**3GPP TSG SA WG3 (Security) Meeting SA3#102bis-e draft\_S3-211050-r3**

**e-meeting, 1-5 March 2021** *revision of S3-211050*

**Source: China Mobile**

**Title: Solution to Key issue #2.2 in TR 33.846**

**Document for: Approval**

**Agenda Item: 2.5**

# 1 Decision/action requested

***The pCR proposes a solution to key issue#2.2 regarding SUCI Replay, and is kindly asked to be approved by SA3.***

# 2 Reference

[1] TR 33.846 v 0.10.0, Study on authentication enhancements in 5G System.

# 3 Rationale

The SUCI replay attack shall be mitgated as it can raise various potential threats. In this contribution, we propose a method based on the public key of UE to address the SUCI replay attack.

Usually there are two ways to mitgate the replay attack: counter-based and nonce-based scheme. For the counter-based scheme, the receiver determines whether the message is a replay attack based on the value of counter. The complexity of the this scheme is that sender and receiver have to maintain the counter and deal with the asynchronization of the counter. The nonce-based scheme is relatively simple, the senders generate a random number that is never used (nonce) and sends it algong with the message, the receiver checks the nonce and determins whether it is a replay attack. There is no need to maintain the state between the sender and receiver.

A UE converts its SUPI into the SUCI by using ECIES scheme. For this, the UE generates a random number as the private key, and derives the public key from the private key. The UE’s public key is included in the SUCI and transimitted to the UDM. To identitfy whether the SUCI message is a replay message or not, the UDM checks whether the UE’s public key has been presented in the database of the UDM. If the UE’s public key can not be found by the UDM, the UDM can determin that the received SUCI message is not a replay message, and store this public key in the database; otherwise, the UDM is confronted with a replay attack, and acknowleges UE with an error message.

UE's public key is a random number that varies within the range of [0, 2*n*], where *n* is the length of the UE's public key. The probability that the UE generates the same public key is 1/2n. The length of the UE's public key is usually greater than or equal to 256 bits, so it is plausible to apply UE's public key to address the SUCI replay attack. The proposed scheme is actually a nonce-based scheme since it used the public key in the SUCI as the nonce.

In order to realize the reliable transmission of SUCI from UE to AMF, TS 24.501 defines the SUCI retransmission mechanism for UE, using two timers: T3519 (60 seconds) and T3510 (15 seconds), as shown in the figure below.



Figure SUCI retransmission mechanism

After the UE generates the SUCI, it starts timers T3519 and T3510, stores the SUCI, and sends the SUCI to the AMF. If AMF receives SUCI, it will pass it to AUSF without cheacking whether the SUCI is a replay message or not. AUSF forward the SUCI to UDM. If the UE does not receive a response from the AMF within the timer T3510, it will resend the stored SUCI to the network. In extreme cases, the UE may send the same SUCI to the network five times, that is, it is possible that the UDM will receive the same SUCI five times. After the timer T3519 expires, the UE generates a new SUCI, stores it, and starts a new round of SUCI transmission. This means that the same SUCI received within 60s in the UDM should be seen as a valid SUCI rather than a replay attack, otherwise the SUCI retransimission mechanism in the UE will not take effect.

In order to identifiy a real SUCI replay attack and not break the SUCI retransimission mechanism, UDM applies the message queue of the storage data unit (DU), together with the database storing public key, to determine whether the received SUCI is a replay attack. The structure of Data Unit (DU: Data Unit) is DU={SUCI, SUPI, Timestamp}. During the timer TUDM whose value is set to T3519, the DU will be used as a comparison tag to identify whether the received SUCI is a SUCI retransmitted by a legitimate user. That is, the SUCI received subsequently will be compared with the SUCI in the DU. If the same, the received SUCI will not be discarded and will be processed further. If they are different, the public key in the SUCI is applied to judge whether the SUCI is a replay attack. The concrete steps are as follows.

1. After receiving SUCI, UDM sets timestamp to the time when SUCI is received, and searches for SUCI in the message queue of the stored DUs. If SUCI is found and the timestamp in the corresponding DU is within TUDMtime, UDM gets SUPI in the corresponding DU. UDM obtains the long-term key of the UE according to SUPI, generates an authentication vector and returns it to the UE.
2. UDM removes the DUs whose time is before 60s in the message queue according to the timestamp.
3. If UDM does not find SUCI in the message queue, it extracts the UE’ public key from the received SUCI.

UDM searches for the public key corresponding to the SUCI in the database. If found, it confirms that the SUCI is a replay attack, and the connection is interrupted; if it is not found, it confirms that the SUCI is not a replay attack. Then UDM deconceals SUCI to obtain SUPI. Based on the long-term key of the UE corresponding to the SUPI, an authentication vector is generated and returned to the UE. UDM constructs the data unit DU={SUCI, SUPI, Timestamp} and moves it into the message queue.

The components in the UDM used for anti-replay attack are shown in the following figure.





Figure 1 Components in the UDM

The proposed method does not impact on current standardized SUPI protection scheme. It also does not need any change on the UE and the serving network. The only change required is that the UDM has to store the received UE's public key after determining that the SUCI is not a replay message, and manage the message queue..

The proposed method prevents the linkablity attack as it can identify the replay attack without returning the RAND and AUTN to the attacker. The possible DoS attack is also mitgated as it does not invoke the computation heavy ECIES algorithm before determining whether the SUCI is a replay message or not.

The proposed method does not have an impact on the SUCI retransimission mechanism in the UE.

# 4 Detailed proposal

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

### 6.2.X Solution #2.X: Mitigate the SUCI replay based on UE's public key

#### 6.2.X.1 Introduction

This solution addresses key issue #2.2: SUCI replay attack.

#### 6.2.X.2 Solution details

In order to identifiy a real SUCI replay attack and not break the SUCI retransimission mechanism, UDM applies the message queue of the storage data unit (DU), together with the database storing public key, to determine whether the received SUCI is a replay attack. The structure of Data Unit (DU: Data Unit) is DU={SUCI, SUPI, Timestamp}. During the timer TUDM whose value is set to T3519, the DU will be used as a comparison tag to identify whether the received SUCI is a SUCI retransmitted by a legitimate user. That is, the SUCI received subsequently will be compared with the SUCI in the DU. If the same, the received SUCI will not be discarded and will be processed further. If they are different, the public key in the SUCI is applied to judge whether the SUCI is a replay attack. The concrete steps are as follows.

1. After receiving SUCI, UDM sets timestamp to the time when SUCI is received, and searches for SUCI in the message queue of the stored DUs. If SUCI is found and the timestamp in the corresponding DU is within TUDM time, UDM gets SUPI in the corresponding DU. UDM obtains the long-term key of the UE according to SUPI, generates an authentication vector and returns it to the UE.
2. UDM removes the DUs whose time is before 60s in the message queue according to the timestamp.
3. If UDM does not find SUCI in the message queue, it extracts the UE’ public key from the received SUCI.

UDM searches for the public key corresponding to the SUCI in the database. If found, it confirms that the SUCI is a replay attack, and the connection is interrupted; if it is not found, it confirms that the SUCI is not a replay attack. Then UDM deconceals SUCI to obtain SUPI. Based on the long-term key of the UE corresponding to the SUPI, an authentication vector is generated and returned to the UE. UDM constructs the data unit DU={SUCI, SUPI, Timestamp} and moves it into the message queue.

Editor’s Note: It is FFS whether the proposed scheme is subject to the poisoning attack

NOTE: How many public keys are stored in UDM is up to the operator’s policy.

The components in the UDM used for anti-replay attack are shown in the following figure.





Figure 1 Components in the UDM

#### 6.2.X.3 Evaluation

The proposed method does not have an impact on current standardized SUPI protection scheme. It also does not need any change on the UE and the serving network. The only change required is that the UDM has to store the received UE's public key after determining that the SUCI is not a replay message.

The proposed method prevents the linkablity attack as it can identify the replay attack without returning the RAND and AUTN to the attacker. The possible DoS attack is also mitgated as it does not invoke the computation heavy ECIES algorithm before determining whether the SUCI is a replay message or not.

The proposed method does not have an impact on the SUCI retransimission mechanism in the UE.

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

#