS server (C) (The communication between Cisco Systems Proxy Server and a SIP PSTN gateway is omitted for brevity):

```
A->B

   INVITE sip:97226491335@213.137.69.38 SIP/2.0
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=216ae97f
   To: sip:97226491335@213.137.69.38
   Contact: sip:12345678@213.137.67.67:5061
   Call-ID: da591c98-f056-4803-a751-0bd296170875@213.137.67.67
   CSeq: 2544265 INVITE
   Content-Length: 150
   Content-Type: application/sdp
   User-Agent: HearMe SoftPHONE

   v=0
   o=HearMe 2544265 2544265 IN IP4 213.137.67.67
   s=HearMe
   c=IN IP4 213.137.67.67
   t=0 0
   m=audio 8000 RTP/AVP 0 4
   a=ptime:20
   a=x-ssrc:009aa330

B->A

   SIP/2.0 100 Trying
   Via: SIP/2.0/UDP 213.137.67.67:5061
   Call-ID: da591c98-f056-4803-a751-0bd296170875@213.137.67.67
   From: <sip:12345678@213.137.67.67>;tag=216ae97f
   To: sip:97226491335@213.137.69.38
   CSeq: 2544265 INVITE
   Content-Length: 0

B->A

   SIP/2.0 407 Proxy Authentication Required
   Via: SIP/2.0/UDP 213.137.67.67:5061
   Call-ID: da591c98-f056-4803-a751-0bd296170875@213.137.67.67
   From: <sip:12345678@213.137.67.67>;tag=216ae97f
   To: sip:97226491335@213.137.69.38;tag=3f5611de-22a007dc
   CSeq: 2544265 INVITE
   Proxy-Authenticate: DIGEST realm="deltathree", nonce="3bada1a0",
algorithm="md5"
   Content-Length: 0
```

A->B

```
   ACK sip:97226491335@213.137.69.38 SIP/2.0
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=216ae97f
   To: sip:97226491335@213.137.69.38;tag=3f5611de-22a007dc
   Call-ID: da591c98-f056-4803-a751-0bd296170875@213.137.67.67
   CSeq: 2544265 ACK
   Content-Length: 0
```

A->B

```
   INVITE sip:97226491335@213.137.69.38 SIP/2.0
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=29e97f
   To: sip:97226491335@213.137.69.38
   Contact: sip:12345678@213.137.67.67:5061
   Call-ID: b0f487c9-04a0-4108-a5a3-580ecbaf0e24@213.137.67.67
   CSeq: 2544266 INVITE
   Content-Length: 150
   Content-Type: application/sdp
   User-Agent: HearMe SoftPHONE
   Proxy-Authorization: DIGEST algorithm="md5",nonce="3bada1a0",opaque=""
        ,realm="deltathree",response="2ae133421cda65d67dc50d13ba0eb9bc"
        ,uri="sip:97226491335@213.137.69.38",username="12345678"

   v=0
   o=HearMe 2544265 2544265 IN IP4 213.137.67.67
   s=HearMe
   c=IN IP4 213.137.67.67
   t=0 0
   m=audio 8000 RTP/AVP 0 4
   a=ptime:20
   a=x-ssrc:009aa330
```

B->A

```
   SIP/2.0 100 Trying
   Via: SIP/2.0/UDP 213.137.67.67:5061
   Call-ID: b0f487c9-04a0-4108-a5a3-580ecbaf0e24@213.137.67.67
   From: <sip:12345678@213.137.67.67>;tag=29e97f
   To: sip:97226491335@213.137.69.38
   CSeq: 2544266 INVITE
   Content-Length: 0
```

```
B->C

   Code = 1 (Access-Request)
   Identifier = 1
   Length = 164
   Authenticator = 56 7b e6 9a 8e 43 cf b6 fb a6 c0 f0 9a 92 6f 0e
   Attributes:
   NAS-IP-Address = d5 89 45 26 (213.137.69.38)
   NAS-Port-Type = 5 (Virtual)
   User-Name = "12345678"
   Digest-Response (206) = "2ae133421cda65d67dc50d13ba0eb9bc"
   Digest-Attributes (207) = [Realm (1) = "deltathree"]
   Digest-Attributes (207) = [Nonce (2) = "3bada1a0"]
   Digest-Attributes (207) = [Method (3) = "INVITE"]
   Digest-Attributes (207) = [URI (4) = "sip:97226491335@213.137.69.38"]
   Digest-Attributes (207) = [Algorithm (5) = "md5"]
   Digest-Attributes (207) = [User-Name (10) = "12345678"]

C->B

   Code = 2 (Access-Accept)
   Identifier = 1
   Length = 20
   Authenticator = 6d 76 53 ce aa 07 9a f7 ac b4 b0 e2 96 2f c4 0d

B->A

   SIP/2.0 180 Ringing
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=29e97f
   To: sip:97226491335@213.137.69.38;tag=7BF5248C-177E
   Date: Tue, 25 Jan 2000 03:41:00 gmt
   Call-ID: b0f487c9-04a0-4108-a5a3-580ecbaf0e24@213.137.67.67
   Server: Cisco-SIPGateway/IOS-12.x
   Record-Route: <sip:97226491335@213.137.69.38:5060;maddr=213.137.69.38>
   CSeq: 2544266 INVITE
   Content-Length: 0

B->A

   SIP/2.0 200 OK
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=29e97f
   To: sip:97226491335@213.137.69.38;tag=7BF5248C-177E
   Date: Tue, 25 Jan 2000 03:41:00 gmt
   Call-ID: b0f487c9-04a0-4108-a5a3-580ecbaf0e24@213.137.67.67
   Server: Cisco-SIPGateway/IOS-12.x
   Record-Route: <sip:97226491335@213.137.69.38:5060;maddr=213.137.69.38>
   CSeq: 2544266 INVITE
   Contact: <sip:97226491335@213.137.69.36:5060;user=phone>
   Content-Type: application/sdp
   Content-Length: 158

   v=0
   o=CiscoSystemsSIP-GW-UserAgent 1901 5895 IN IP4 213.137.69.36
   s=SIP Call
   c=IN IP4 213.137.69.36
   t=0 0
   m=audio 17724 RTP/AVP 0
   a=rtpmap:0 PCMU/8000
```

A->B

```
   ACK sip:97226491335@213.137.69.38:5060 SIP/2.0
   Via: SIP/2.0/UDP 213.137.67.67:5061
   From: <sip:12345678@213.137.67.67>;tag=29e97f
   To: sip:97226491335@213.137.69.38;tag=7BF5248C-177E
   Call-ID: b0f487c9-04a0-4108-a5a3-580ecbaf0e24@213.137.67.67
   CSeq: 2544266 ACK
   Content-Length: 0
   Route: <sip:97226491335@213.137.69.36:5060;user=phone>
```

References

[1] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[2] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, "SIP: Session Initiation Protocol", draft-ietf-sip-rfc2543bis-03.txt, IETF work in progress, May 2001.

[3] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

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

[5] J. Arkko, V. Torvinen, A. Niemi, "HTTP Authentication with EAP", draft-http-eap-basic-04.txt, IETF work in progress, June 2001.

[6] Srinivas, Chan, Sengodan, Costa-Requena, "Mapping of Basic and Digest Authentication to DIAMETER AAA Messages", draft-srinivas-aaa-basic-digest-00.txt, IETF work in progress, July 2001

Acknowledgements

We would like to acknowledge Kevin Mcdermott (Cisco Systems) for providing comments and experimental implementation.

Authors's Addresses

Baruch Sterman
Daniel Sadolevsky
David Schwartz

deltathree, Inc.
Jerusalem Technology Park
P.O. Box 48265
Jerusalem 91481 Israel
{baruch,daniels,davids}@deltathree.com

David Williams
Cisco Systems
7025 Kit Creek Road
P.O. Box 14987
Research Triangle Park, NC 27709
USA
Email: dwilli@cisco.com