3GPP TSG SA WG3 Security — S3#29 S3-030422

15 – 18 July 2003

San Francisco, USA

Source:	**Nokia**
Title:	**SA handling regarding to the lifetime of SA**
Document for:	**Discussion&Decision**
Agenda Item:	**7.1 IMS**

Abstract

This paper discusses the contribution [S3-030399] and the companion CR regarding to the lifetime of SA. The implications from keeping or removing the lifetime are both presented. Our proposal is given in the end of the present paper.

1. SA lifetime discussion in CN LS, S3 specification and [S3-030399]

CN and CN1 have sent S3 two LSs regarding to lifetime of SA. [S3-030399] raises the discussion. Under the discussion of Question 4, [S3-030399] has commented that the lifetime has 2 functions: 1) is for the temporary SA before the authentication is confirmed, and 2) is the own lifetime to each established SA.

We agree that the temporary lifetime 1) should be kept for the security reason. This could be accomplished by a state machine in P-CSCF, such as a timer to trigger the deletion of those un-established SAs. The timer could be based on SIP in-build timer for transaction. Since it is much shorted time for SA in temporary phase than in established phase, using such a short timer would save more resources in P-CSCF.

On the other hand, we feel the long lifetime 2) is not necessary. When there is critical failure in S-CSCF the SA lifetime is not useful anyway, because the S-CSCF would need re-boot, re-set registration of all users and all existing SAs would be deleted anyway.

For the benefit of lifetime 2) pointed by [S3-030399] that “limited lifetime, as this makes it very clear to which point a particular key can be used to protect data”, we are doubtful to the statement. In fact this seems to introduce confliction to SIP usage. See section 2 to address this issue in detail.

Under question 3 discussion, [S3-030399] raises a concern that “The CN1 proposal also seems to mandate the P-CSCF holding registration lifetimes. This is not mandated anywhere else and, in our view, the potential benefits from such a requirement do not warrant the expected storage and processing overhead”.

We feel this does not reflect exactly CN1 proposal. The NOTIFY event is initiated by S-CSCF to indicate registration-status. When there is no IMPUs registered, the NOTIFY shall indicate that to the UE via P-CSCF by giving no IMPUs. So the P-CSCF does not need to hold registration lifetime to each IMPU (24.229).

In the Question 2, CN LS reads: “It is possible that a re-registration results in an end time **earlier** or later than that set by existing registrations, even when this expires value is shorter than one received for a previous registration (e.g. the previous registration may be close to expiry). Can SA3 please confirm that when setting the lifetime of the SA the intent is to utilise the latest end time?” They express such a scenario as depicted as Figure 1 below: the latest end time (1 or 2 in Figure 1) maybe shorter than previously existing registrations. During the rest of the SA lifetime there is no any IMPU registered for that UE.

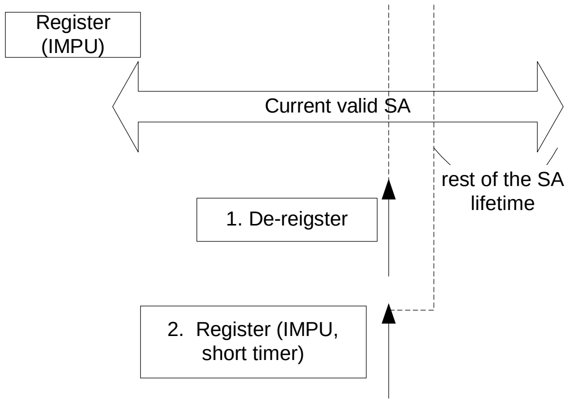

Figure 1: Latest end time shorter/earlier than old SA lifetime

Recall the S3 discussion in previous meeting, it is agreed that SA should be equally long to allow all services subscribed during that registration. The contribution [S3-030399] seems also support such view:” In fact, given that there are no existing registrations there are no SAs hence in this case the two methods are equivalent.”

In [S3-030399], the view seems to confirm that lifetime of the SA is to utilise the latest end time. However it seems taking consideration only the **longer** end time against the SA lifetime. It reads:” The lifetime of the SA must be set to the latest end time to ensure the SA does not expire before all registration have expired as this would make the UE unreachable.” So the suggested text states that the lifetime of the new SAs still depend on **longer** life from either lifetime of old SAs or latest registration timer in the message.

CN1 scenario refers to an old SA, not necessarily the new SA during SA handling. This is further discussed in [S3-030399] under Question 3. It states that S3 does not support deletion of old SA except SA handling or Network initiated authentication. This is also reflected in the companion CR with text:” The P-CSCF sets the expiry time of the new SAs such that they either equals the latest lifetime of the old SAs or it will expire shortly after the registration timer in the message, depending which gives the SAs **the longer life**”.

On the other hand, in the scenario depicted in Figure 1, we feel there is no reason to keep the SA longer, since UE is effectively deregistered. De-registering UE shall send a REGISTER with 0 value in Expire header. In Figure 1 case, it would happen after either moment marked with dash line. So it does not seem to make sense at this stage, that P-CSCF would still need to compare the old SA lifetime with this latest expire header =0, and insisting old SA to the original life length. [S3-030399] seems also support this view with text in the companion CR:” The P-CSCF shall delete all SAs it holds that are associated with a particular IMPI once all the associated IMPUs are de-registered.”

If the lifetime of SA relies on the NOTIFY message sent from S-CSCF, it would be extra gain to delete the SA when no more IMPUs registered.

We feel there are some contradiction on the statement in the discussion paper and the companion CR. Clarification is needed. Based on our discussion, we support the view that SA should be possible to be shortened, and it is easily achieved by NOTIFY method.

2. Transaction completion during SA-handling

The SA handling mandates UE to start using new SA after receiving the successful authentication message (SM12). This has been a problem that we consider as severe. Figure 2 below is a very simple flow copied from RFC3261. There could be many requests between the INVITE and the media session, such as PRACK and UPDATE, see example from 24.228.

```
                  atlanta.com  . . . biloxi.com
                      .     proxy          proxy    .
                    .                             .

           .                                              .
   Alice's  . . . . . . . . . . . . . . . . . . . . . . . .……... . . . . . . . . . . . . . . Bob's
  softphone                                                  SIP Phone
     |                          |                     |          |
     |   INVITE F1                    |                          |              |
     |--------------->                |INVITE F2                 |              | <a complete transaction start>
     |  100 Trying F3                 |------------------------->|INVITE F4|
     |<-------------------------------|100 Trying F5             |----------->|
     |                                |<------------------------ |180 Ringing F6| <= old SA expires
     |                                |180 Ringing F7            |<-----------|
     | 180 Ringing F8                 |<-------------------------|200 OK F9|
     |<-------------------------------|   200 OK F10             |<-----------|
     |   200 OK F11                   |<-------------------------|             |
     |<-------------------------------|                          |             | <a complete transaction ends>
     |                 ACK F12                                                 |
     |------------------------------------------------------------------------>|
     |               Media Session                                             |
     |<=======================================================================>|
     |                 BYE F13                                                 |
     |<------------------------------------------------------------------------|
     |                200 OK F14                                               |
     |------------------------------------------------------------------------>|
```

Figure 2: SIP session setup example with SIP trapezoid from RFC3261

It shows a complete transaction in blue. It seems the consensus in S3 that a transaction should be completed using same SA. The problem is that INVITE session may start at any moment, such as just before expiration of old SA, which possibly leads the old SA lifetime expiration fall into middle of a transaction, as the red sign. As a consequence, the rule cuts the connection forcefully. The implication is that

- The new SA may not established in place due to radio interface error, is used by force.
- Several simultaneous TCP as well as UDP connections must be present during SA-handling. When there are many simultaneous calls/dialogs in future service, the potential parallel TCP/UDP connections is not avoidable, thus not an exceptional case. If one of them used UDP and another TCP, we would end-up having four parallel transport layer connections when re-authentication takes place at the same time. This certainly sets limitations to applications.
- A transaction may not finish within lifetime of a SA. And the method has to be cancelled and restarts again. This is due to the radio interface usage that SIP transaction timers are longer than they're defined in RFC. Though 33.203 can mandate a completion of a transaction with same SA, the lifetime of a SA may not permit the statement.

This problem has been in the 33.203 for a while. We understand [S3-030399] lifetime discussion is not meant for this topic in general, but we see overlapping in two issues. When forcing the track of each SA lifetime and mandating the expiration of them, SIP services initiated in the same dialog request would need to move on to the new SA unconditionally.

The removal of lifetime for old SA would rescue this situation. Old SA will be deleted when UE initially sends request with new SA. It would give flexibility to the extension of the transaction that may last. The other benefit is to remove the clock synchronisation requirement in P-CSCF and the S-CSCF. CN plenary LS suggests that the solution already exists in CN1 specification 24.229 based on SIP basic behaviour RFC 3265 Specific Event Notification. However it is worth of noting that any change of port would still mandate simultaneous TCP or UDP connections.

On the other hand, we see the mandatory requirement of completing one transaction over same SA is still valid.

The other solution is to remove the requirement of 'refreshing' port to differentiate the SA in IPsec level. Rather it is handled by the SIP level knowledge, based on the Security-* headers defined in RFC3329 as well as CN1's specifications. These headers are present in **every UE initiated SIP** method and are different during each re-authentication phase. In the IPsec level, the UE and P-CSCF shall always use the latest SA stored in IPsec SA database. This is supported by the IPsec SA RFC 2401 (p. 30 step 2). There are also products that have implemented the function.

It also brings many benefits in:

1. Keeping same TCP/UDP connection for same sip session. An ongoing SIP session can switch from the old SA to the new SA smoothly without interference to the upper session. Note a 3 way handshaking for TCP establishment is a heavy thing, particular when it is running simultaneously with re-authentication, and an on-going SIP session, the impact is un-neglectable.

2. The transaction can always be finished over same SA.

The two solutions both address the same problem. The similarity is that both allow the usage of old SA to be extendable and flexible. While the first solution requires several connections at a time, the second solution allows smooth handover completely, but may have more security concern. This is because usage of SA relies on SIP knowledge, thus an attacker can spoof the SIP application in P-CSCF protected with a compromised SA. In the first solution this is more stringently required in re-authentication scenario by the new client port used.

There is also concern regarding to the bid down attack against security-agree (RFC3329). We believe this is possible such as in case one of the two algorithms deployed in R5 is broken; but it still requires the attack to obtain a valid session key to pose such attack. And if it really happens, we need new patch to the current specification as well.

3. Proposal

Due to the nature of SIP services, we feel the issues identified in section 2 should be taken seriously, and should be taken into account when deciding about the fate of the SA lifetimes.

Based on our further discussion in section 2 we propose that S3 endorses removal the lifetime of SA and rely on SIP provided mechanism. The attached CR is the concrete text based on the first solution.