**3GPP TSG SA WG3 (Security) Meeting # 104-e S3-212408**

**E-meeting, 16-27 August 2021**

**Source: JSRPC Kryptonite**

**Title:** **Add randomness on the UE side**

**Document for: Approval**

**Agenda Item: 5.5**

# 1 Decision/action requested

***New solution proposal to mitigate the*** ***Identifier linkability attack, SUCI replay attack, SUPI guessing attack and attack on re-synchronization described in TR 33.846***

# 2 Reference

[1] 3GPP TR 33.846 V0.12.0, Study on authentication enhancements in 5G System

[2] Krawczyk, Hugo. "SIGMA: The ‘SIGn-and-MAc’approach to authenticated Diffie-Hellman and its use in the IKE protocols." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2003

# 3 Rationale

The solution addresses key issue #2.1 (LFM attack), key issue #2.2 (SUCI based attacks), key issue #3.2 (SUPI guessing attacks) and key issue #4.1 (AMA attack).

More detailed arguments supporting our concerns can be found below.

This solution proposes the following basic modifications:

* Add randomness from the UE side (RANDMS) and bind all the further 3GPP-AKA protocol messages to both RANDHE and RANDMS values. This simple measure provides security guarantees against all types of replay attacks (key issues #2.1, #2.2 and #4.1) since each time the adversary would try to replay any information, it will always get the same failure cause MAC\_failure.
* In order to prevent SUPI guessing attacks (key issue #3.2) and SUPI check attacks (key issue #2.2, clause 2.2.1.1.1 of TR 33.846 [1]) we suggest using the idea from the solution #2.8 (clause 6.2.8 of TR 33.846 [1]), where the SUCI value is additionally binded to the secret key K via KSUCI. It can be done by means of MAC calculation on the secret key K. This measure guarantees that only legitimate user with the secret key K can calculate valid *Registration Request* message.
* Add new input parameter to currently used functions f1, f5, f1\*, f5\*. In addition to RANDHE, these functions should take as input new argument RANDMS, which is 128-bit random value generated on the UE side.

NOTE 1: To optimize calculations the MAC calculation step can be excluded from the ECIES scheme, as the integrity of the SUCI value is already guaranteed by MAC0 calculation (f0 function)

NOTE 2: To optimize calculations the UE public ephemeral key from the ECIES scheme can be used to generate RANDMS value

Even though this proposal is not (yet) theoretically justified with the use of formal methods, such as provable security framework, the preliminary results can be stated as follows:

|  |  |
| --- | --- |
| **Security properties** | **Our consideration** |
| #2.1 (linkability attack by using error message code) | The adversary is unable to replay old (RANDHE, AUTN) request, because the AUTN value is binded to the randomness from the UE side. If adversary tries to replay the old value, it will cause MAC\_failure regardless of the UE. |
| #2.2 (SUCI replay attack, SUPI check attack) | The adversary is unable to replace SUCI with another chosen SUCI’: it will lead to a MAC failure from the HN side, hence causing Registration Reject message. The adversary is unable to generate a valid SUCI from the SUPI: he does not know the required long-term secret key, hence he can not generate valid MAC. |
| #3.2 (SUPI guessing attack) | The adversary is unable to form valid SUCI from some chosen SUPI: he does not know the required long-term secret key. |
| #4.1 (protection of SQN during AKA re-synchronisations) | The adversary is unable to obtain correct SQNMS ⊕ AK\*: it has no control over the generation of authentication vector (RANDHE, AUTN); replacing it with the old one will trigger MAC\_failure due to the binding to the randomness from the UE. |

At the same time, we draw the attention that the presented solution does not require any changes beyond the one used in the already approved proposals from TR 33.846:

|  |  |  |
| --- | --- | --- |
| **Proposed modification** | **Impact** | **Similar modifications in TR 33.846** |
| Add randomness from the UE side (RANDMS) to prevent replay attacks | **Update functions f1, f1\*, f5, f5\***  Add new input parameter RANDMS | Solution #4.1 (TR 33.846, paragraph 6.4.1) – the value MAC-S is added as a function input for f5\* |
| **Expand Registration request message by adding RANDMS** | Solution #4.5 (TR 33.846 paragraph 6.4.5) – AUTS and RANDMS values added  Solution #2.7 (TR 33.846 paragraph 6.2.7.2): «Current usage of ECIES for concealment of SUPI can be expanded… Maximum allowed size of cipher text from concealment of protection scheme output is 3000 digits. SUPI utilizes only few bytes of those … digits» |
| Add MAC calculation on the secret key K to prevent SUPI guessing attacks | **Expand Registration request message by adding MAC0** |
| **Add new function f0**  Add new MAC function f0  to calculate MAC0 = f0(K, RANDMS, SUCI) value for prevention of SUPI guessing | Solution #4.2 (TR 33.846 paragraph 6.4.2) – add new function f6 (symmetric encryption algorithm) |

# 4 Detailed proposals

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of Change 1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

6.0 Mapping of solutions to key issues

**Table 6.0-1: Mapping of solutions to key issues**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Key Issues** | | | | | | | |
| **Solutions** | #1.X | #2.1 | #2.2 | #3.1  \*) | #3.2 | #4.1 |  |  |
| **Solutions for anchor keys security** |  |  |  |  |  |  |  |  |
| No solution so far |  |  |  |  |  |  |  |  |
| **Solutions for resilience against identifier linkability** |  |  |  |  |  |  |  |  |
| #2.1: Handling of Sync failure by AUTS encryption |  | x |  |  |  | x |  |  |
| #2.2: Encryption of authentication failure message types by UE with new keys derived from K\_AUSF |  | x |  |  |  | x |  |  |
| #2.3: Unified authentication response message by UE |  | x |  |  |  |  |  |  |
| #2.4: MAC-S based solution |  | x |  |  |  | x |  |  |
| #2.5: Encryption of authentication failure message with SUCI method |  |  | x |  |  |  |  |  |
| #2.6: Certificate based encryption of unicast NAS message |  | x |  |  |  | x |  |  |
| #2.7: Mitigation against the SUCI replay attack |  |  | x |  |  |  |  |  |
| #2.8: Assuring SUCI generation by Legitimate SUPI owner using KSUCI |  |  | x |  |  |  |  |  |
| #2.9: MAC, SYNCH failure cause concealment |  | x |  |  |  |  |  |  |
| Solution to Key Issue #2.2: SUCI replay |  |  | x |  |  |  |  |  |
| Solution #2.11: Mitigate the SUCI replay based on UE's public key |  |  | x |  |  |  |  |  |
| Solution #2.Y: Adding randomness and MAC calculation on the UE side |  | x | x |  | x | x |  |  |
| **Solutions for availability aspects of SUCI usage** |  |  |  |  |  |  |  |  |
| Solution #3.1: Mitigation of SUPI guessing and SUCI replay attack using long term key |  |  | x |  | x | x |  |  |
| Solution #3.2: Adding Check Value behind SUPI to mitigate the SUPI guessing attacks |  |  |  |  | x |  |  |  |
| Solution #3.3: Mitigation of SUPI guessing attack |  |  |  |  | x |  |  |  |
| **Solutions on re-synchronisation in AKA** |  |  |  |  |  |  |  |  |
| #4.1: Using MACS as freshness in the calculation of AK |  |  |  |  |  | x |  |  |
| #4.2: Using symmetric encryption function to protect SQN during a re-synchronisation procedure in AKA |  |  |  |  |  | x |  |  |
| #4.3: SQN protection by concealment with SUPI in USIM |  |  | x |  | x | x |  |  |
| #4.4: SQN protection during re-synchronisation procedure in AKA |  |  |  |  |  | x |  |  |
| #4.5: AUTS SQNMS solution for 5GS |  |  |  |  |  | x |  |  |
| #4.6: Using time-based or partly time-based SQN generation |  |  |  |  |  | x |  |  |
| #4.7: SQN protection by concealment with SUPI with f5\* |  |  | x |  | x | x |  |  |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of Change 1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of Change 2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

### 6.2.Y Solution #2.Y: <Adding randomness on both sides to mitigate all replay-attacks and assuring SUCI generation by legitimate entity using MAC calculation on secret key>

#### 6.2.Y.1 Introduction

The solution is declared to address key issue #2.1 (LFM attack), key issue #2.2 (SUCI based attacks), key issue #3.2 (SUPI guessing attacks) and key issue #4.1 (AMA attack).

#### 6.2.Y.2 Solution details

The steps of the proposal are as follows (see Figure 1):

1. During the primary authentication procedure, the UE sends the *Registration Request* message with a REQUEST value which is generated as follows (the REQUEST value is sent instead of SUCI):

REQUEST = SUCI, RANDMS,MAC0,

where RANDMS is a 128-bit random number, MAC0 = f0 (K, SUCI, RANDMS) is a key-based MAC function.

1. After receiving the *Registration Request* message, the HN de-conceals SUCI to obtain SUPI and checks the MAC0 value using the secret key K and RANDMS.
2. If MAC is incorrect, the HN sends the *Registration Reject* message and terminates the authentication procedure.

If MAC is correct, the HN sends the *Authentication Request* message with an authentication vector AV = (RANDHE, AUTN) calculated as follows:

* Generate RANDHE;
* MAC = f1(K, SQNHE, RANDHE, RANDMS);
* AK = f5(K, RANDHE, RANDMS);
* AUTN SQNHE ⊕ AK, MAC.

1. After receiving the *Authentication Request* message, the UE de-conceals SQNHE and checks the MAC value:

If MAC is correct and SQNHE is in correct range, authentication protocol continues in the standard mode.

If MAC is incorrect, UE sends the *Authentication Response* message with MAC\_failure to HN.

If MAC is correct and SQNHE is not in a correct range, the UE sends the *Authentication Response* message with SYNC\_failure and AUTS value calculated as follows:

* MAC\* = f1\*(K, SQNMS, RANDHE, RANDMS);
* AK\* = f5\*(K, RANDHE, RANDMS);
* AUTS SQNMS ⊕ AK\*, MAC\*.

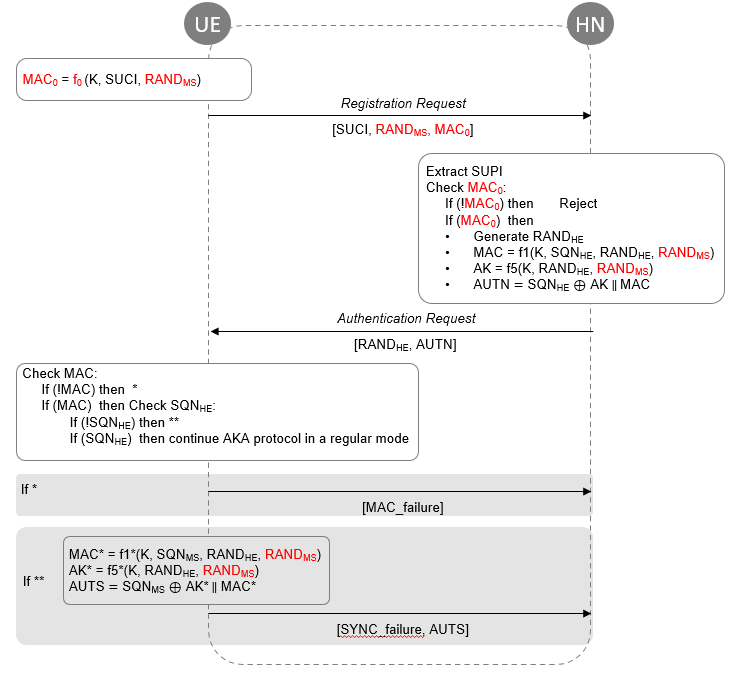


Figure 1

#### 6.2.Y.3 Evaluation

The solution suggests changes to the calculation of functions: f1, f1\*, f5, f5\*. Also, the ECIES scheme should be slightly updated. This variant of the protocol follows the accepted paradigm of constructing SigMa-like protocols [2], on which many modern protocols (e.g., TLS and IPSec) are based.

We strictly demand binding to the random numbers of **both** parties, as this is required for the authentication property.

If no binding is done, we cannot guarantee the freshness of the answer, and this can potentially lead to the various sophisticated (but practical!) replay attacks.

The proposed scheme has impact on the USIM and UDM.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of Change 2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*