SIPPING Working Group
Internet Draft
Category: Standards Track
Expires on: May 2002

Sanjoy Sen
Lee Valerius
Nortel Networks

Vesa Torvinen
Ericsson

November 2001

Single Hop Message Authentication in SIP

<draft-sen-sipping-onehop-digest-00.txt>

Status of this Memo

Abstract

To date, the HTTP access authentication framework, as described in [RFC2617] and as used in [SIPbis05], has permitted limited SIP message authentication from UAC to Proxy/UAS, Proxy to Proxy, and Proxy to UAS. This draft addresses some of the shortcomings of SIP usage of Digest for message authentication between a SIP User Agent and a Proxy one hop away (e.g., an outbound Proxy). For the messages exchanged between the UA and a Proxy one hop away, the Service Provider may want to provide a different level of protection than that possible for the same messages end-to-end. Authentication of both requests and responses traveling in either direction should be possible with minimum number of necessary roundtrip exchanges. We discuss some the limitations of SIP Digest message authentication framework in satisfying these requirements and propose possible solutions. A new value of the "qop-options" parameter would indicate to a SIP entity that the challenging entity is one hop away and the

maximum protection of SIP message is required. Some other aspects of

this solution are in the form of behavior enhancements of SIP Proxy and UA.

Table of Contents

1 Introduction

To date, the HTTP access authentication framework, as described in [RFC2617] and as used in [SIPbis05], has permitted limited SIP message authentication from UAC to Proxy/UAS, Proxy to Proxy, and Proxy to UAS. This draft addresses some of the shortcomings of SIP usage of Digest for message authentication between a SIP User Agent and a Proxy one hop away (e.g., an outbound Proxy). For the messages exchanged between the UA and a Proxy one hop away, the Service Provider may want to provide a different level of protection than that possible for the same messages end-to-end. Thus, it may be required that integrity protection of the entire message (except perhaps the header carrying the credential) be provided. Authentication of both requests and responses traveling in either direction should be possible with minimum number of necessary roundtrip exchanges. The latter consideration is particularly important for access networks that are resource-constrained and prone to large round-trip times.

In Section 3, we discuss some the limitations of SIP Digest message authentication framework in satisfying some of the above requirements and propose possible solutions. A new value of the "qop-options" parameter would indicate to a SIP entity that the challenging entity is one hop away and the maximum protection of SIP message is required. Other aspects of this solution are in the form of behavior enhancements of SIP Proxy and UA. In Section 4, the solution is exemplified with some high-level call flows.

2 Conventions used in this document

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.

3 Digest for SIP Message Authentication between UA and Proxy one hop away

We believe that the requirements discussed in the rest of this section are either not clearly addressed in the existing SIP authentication framework or not addressed at all.

Requirement# 1: It would be possible to authenticate all SIP messages between the UA and the Proxy at the level of protection negotiated between them.

This can be decomposed into two scenarios.

A) UAC-Proxy message authentication:

For authenticating requests from the UAC [RFC2617], the Proxy issues the Digest challenge in the Proxy-Authenticate header in a 407 response. In response to the challenge, the UAC should include the credential in Proxy-Authorization header and resubmit the request.

It is not clear from [RFC2617] or [SIPbis05] how the response forwarded upstream by the Proxy towards the UAC will be authenticated at the protection level negotiated between the Proxy and the UAC. It is proposed here that the Proxy insert the Authentication-Info header (with the proper credential) in the response that it forwards upstream towards the UAC.

B) UAS-Proxy message authentication:

According to [RFC2617], the UAS can authenticate requests forwarded by the Proxy as follows: the UAS must generate a 407 response with a Proxy-Authenticate header containing a Digest challenge. In response, the Proxy should re-submit the request with a Proxy-Authorization header carrying the credential. All subsequent responses from the UAS to be authenticated by the Proxy should carry the Proxy-Authentication-Info header with proper credential.

However, a couple of problems arise for UAS-Proxy authentication in SIP. First, the use of Proxy-Authentication-Info header is not mentioned in [SIPbis05]. Secondly, a Proxy is prohibited from adding the Proxy-Authorization header to a forwarded request, unless the request is re-submitted. It is required that a Proxy re-submitting a

request must increase the CSeq header field of the request implying that when the corresponding response is received at the UAC, it would be dropped. To alleviate the problem, it has been suggested in the list that the Proxy should be able to "resubmit" a request just by changing the branch parameter of the top-most Via header (this is equivalent of doing an empty fork). To the UAS, this is a new transaction anyway.

If the UA and the Proxy had already authenticated each other, this would allow the Proxy to insert a Proxy-Authorization header (containing its credential) in an incoming request to be forwarded preemptively (i.e., without waiting for a challenge, and thereby avoiding a roundtrip) to the UAS. If the credential is deemed valid by the UAS, the response sent back should contain a Proxy-Authentication-Info header for mutual authentication by the Proxy. If the credential is deemed invalid to the UAS, it will send a 407 response with a Proxy-Authenticate header containing a Digest challenge and the Proxy would "re-submit" the request in the same way as above.

Requirement # 2: The security mechanism must be able to protect a SIP message to the maximum extent possible, when the SIP entities are just one hop away. Also, the framework should support replay protection for all messages.

This is decomposed into two parts, which are evaluated separately.

A) Maximum Integrity protection of SIP messages:

Digest supports integrity protection of the SIP message body (not the headers) when the qop-options directive within the Digest challenge is set to the value "auth-int". A new qop-options value - "auth-extd-int" is proposed, which when set by the SIP entity one hop away issuing the challenge, will direct the client to include for integrity protection all headers and bodies of the message that are mutually agreed on for maximum protection. For example, this might mean that the A2 parameter of the Digest response [RFC2617] is computed as follows:
A2 = H(entire message with all headers in canonical form, excluding the header which carries the credential).

B) Anti-replay protection:

This is really a function of how the server generates the nonce. In order to limit performance impact, it may be required that the same nonce be used over multiple messages. In that case, the nonce-count is useful to provide replay protection. It is recommended that the Proxy server generate a new nonce value whenever possible. For example, if the UAS sends its authorization credentials to the Proxy

in the Proxy-Authentication-Info header, it should send a new next-nonce value.

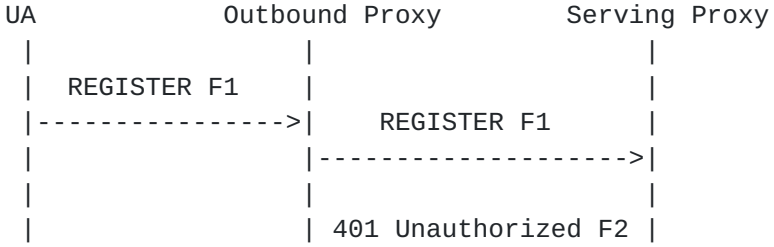
Requirement # 3: In order to avoid excess round-trip, a Proxy should be able to piggyback its challenge in a 401 or 407 response that it forwards upstream to the UAC. This is useful in certain operations where the user authentication and message authentication mechanisms are different and take place at different network entities. An example of this is the third generation mobile network [3gpp-req] where the authentication of the SIP UA might be conducted at an entity different than the Proxy with whom the UA establishes the message integrity relationship.

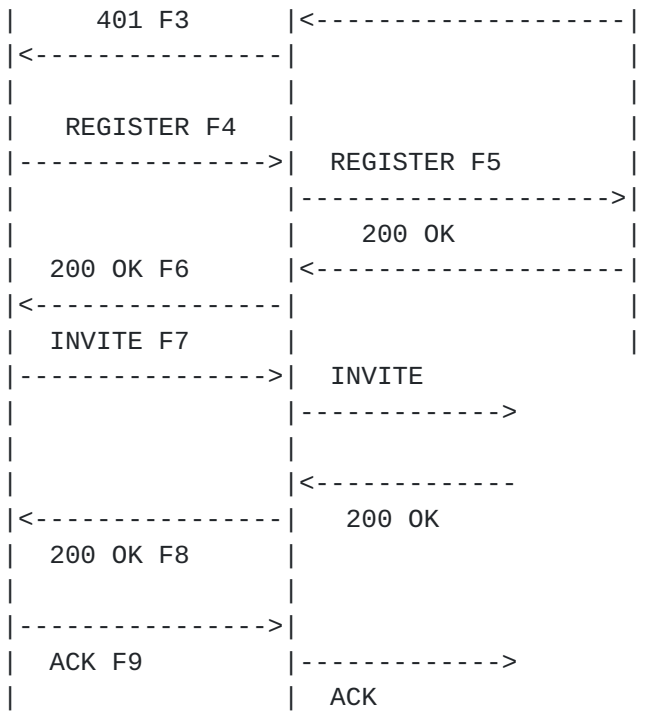
[RFC2617] notes that if a client is to be authenticated by multiple entities, the challenges must be carried in different responses. However, [SIPbis05] allows for the Proxy to aggregate multiple challenges in responses to forked requests and insert them to a single 401 or 407 response to be sent upstream. The same mechanism can possibly be leveraged by the Proxy, which can add a Proxy-Authenticate header (carrying its challenge) to a 401/407 response that will be forwarded upstream. Generally, a Proxy sends its challenge upstream in a 407 response. The UAC responds with a matching credential for each challenge.

4 Example Call Flows

We will consider an example utilizing a mobile, wireless terminal as UA to illustrate some of the above proposals. There is a SIP serving proxy (also acting as a Registrar) that would authenticate the UA and would also support the ability to terminate INVITEs to the UA. There is a SIP outbound Proxy that acts as a "point of presence" for the roaming UA to the SIP world. At the time of registration, the roaming user is authenticated by the serving Proxy. Subsequently, all messages between the user agent and the outbound Proxy must be authenticated. Two cases are considered.

CASE 1: UA registering and originating a call





F1: UA sends a REGISTER message to the outbound Proxy, which is forwarded to the serving Proxy.
F2: The serving Proxy returns a 401 "Unauthorized" message containing a WWW-Authenticate header carrying an authentication challenge. The challenge may utilize any known authentication method.

```
SIP/2.0 401 Unauthorized
...
WWW-Authenticate:...
```

F3: The outbound Proxy adds a Proxy-Authenticate header to 401 containing the proxy-initiated security challenge. This example features a Digest challenge so as to illustrate the usage of the new qop-options value "auth-extd-int".

```
SIP/2.0 401 Unauthorized
...
WWW-Authenticate:...
Proxy-Authenticate: Digest realm=MOBILEUSR nonce=<anyvalue>,
algorithm=MD5, qop=auth-extd-int
```

F4: The UA re-sends the REGISTER with the authentication response in Authorization header and the Digest response in Proxy-Authorization header.

```
REGISTER sip:server.nortel.com SIP/2.0
...
Authorization:...
Proxy-Authorization: Digest username=<user>, realm=MOBILEUSR,
nonce=<anyvalue>, uri=<SIP-URI>, response=<message-digest>,
cnonce=<value>, nc=1, qop=auth-extd-int
```

F5: The outbound Proxy forwards the REGISTER after verifying the Digest response and stripping off the Proxy-Authorization header.

```
REGISTER sip:server.nortel.com SIP/2.0
...
Authorization:...
```

F6: The 200 OK to the REGISTER arrives at the Proxy. The Proxy inserts the Authentication-Info header in the 200 OK for authenticating the message to the UAC [Note: this assumes that the authentication of the REGISTER message at the Proxy in step F5 is successful].

```
SIP/2.0 200 OK
Authentication-Info: nextnonce=<anyvalue>, qop=auth-extd-int,
rspauth=<message-digest>, nc=1
```

F7: A subsequent INVITE request to a user Bob, must also be integrity protected - the UA pre-emptively adds the Proxy-Authorization header.

```
INVITE sip: bob@server.nortel.com SIP/2.0
...
Proxy-Authorization: Digest username=<user>, realm=MOBILEUSR,
nonce=<anyvalue>, uri=<SIP-URI>, response=<message-digest>,
cnonce=<value>, nc=2, qop= auth-extd-int
```

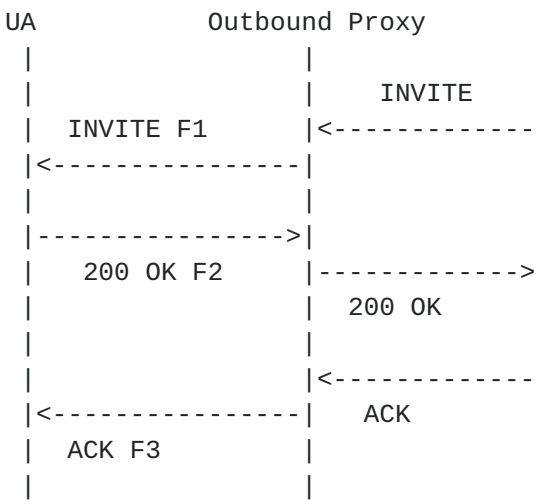
F8: The 200 OK response is forwarded to the UA by the Proxy after inserting the Authentication-Info header.

```
SIP/2.0 200 OK
Authentication-Info: qop=auth-extd-int, rspauth=<message-
digest>, nc=2
```

F9: UA sends an ACK message complete the INVITE transaction

```
ACK sip: bob@server.nortel.com SIP/2.0
...
Proxy-Authorization: Digest username=<user>, realm=MOBILEUSR,
nonce=<anyvalue>, uri=<SIP-URI>, response=<message-digest>,
cnonce=<value>, nc=3, qop= auth-extd-int
```

CASE 2: UA receives an incoming INVITE through the outbound Proxy. The UA and the outbound Proxy has mutually authenticated as described in CASE 1.



F1: The Outbound Proxy receives an incoming INVITE. The Proxy modifies the branch parameter in the top-most Via header, inserts the Proxy-Authorization header containing the Digest credentials and "re-submits" the request to the UAS.

```
INVITE sip: tom@host.nortel.com SIP/2.0
Via: SIP/2.0/UDP server.nortel.com;branch=23ade45.1
  ...
Proxy-Authorization: Digest username=<user>, realm=MOBILEUSR,
nonce=<anyvalue>, uri=<SIP-URI>, response=<message-digest>,
cnonce=<value>, nc=1, qop= auth-extd-int
```

F2: If the authentication is successful, the UAS sends a 200 OK with the Authentication-Info header.

```
SIP/2.0 200 OK
Authentication-Info: qop=auth-extd-int, rspauth=<message-
digest>, nc=1
```

F3: The Proxy inserts the Proxy-Authorization in the incoming ACK message and again "resubmits" the request

```
ACK sip: tom@host.nortel.com SIP/2.0
Via: SIP/2.0/UDP server.nortel.com;branch=23ade45.1
  ...
Proxy-Authorization: Digest username=<user>, realm=MOBILEUSR,
nonce=<anyvalue>, uri=<SIP-URI>, response=<message-digest>,
cnonce=<value>, nc=2, qop= auth-extd-int
```

5 Security Considerations

Most of the security considerations in Section 4 of [RFC2617] still apply except that now we can provide a better level of integrity protection with consequent reduction in risk for MITM attacks. However, since the authentication mechanisms are carried in the challenges in clear-text, bidding-down type of attack is still possible.

6 References

[SIPbis05] Session Initiation Protocol, draft-ietf-sip-rfc2543bis-05.txt
[RFC2617] HTTP Authentication: Basic and Digest Access Authentication, RFC 2617
[3gpp-req] 3GPP requirements on SIP, draft-garcia-sipping-3gpp-reqs-00.txt

7 Acknowledgments

The authors would like to thank Scott Orton of Nortel Networks and Tao Haukka of Nokia for their useful comments and suggestions related to this draft.

8 Author's Address

Sanjoy Sen
Nortel Networks
sanjoy@nortelnetworks.com

Lee Valerius
Nortel Networks
valerius@nortelnetworks.com

Vesa Torvinen
Oy LM Ericsson Ab

vesa.torvinen@ericsson.fi

9 Full Copyright Statement