3GPP TSG SA WG3 Security — S3#21 **S3-010664**

Sophia Antipolis, France

27-30 Nov, 2001

Source: **Siemens**

Title: **Problems with the replay protection scheme in the SIP level integrity solution in Annex C of TS 33.203, v070**

Document for: **Discussion**

Agenda Item:

Abstract

The replay protection scheme for the SIP level integrity solution in Annex C of TS 33.203, v070 has not yet been specified in full detail. But it can be seen from the example information flow provided there how it is intended to be used. It is conluded in this contribution that this replay protection scheme (if interpreted correctly by the author) is flawed and may lead to the loss of calls due to the lack of synchronisation of counters at UE and P-CSCF respectively.

For the convenience of the reader, an excerpt from Annex C of TS 33.203, v070 has been appended to this contribution.

Integrity is provided between the UE and the P-CSCF. Replay protection certainly is a requirement on any integrity mechanism in the IMS. The replay protection scheme proposed in Annex C for SIP level integrity seems to be intended to work as follows:

1. There is one counter (let us call it nc_UE) at the UE and one counter (let us call it nc_P) at the P-CSCF. These counters may be preset to 0.
2. When an entity sends a message it increases the counter by one and includes the (new) counter value nc in the integrity-protected part of the message.
3. When an entity receives a message and can verify its integrity it sets its counter to the nc-value in the received message provided the received nc-value is higher than the counter value. Otherwise, the received message is rejected.

The problem with the proposed replay protection scheme can be seen when discussing a modification of the example information flow provided in Annex C of TS 33.203, v070. It goes as follows:

After the reception of message 3 (2xx Auth OK) both counters nc_UE and nc_P have the value 2. In message 4, the UE sends an INVITE which includes replay protection value nc = 3.

Assume now that message 4 gets lost. Then we have nc_UE = 3 and nc_P = 2.
Assume further that the INVITE is resent by UE and the resent message is lost again. (This second assumption is not strictly necessary to show the problem but increases it.) Then we have nc_UE = 4 and nc_P = 2. UE now abandons its attempts to send INVITES. It is unavoidable that INVITE messages may be lost due to e.g. problems with the radio interface or with the P-CSCF.

Next, there is an incoming call and the P-CSCF sends an INVITE to the UE. The P-CSCF sets nc_P = 3 and includes nc=3 in the INVITE. The message will be rejected because nc is less than or equal to nc_UE = 4, hence the UE assumes the message is a replay although, in fact, it is not a replay. A resending of this INVITE with nc=4 would again result in a rejection by the UE for the same reason. This time, the call is unnecessarily lost due to the replay protection scheme.

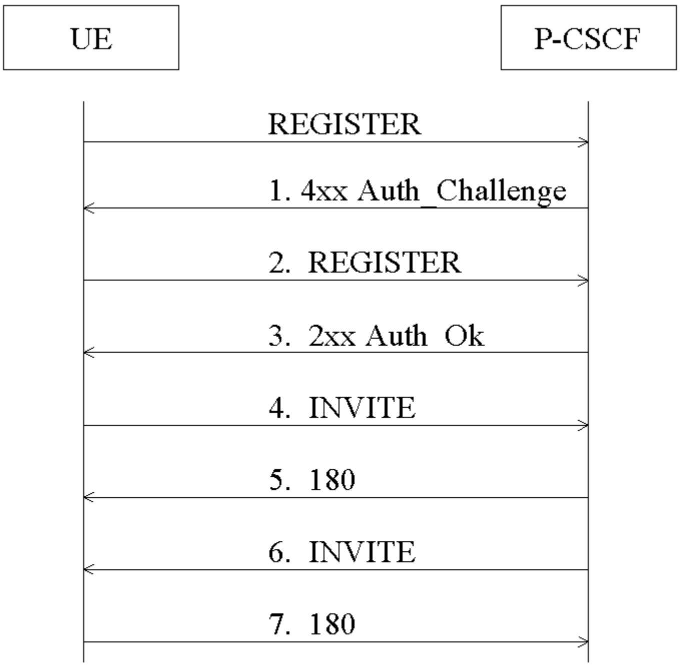
Comparison with replay protection in the http digest:

The above replay protection schemes seems to be modeled after the specification of the http digest authentication in rfc 2617. There, the replay protection scheme seems to work fine because only one side initiates requests, namely the client or the user agent. Then lost messages do not cause problems with the replay protection counter. But in the IMS, both, the UE and the P-CSCF can initiate requests by sending INVITE messages.

Conclusion:

Provided the replay protection scheme implicit in Annex C of TS 33.203, v070 was understood correctly in this contribution, the contribution shows that this replay protection scheme is flawed because it may lead to un unnecessary loss of calls. A remedy may be to use to different counters on each side, one for each direction. The flaw also seems to suggest that care is needed when re-using procedures from an IETF rfc in an IMS context.

The following is taken from Annex C of TS 33.203, v070



1. **4xx 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 4xx Auth_Challenge

   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 2xx response is also integrity protected – the P-CSCF adds the Authentication-Info header to carry the message digest:**

   SIP/2.0 2xx Auth_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 2xx 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

[Editors Note: Further details will be provided on how replay protection is accomplished.]