3GPP TSG SA WG3 Security — S3#20 **S3-010509**

16 - 19 October, 2001

Sydney, Australia

Source: **Nortel Networks, Ericsson, Nokia**

Title: **Integrity Protection: Mechanism for SIP-level solution**

Document for: **Discussion/Decision**

Agenda Item: **7.3, IP Multimedia Subsystem Security**

Abstract

This contribution proposes text for inclusion in Annex C of draft TS 33.203, section C.2 Integrity mechanisms*. The text describes a SIP-level solution for message integrity protection in the 3GPP IMS. The solution involves the use of the HTTP Digest security framework.*

1. Introduction

As was agreed in the September ad hoc meeting of SA3, Annex C has been created in draft TS 33.203 to permit the description of SIP-level solution mechanisms for SIP confidentiality and message integrity protection in the 3GPP IMS. The following text is proposed for inclusion in section C.2 to describe a SIP-level solution mechanism for message integrity protection.

The solution involves the use of the HTTP Digest security framework. Extensions to Digest [1] that are reflected in the description are indicated in bold print. These extensions are being worked within the IETF.

2. Proposed Text for 33.203 Section C.2

[Editors note: There seems to be an unexpected shortcoming in the way SIP provides integrity protection on messages between UE and Proxies. In current SIP, HTTP Digest can be used to partially integrity protect the messages originated by an UE. However, SIP fails to provide integrity for Proxy to UE communication, i.e. for terminating INVITEs, for example. Proxies are not able to add Authorization headers on these messages, thus leaving the messages unprotected.

For the reason above, the headers and field names used in this section may not be final. However, the found inconsistency will probably make it easier for 3GPP to discuss about new SIP level integrity protection schemes with IETF.]

HTTP Digest shall provide integrity protection of SIP signalling between the UE and the P-CSCF, protecting all SIP signalling messages at the SIP level.

The SA that is required for Digest integrity protection shall use the 128-bit integrity key IK generated through IMS AKA, as specified in section 6.1. The integrity algorithm and key are identical for integrity protection applied to messages travelling in either direction. Negotiation of the integrity algorithm to use occurs in the following way: The UE communicates the set of integrity algorithms that it supports to the P-CSCF through the Security-setup header field of the REGISTER message, as described in section 7.2. The P-CSCF selects an algorithm to use from the set of algorithm capabilities common to both the UE and the P-CSCF. The P-CSCF indicates the algorithm to use in the “algorithm” directive of the Digest challenge that is subsequently issued to the UE.

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 “int”. Digest supports integrity protection of the entire SIP message when the “qop-options” directive within the Digest challenge is set to the value “extended-int”. (Use of either of these values of “qop-options” assumes that a context of client authentication has been previously established.) To provide for

protection of the entire SIP message, the P-CSCF shall issue a Digest challenge to the UE specifying the value “extended-int” for the “qop-options” directive.

The message ‘digest’, or message authentication code, is conveyed in the “response” directive of the Digest response. The rules for computing “response” are as described in [1] with the following consideration: if the UE receives a Digest challenge with the “qop-options” directive set to either “int” or “extended-int”, and the associated authentication challenge was an IMS AKA challenge, then the UE substitutes IK for the “password” component of A1 when computing “response=” in the Digest response. The UE sets the “username” component of A1 to a fixed value (e.g., “ims-user”). When sending messages to the UE that are to be integrity protected, the P-CSCF applies the same rules when computing “response”. In this manner, the whole SIP message is always protected.

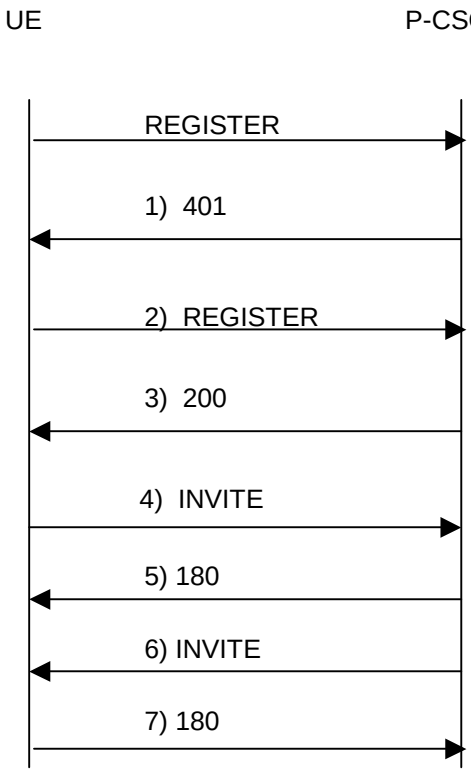
The Digest framework specifies that a server-initiated nonce is to be used by the client as a random number input to the production of the message digest. This nonce, along with a counter that is incremented by either endpoint when sending a message that is to be protected, facilitate anti-replay protection.

In the 3GPP IMS, then, normal operation of the Digest challenge-response mechanism for integrity protection is as follows:

Per RFC 2617, the Digest challenge-related directives are carried in either the WWW-Authenticate or Proxy-Authenticate header fields. The P-CSCF adds a Proxy-Authenticate header field to the 401 that is sent by the S-CSCF (SIP registrar) toward the UE; the Proxy-Authenticate contains the Digest challenge that has been constructed by the P-CSCF.

Per RFC 2617, the Digest response-related directives are carried in either the Authorization or Proxy-Authorization header fields, depending upon which header field carried the corresponding Digest challenge. These directives contain the credentials for the message integrity check. In the IMS context, the UE responds to the initial Digest challenge by adding a Proxy-Authorization header field to the REGISTER toward the S-CSCF (registrar). The UE pre-emptively adds a Proxy-Authorization header field to all subsequent UE-initiated SIP requests. The UE and the P-CSCF add the Authentication-Info header to all SIP responses. **Finally, the P-CSCF adds an Integrity header field to all SIP requests sent toward the UE.**

The simplified message flow shown below illustrates the relevant header fields and contents for the SIP-level integrity protection mechanism. Please note that the message flow contains three cases: a registration (1-3), and two SIP sessions: one UE initiated (4-5) and one UE terminated (6-7).



1) 401 response – this carries both the IMS AKA challenge (from the registrar) and the Digest challenge for integrity protection (from the P-CSCF):

   SIP/2.0 401 Unauthorized

   WWW-Authenticate: EAP <RAND AUTN>

   Proxy-Authenticate: Digest realm=3GPP-IMS nonce=<random-number> algorithm=MD5 qop=extended-int

2) Integrity protection is turned on with the next REGISTER – the integrity credentials are placed in the Digest response:

   REGISTER sip: ... SIP/2.0

   Authorization: EAP <RES>

   Proxy-Authorization: Digest username=ims-user, realm=3GPP-IMS, nonce=<echo-random-number>, uri=<SIP-URI>, response=<message-digest>, cnonce=<value>, nc=1, qop=extended-int

3) The 200 response is also integrity protected – the P-CSCF adds the Authentication-Info header to carry the message digest:

   SIP/2.0 200 OK

   Authentication-Info: qop=extended-int, rspauth=<message-digest>, nc=2

4) A subsequent INVITE request must also be integrity protected – the UE pre-emptively adds the Proxy-Authorization header:

   INVITE sip: … SIP/2.0

   Proxy-Authorization: Digest username=ims-user, realm=3GPP-IMS, nonce=<echo-random-number>, uri=<SIP-URI>, response=<message-digest>, cnonce=<value>, nc=3, qop=extended-int

5) The 180 is integrity protected in the same fashion was the 200 response (message #3):

   SIP/2.0 180 Ringing

   Authentication-Info: qop=extended-int, rspauth=<message-digest>, nc=4

6) An incoming INVITE must also be integrity protected – the P-CSCF adds the Integrity header, which has the same syntax as Proxy-Authorization:

   INVITE sip: … SIP/2.0

   Integrity: Digest username=ims-user, realm=3GPP-IMS, nonce=<echo-random-number>, uri=<SIP-URI>, response=<message-digest>, cnonce=<value>, nc=5, qop=extended-int

7) The UE protects the 180 response by adding Authentication-Info:

   SIP/2.0 180 Ringing

   Authentication-Info: qop=extended-int, rspauth=<message-digest>, nc=6

6. Recommendation

It is recommended that SA3 adopt the text in section 2 of this contribution for inclusion in Annex C of draft TS 33.203.

REFERENCES

[1] "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617