International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 7 Issue 2 Ver. V || February 2018 || PP.42-50

Enhancing Privacy of Paging Procedure in LTE

Abdulrahman Muthana[1], Mamoon M. Saeed[2], Abdul AzimAbd Ghani[3], Ramlan Mahmod[3]

[1](*Faculty of Computer Science and Information Systems, Thamar University, Yemen*)
[2](*Dept. of Telecomm& Networks, University of Modern Sciences, Yemen)*
[3](*Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Malaysia)*
Corresponding Author: Abdulrahman Muthana

ABSTRACT :*The mechanisms adopted by cellular technologies for user identification allow an adversary to collect information about individuals and track their movements within the network; and thus exposing privacy of the users to unknown risks. Despite efforts have been made by Long Term Evolution LTE toward enhancing privacy preserving capabilities, LTE does not eliminate the possibility of user privacy attacks. LTE is still vulnerable to user privacy attacks. This paper includes an evaluation of LTE security architecture and proposes a security solution for the enhancement of paging procedure privacy in LTE. The solution is based on introducing of frequently changing unrelated temporary mobile subscriber identifiers (TMSI) used for identification. The scheme provides secure and effective identity management in respect to the protection of user privacy in LTE during paging process. The scheme is formally verified using proVerif and proved to provide an adequate assurance of user privacy protection.*

KEYWORDS*-LTE (Long Term Evolution), TMSI (Temporary Mobile Subscriber Identifier), Linkability, Traceability, Paging.*

I. INTRODUCTION

Recently protecting user privacy in cellular networks has received an increasing interest more particularly in Long Term Evolution (LTE) cellular technology. LTE cellular technology, which is recently proposed by the Third Generation Partnership Project [1], has security enhancements comparing to its predecessors: Global System Mobile Communication (GSM) and Universal Mobile Telecommunications System (UMTS). LTE security architecture is substantially different from its predecessors in GSM and UMTS and offers a range of security features.

To protect the user identity privacy, the LTE allocates various different temporary identities such as Global User Temporary Identifier (GUTI), temporary mobile subscriber identifier (TMSI), and cell radio network temporary identifier (C-RNTI) to a single user equipment (UE) at different levels of LTE network architecture for different services. The UE can use these identities instead of the International Mobile Subscriber Identifier (IMSI) to identity itself. This strategy aims at eliminating the IMSI exposure problem and mitigating user identity privacy attacks.

Despite this security strategythe LTE still has a number of security flawsanduser is still vulnerable to privacy attacks[2-6]. Temporary identifiers remain unchanged for amount of time sufficient for hacker to track the user and are transmitted in clear. For example, TMSI will not be changed within certain tracking area and that the paging messages are not encrypted[3].

In this paper we analyze the paging privacy issues in LTE. We also present a solution for enhancing paging privacy. The solution provides a high level of user anonymity and unlinkability within LTE network through introducing of frequently changing unrelated temporary mobile subscriber identifiers (TMSI).User identity privacy is preserved with minimal modifications at network architecture. The proposed solution design strategy aims at keeping the messaging system away as much as possible from the modifications and changes. We believe that this solution could be fit easily in current cellular network architecture.

Our main contribution is the demonstration how a particular realization of an existing normal protocol employed by LTE can be obtained that substantially enhances the user identity privacy preserving capability in LTE. The privacy enhancement is obtained with minimal changes on the network entities (i.e., Mobile Management Entity MME and UE) and with no changes in the message system. The second contribution is an extensive theoretical study on paging privacy in LTE.

The remainder of this paper is organized as follows: Section 2 describes paging procedure privacy issues in LTE. A summary of related work is given in Section 3. Section 4 and 5 present the proposed solution and its security analysis and Section 6 concludes.

II. PAGING PROCEDURE ISSUES IN LTE

The LTE network uses the paging procedure to locate an idle UE in order to deliver a service to it (e.g., an incoming call, SMS message). MME locates an idle UE as per tracking area TA basis and sends the paging request message to everyevolved eNodeB (eNB) within a particular tracking area. The transmitted paging message shall contain the identity of one or more UEs. The UEs intended by the paging request are normally identified by temporary identities (*S*-TMSIs) in order to provide anonymity of the UEs [3].

Once a UE finds its TMSI in the paging message, it establishes a dedicated channel to allow the delivery of the service (responding to incoming call or receiving the SMS). It should be noted that TMSI will not be changed within a certain tracking area TA and that the paging message are not encrypted.

The possibility of initiating a paging request for a specific TMSI allows an attacker to check for the presence of a particular UE within a specific area. Assume that an attacker initiates several calls to a specific user within the user's tracking area and monitors the paging channel to obtain several sets of TMSIs that have been paged by the eNB. The attacker could reveal the TMSI of the intended user by intersecting the sets of TMSIs.

III. RELATED WORK

Many research works have discussed privacy in LTE and suggested solutions for protecting privacy in LTE. A number of researches have focused on user identity privacy in LTE [2-13, 16, 19]. Paging and location privacy have also been highlighted in [3- 6, 15, 17,18, 20,21]. In this paper, we restrict ourselves to the closest related works (i.e., the research works attempt to solve paging privacy issue in LTE).

Several researches investigate security issues with the paging procedure. The authors in [9] proposed to encrypt the paging request using a shared session key, which it called unlinkability key. This key is generated by applying a new one-way keyed function f to the long-term shared key KIMSI, and a random number rand contained in the paging request. This key should be used for privacy preserving purposes only. Furthermore, the encrypted request message is required to include a random challenge *chall* and a sequence number SQN.

The network stores the random challenge and checks it against the one sent by the UE in the paging response. The aim of the SQN is to ensure freshness of the paging request and avoid replay attacks. The SQN should be handled in the same way as in the AKA protocol. A UE receiving a legitimate IMSI paging request should discard it if the SQN is not in the correct range. The use of this procedure should still be kept minimal in favor of the use TMSI whenever possible to avoid burdening the signaling communication with cryptographic operations. In fact, each UE has to decrypt and check all the received IMSI paging to determine if it is the recipient.

In [21] the authors study problems with information leakage in the paging procedure. They also provide a solution by using a physical layer identification scheme. The scheme could be complementary and would not replace the need for enhanced privacy in the other signaling procedures. The scheme for preserving privacy within the paging procedure proposes using a function that has a tag as an outcome. As input to that function, the UE's temporary ID would be utilized. During the paging period of a subscriber, instead of transmitting TMSI, the corresponding tag would be inserted. However, any correlation among the tags for different users should not exist. An interesting point is that in this case, the transmission power of the signal need not to be at such a level that the receiver could decode it. The receiver should only be able to detect the signal to be able to ensure if the user has been paged or not. This results in saving energy. Despite the efficiencies of this approach, one drawback of it is the need to change the physical layer procedure that would lead to changing the hardware, which might be costly.

The authors in [3] suggest a solution to mitigate privacy attacks against the paging procedure. In order to mitigate these attacks, they recommend sending a hashed value of TMSI identifier assigned to the user. The only condition is that along with those pseudonyms, a random value like a nonce or a time stamp should be utilized as input to the hash function in order to change the hashed value of the pseudonym after each calculation.

IV. THE SOLUTION

To protect against paging related attacks, it is required that the allocated M-TMSI identifiers cannot be easily linked with the IMSI nor tracked. To meet these requirements, our method replaces the fixed M-TMSI identifier with frequently changing M-TMSI identifiers. The serving network (MME) first allocates a range of M-TMSI identifiers and delivers the range to the UE. We refer to the range as *R*. Each range *R* is a pair(*S*, *L*), where *S* is 32bit value representing the starting point of *R* (i.e., the first M-TMSI value in *R*) while *L* is 16bit value representing the length of the range *R* (thus, the maximum length of in *R* is 65536). The UE, interprets the allocated range *R* as follows: *S* is the smallest M-TMSI value in *R* while the value (*S+L*) is the largest M-TMSI in *R*. The UE also understands that the valid M-TMSI used for paging UE should lie between *S* and *S+L*. Subsequently, whenever it wishes to page the UE, the MME randomly generates a fresh M-TMSI value between *S* and *S+L*, and includes it within the paging message to be transmitted to the UE. On the receipt of the paging

message, the UE verifies whether the received M-TMSI lies between*S* and *S*+*L*. If the received M-TMSI is not within the correct range, the UE discards the paging message request; otherwise it may responds and initiates a service request procedure.

The proposed solution makes changes on the way of creating and allocating M-TMSI identifiers used for paging UEs in order to ensure unlinkability of the subscribers. It enhances the characteristics of the allocated M-TMSI identifiers such that: (1) the allocated M-TMSI is independent from the IMSI, the GUTI, and from any previous allocated M-TMSI. An adversary, who is monitoring the paging channel, cannot correlate the intercepted M-TMSIs with each other nor correlate them with a particular UE, (2) the M-TMSI is computationally unpredictable, (3) the M-TMSI has a limited life time, (4) it is changed frequently, (5) it is not reused, (6) there are no collisions in the allocation areas, and (7) the concerned UE can easily verify if its identifier is included in the paging message.

While the design of the method we restrict ourselves to achieve the security objective with a minimal modifications at the two network nodes (i.e., the MME and the UE) without any modification imposed on any other network node. The computational power and storage capabilities of both the MME and the UE are considered. As, we will see below, the method ensures the unlinkability of UEs with a minimal modifications at the MME and the UE. Besides that, it introduces an affordable computation overhead at the MME and a negligible computation overhead at the UE. The method protects against paging-related attacks at a minimal cost and thus it can be easily integrated with the current mobile technology.Moreover, the proposed solution requires no changes on the messaging system. The values*S* and *L*are includedin normal messages the serving network SN transmits to the UEduring communications between UE and SN.

Themain steps of the allocation and the delivery of M-TMSI range to the UE are shown in Figure 1 and explained below:

1. Once it receives an attachment request from a UE, the MME allocates M-TMSI range *R*(*S*,*L*) to the UE (the details of the allocation procedure will be elaborated later on in Section 4.1).
2. The MME computes *L'* as a result of XORing*L* and first half of *S*, and forwards *L'* along with the attachment request to Home Subscriber Station HSS.
3. The HSS generates random token *RAND* and embeds *L'* into *RAND*.The calculation of authentication vector AV proceeds as in normal authentication and key agreement AKA procedure and transmitted to the MME.
4. The MME forwards the authentication request to the UE and completes with the UE the AKA procedure steps.
5. If authentication succeeds, the UE extracts *L'* from *RAND* and get back *L* by XORing*L'* with first half of *S*, which is received in GUTI message.
6. Finally, the UE stores *S* and *L* for the purpose of paging procedure.

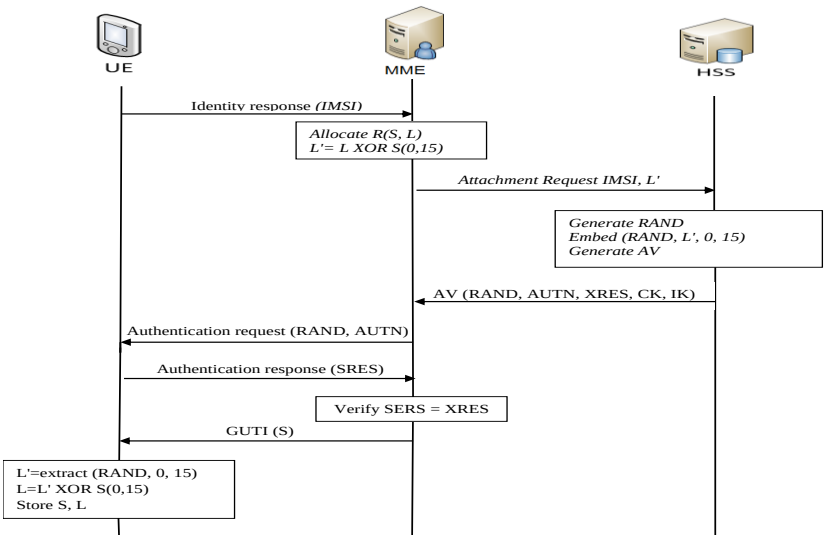

Figure 1: The main steps of allocating and delivery of M-TMSI range to the UE

4.1 The MME

The MME storage is extended with one table called P-table that stores M-TMSI information of all UEs in its service area. The P-table structure is shown in Figure 2. A tuple of P-table stores M-TMSI information of one UE and comprises a set of fields: IMSI, *S*,*L*, *T* and *V*. The IMSI holds the UE's IMSI identifier. *S* and *L* are the start and the length of M-TMSI range being allocated to the UE. The *T* is the M-TMSI value that was last transmitted for paging the UE and while the *V* is the M-TMSI value that was last received from UE for initiating service request.

S	*L*	*STATUS*
S1	L_1	1
...	...	…
Si	L_i	1
...	...	…
SK	*LK*	0
...	...	…
Sn	*Ln*	0

(a) M-TMSI Pool

IMSI	*S*	*L*	*T*	*V*
$IMSI_1$	S_1	L_1	*T*1	V_1
…	...	...	…	…
…	...	...	…	…
IMSIi	*Si*	*Li*	*Ti*	*Vi*
…	...	...	…	…
…	...	...	…	…
$IMSI_K$	S_K	L_K	T_K	V_K

(b) P-table

Figure 2: (a) M-TMSI Pool (b) P-table

The MME also keeps a list of M-TMSI ranges called M-TMSI-pool. The M-TMSI-pool, as shown in Figure 2, is a table with three columns: *S*, *L* and *STATUS*. *S* and *L* store respectively the beginning and the length of M-TMSI ranges. The *STATUS* against each range indicates whether the range is free for the use or not. The value 0 in *STATUS* indicates that the corresponding range is free for the use. The allocated range will have 1 in the corresponding *STATUS*.

For a successful operation of the proposed scheme, a number of algorithms are used at MME side:

A) Initialize-Pool(): initializes M-TMSI-pool with the M-TMSI ranges boundaries (Figure 3). The major steps of the algorithm are as follows:

Input: limits of range length *min* and *max*
1: Set Avail← 2^{32}
2: Set Stop← 0
3: *Set S* ← 0
4: While Avail ≥ *min* do
5: create a random *L* between *min* and *max* (*min* ≤ *L* ≤ *max*)
6: if Avail <*min* then
7: *L*= Avail
8: end if
9: *S* = Stop;
10: Stop = Stop +*L*
11: create a new record at M-TMSI-pool
12: insert a tuple (*S*, *L*) into the new record at M-TMSI-pool
13: Avail =Avail – *L*
14: end while

Figure 3: M-TMSI Pool initialization algorithm

1. The M-TMSI sequence (2^{32} unique M-TMSI values) is partitioned into a set of non-overlapping partitions called ranges *R*. Each range *R*hasa length lies between minimum and maximum boundaries referred to as*min* and *max*. It is up to the operator to decide the values of *min* and *max*.

2. Once, a range *R* is created, its boundaries are stored in *S* and *L* fields of a particular record at M-TMSI-pool. The field *S* stores the first M-TMSI value in *R* whereas the field *L* stores the length of *R*. Initially, all M-TMSI ranges are marked as free for the use (i.e., *STATUS* is set to 0 for all M-TMSI ranges).

B) Allocate(IMSI) allocates M-TMSI range *R* from M-TMSI-pool to the UE (Figure 4). The MME runs *Allocate* algorithm to allocate M-TMSI ranges whenever a new UE is moving into the MME's service area. The major steps of the algorithm are as follows:

1. The MME selects a fresh not-in-use M-TMSI range *R*from the M-TMSI-pool, updates the corresponding STATUS to 1 and associates *R* with IMSI of the requesting UE. If no free M-TMSI range to allocate to the UE because all ranges at M-TMSI-pool are allocated, the MME chooses an allocated M-TMSI range. The candidate M-TMSI range, which is already in use by some other UEs, should cause no M-TMSI collisions in the tracking area the UE is in. To avoid M-TMSI collisions, the MME selects an M-TMSI range allocated to an UE whose

Tracking Area List (TAL) does not overlap with the TAL of the concerned UE and reuses M-TMSI range for the concerned UE.
2. A new record is inserted into the P-table in which the IMSI identifier, the range boundaries *S* and *L*, and the initial values of *T* and *R* are also stored.

Input: the IMSI identifier of UE ($IMSI_{UE}$)
1. If there are free M-TMSI ranges at M-TMSI-pool then
2. select one free M-TMSI range *i* (*S*,*L*) from
3. M-TMSI-pool
4. update *STATUS* = 1
5. allocate UE ← (*Si*, *Li*)
6. else
7. let range x (*Sz*, *Lz*) be an M-TMSI range
8. allocated to a $UE_z$where:
9. TAL(*UE*) ∩ TAL(*UEz*) = φ
10. allocate UE ← (*Sz*, *Lz*)
11. End if
12. create a new blank record at P-table
13. insert the tuple (*IMSI*UE, *S*UE*L*UE, 0, 0) into the new record

Figure 4: M-TMSI range allocation algorithm

C) DeAllocate(IMSI): removes the UE's entry from P-table and de-allocates the M-TMSI range allocated to the UE, and if possible frees up the de-allocated M-TMSI range. The MME runs DeAllocate algorithm whenever an existing UE is moving out of the MME's service area (Figure 5).
The major steps of the algorithm are as follows:
1. The MME locates the UE's IMSI at P-table, gets the endpoints of the allocated M-TMSI range *R*, and then removes the corresponding entry from P-table.
2. After removing the UE's entry from P-table, the MME again locates *R* at the P-table and verifies whether *R* is currently in use by another UE or not. If no match is found, the MME frees up the range *R* by updating the associated *STATUS* field at M-TMSI-pool to 0 indicating that *R* is now a free for the use, but if a match is found the MME refrains from freeing up *R* since *R* is still in use by other UEs.

Input: the IMSI identifier of UE
1. locate P-table for the IMSI
2. get the boundaries of the M-TMSI range *S* and *L*
3. remove the UE's record from P-table
4. locate P-table for another copies of *S* and *L*
5. if no match is found then
6. locate M-TMSI-pool for the record that contains *S* and *L*
7. update *STATUS* = 0
8. end if

Figure 5: M-TMSI range de-allocation algorithm

D) ReAallocate(IMSI): This procedure allocates a UE an M-TMSI range different from the currently allocated M-TMSI range. The MME usually runs the *ReAllocate* algorithm during the UE's movement within the service area when the UE moves into a new tracking area which is not in the tracking area list TAL registered in the UE. *ReAllocate*algorithm can also be invoked by the MME at regular time intervals (Figure 6).It is worth mentioning that the newly allocated range has the same length as the currently allocated *R*, which is currently allocated to the UE. This is because the MME sends only *S* to the UE during the Reallocation procedure. The major steps of the algorithm are as follows:
1. The MME allocate a new range *R* whose *L* is the same as existing range *R* currently allocated to the UE.
2. The MME initiates GUTI relocation procedure and sends new *S* to the UE, which will replace its *S* with the newly received *S* and recalculate the boundary value (*S*+*L*).

Input: the IMSI identifier of UE
1. Allocate a new range *R* whose *L* is the same as existing range *R* currently allocated to UE
2. Initiates GUTI relocation procedure

Figure 6: M-TMSI range re-allocation algorithm

***E) Paging-UE(IMSI)*:** generates an M-TMSI value and includes it in the paging message to be transmitted to the concerned UE. The MME runs Paging-UE algorithm, when it wishes to page an idle UE (Figure 7). The major steps of the algorithm are as follows:

Input: the IMSI identifier of UE
1. locate the UE's IMSI at the P-table
2. get the associated S_{UE}, L_{UE}, T_{UE}, and V_{UE}
3. generate a random fresh M-TMSI identity (M_{MME}) where:
4. $S_{UE} \leq M_{MME} \leq (S_{UE} + L_{UE})$ and
5. $M_{MME} \neq T_{UE}$ and
6. $M_{MME} \neq V_{UE}$
7. update the UE's record at the P-table by setting
8. $T_{UE} = M_{MME}$
9. embed a copy of *M* within the paging message
10. continue with the normal paging procedure

Figure 7: Paging-UE algorithm

1. Locates the UE's IMSI at P-table and generates a random fresh M-TMSI value M_{MME} that satisfy the conditions: (1) it is within the M-TMSI range allocated to the UE, (2) it is different from the M-TMSI was last sent to the UE, and (3) it is different from the M-TMSI was last received from the UE.
2. Invoke the normal paging procedure that takes *M*MME as input and completes the normal paging process steps.

F) Validate(IMSI, M-TMSI): Once it receives a request from a UE including M-TMSI identifier, the MME runs Validate-Request algorithm to validate the M-TMSI included in the request. Depending on the validation results, the MME decides whether to respond to the request or not. If the M-TMSI is valid (i.e., TRUE is returned from Validate-Request algorithm), the MME is confirmed that the request has been initiated by a genuine UE and thus responds to the incoming request, otherwise the incoming request is discarded (Figure 8). The major steps of the algorithm are as follows:

1. Locate P-table to find the M-TMSI range that contains the incoming M-TMSI. If no match is found, the request is discarded; otherwise the algorithm proceeds with the next step.
2. Check that the included M-TMSI is different from the M-TMSI were last sent to the UE, and from the M-TMSI were last received from the UE.

Input: IMSI, M-TMSI value M_{UE} received from UE
Output: Boolean value stored in variable Valid
1. Valid = *false*
2. Locate P-table for M-TMSI range where:
3. $S_{UE} \leq M_{UE} \leq (S_{UE} + L_{UE})$
4. If a particular range is found
5. If ($M_{UE} \neq T_{UE}$ and $M_{UE} \neq V_{UE}$)
6. Valid = TRUE
7. End if
8. End if
9. return Valid

Figure 8: Validate request algorithm

4.2 The UE

The proposed scheme requires that the UE is extended to store four 32bit M-TMSI values: S_{UE}, L_{UE}, $T_{UE}$and V_{UE}. The $S_{UE}$and $L_{UE}$are used to store the boundaries of the M-TMSI range supplied by the MME. The $T_{UE}$and the V_{UE} are used to store the M-TMSI identities that were last transmitted and received by the UE respectively. The functionalities of the UE with respect to GUTI relocation, paging, and service request procedures should also be modified. However, the modifications at the UE's functionalities are minimal and introduce a negligible computational overhead at the UE. The security of the above mentioned procedures are effectively enhanced as we can see in the next sections.

For a successful operation of the proposed scheme, a number of algorithms are used at UE side:

A) Receive GUTI messages: As described earlier, the UE receives *L*UE, the length of the M-TMSI rangeallocated for itincluded in *RAND* token during Authentication and key Agreement AKA procedure and then supplied with starting point of M-TMSI range, *S*UE, within the GUTI messages. Subsequently; the UE receives the starting point of M-TMSI range, S_{UE},included in the GUTI message in the following occasions:

- After a successful run of AKA authentication procedure

- After Inter-MME handover request
- After a successful TAU request
- The UE can be scheduled to receive an arbitrary GUTI message at regular time intervals given by the serving network
- The UE can also be provided with the capability to request a fresh M-TMSI range.

B) Receive a Paging Message: Once it receives a paging message request from the MME, the UE verifies that the embedded M-TMSI value (M_{MME}) within the paging message is within the correct range (i.e., M_{MME} is between *S*UE and *L*UE) and is different from the M-TMSI was last sent by UE,T_{UE}, and the M-TMSI was last received by the UE, *V*UE. If so, the UE responds by initiating a service request and updates its MUERCV to the newly arrived M_{MME} identity; otherwise the message request is discarded. The M-TMSI validation procedure is presented as algorithm in Figure 9.

Input: paging message including the M-TMSI (MMME) received from MME	
1:	if ($S_{UE} \leq M_{MME} \leq S_{UE} + L_{UE}$)
2:	if ($M_{MME} \neq T_{UE}$ And $M_{MME} \neq V_{UE}$)
3:	update $V_{UE} = M_{MME}$
4:	initiate a service request
5:	else
6:	discard the request
7:	end if
8:	else
9:	discard the request
10:	end if

Figure 9: Paging message validation algorithm

C. Initiate a Service Request

As described earlier, a UE initiates a service request if it confirms that it is a target of the paging message received from the serving network. To initiate a service request, the UE first generates a random new M-TMSI value M_{UE} and, embeds it within the request message to be sent to the serving network, and updates S_{UE} to F_{UE}. The service request steps are described by Algorithm presented in Figure 10.

1:	create a random fresh 32bit value M_{UE} such that:
2:	$S_{UE} \leq M_{UE} \leq (S_{UE} + L_{UE})$,
3:	$M_{UE} \neq T_{UE}$, and
4:	$M_{UE} \neq V_{UE}$
5:	update $T_{UE} = M_{UE}$
6:	initiate service request

Figure 10: Service Request algorithm

The condition in step 2 is to allow the UE to be uniquely identified by the MME, while the conditions (in steps 3 and 4) are to ensure that the transmitted M-TMSI is different from the M-TMSIs that were last exchanged. The goal of the conditions (in steps 3 and 4) is to eliminate any possibility of replay attack.

V. ANALYSIS

In current LTE architecture, TMSI identifier is assigned to a user in order to uniquely identify the user during the paging process. Whenever, MME wishes to page an idle mobile phone, it includes the UE's TMSI within the paging request message and transmits it to the UE. The problem is that the assigned TMSI remains for a duration that is sufficient for an attacker to link the TMSI used within the paging messages requests to the user permanent identity IMSI. Thus, the existing paging procedure does not protect against user linkability attack. The proposed enhancements of the characteristics of TMSI identifiers and the TMSI allocation procedure adds security performance to the paging procedure and protects against linkability attacks. Using random TMSI identifier each time a UE is paged guarantees that an observer cannot link the paging requests to the same user.

5.1 The Key Features

A) Minimal Computation Overhead: This solution places the majority of computation overhead on the HSS while a minimal computation is placed on the UE. Since the computation power of the HSS is unlimited, we claim that the overhead is negligible. We also claim that the computation overhead at the UE is negligible.

B) Minimal system Impact: The solution does not require a changes in the messages and the messaging system, which makes it transparent to the intermediary networks.
C) Compatible with LTE architecture: The solution can fit easily in the current architecture as it imposes minimal modifications on the network parties.

1.2 Security Analysis

In this section, the security of the solution is analyzed in terms of unlinkability and untraceability.

A) User Unlinkability
Linkability refers to possibility of the linking between permanent identity and temporary identities of users. The proposed scheme eliminates user linkability and protects user against tracking attack through provides unlinkability of LTE network subscribers. The UE is assigned a sequence of temporary identities (M-TMSIs, and C-RNTIs) M-TMSIs instead of a fixed M-TMSI that can be tracked and linked to a specific UE.
B) User Untraceability
Traceability refers to the possibility of identifying past of identity requests and responses of the same subscriber. The proposed scheme eliminates user traceability and protects user against tracking attack through enhancing the characteristics of, and the allocation procedures of, the pseudonyms (TMSIs). The allocation procedure of TMSI pseudonyms adopted by the presented scheme prevent tracking of the user. The user is assigned ranges of TMSIs and upon each request message, a random pseudonym TMSI is chosen from within the range. Moreover, each pseudonym is utilized only once by respective network parties. This makes it difficult for an observer to identify the identity requests and responses destined the same user as the pseudonyms exchanged in the network, from the observer's view point are random and unrelated. Consequently, the observer cannot identify the past identity requests and responses of the same user, and the untraceability of the user is provided.

VI. CONCLUSION

This paper presents a convenient solution to the problem of protecting paging procedure privacy of the in LTE network. The paging procedure privacy is maintained through a secure identification scheme that allows a user to be uniquely identified by the network while the user remains anonymous within the network, and thus prevents adversaries from being able to track and identify the user. The solution derives its advantages from the fact that it is compatible with current standards of LTE cellular technology and easily fits within the current architecture. The presented solution preserves the paging procedure privacy in LTE and ensures user untraceability and unlinkability with minimal modifications at both the network and the UE and low computation overhead on the part of the network and negligible computation overhead on the part of the UE.

APPENDIX A FORMAL VERIFICATION OF ENHANCED PAGING PROTOCOL

The main result of this project is that the proposed Paging indeed enforces secure paging and preserves privacy (i.e., unlinkability). The underlying idea behind the proof is that an attacker (outside observer) sees no difference in the output of two executions of protocol that they differ only in user identities. The proVerif[14] is used for verifying that proposed solution enforces secure paging and preserves privacy The proof proceeds using through the notion by using observational equivalence.

```
Enhanced Paging Protocol:
event accept TMSI (bitstring, bitstring).
freenet:channel.
free A: bitstring.
free B: bitstring.
(* constant values *)
const PAGING_REQ: bitstring.
const PAGING_RSP: bitstring.
let MS(id: bitstring, otmsi: bitstring) =
in(net, (=PAGING_REQ, tmsi_in:bitstring));
out(net, (PAGING_RSP, otmsi)).
let SN(id: bitstring, itmsi:bitstring) =
(*new itmsi: bitstring;*)
out(net, (PAGING_REQ, itmsi));
in(net, (=PAGING_RSP, otmsi:bitstring)).
process
 ((! (new otmsi1a: bitstring;  new otmsi1b: bitstring;
new itmsi1a: bitstring;  new itmsi1b: bitstring;
(! ((SN(choice[A, B], choice[itmsi1a, itmsi1b])) | (MS(choice[A, B], choice[otmsi1a, otmsi1b])))).
```

REFERENCES

[1]. 3GPP, 3GPP System Architecture Evolution (SAE); Security architecture. 3GPP, TS 33.401, 2013.
[2]. Choudhury H., Roychoudhury B. and Saikia D. K., Enhancing user identity privacy in LTE. In IEEE 11th International Conference on Security and Privacy in Computing and Communications (TrustCom), 2012. p. 949–957.
[3]. HamidrezaGhafghazi, Amr El-Mougy, Hussein T. Mouftah, Enhancing the Privacy of LTE-based Public Safety Networks. In 13th Annual IEEE Workshop on Wireless Local Networks, Edmonton, Canada 2014.
[4]. Bikos A. and Sklavos N., LTE/SAE security issues on 4g wireless networks. IEEE Security and Privacy, 2013. 11(2):p. 55–62
[5]. Seddigh N., Nandy B., Makkar R. and J. F. Beaumont H. F., Security advances and challenges in 4g wireless networks. In Eighth Annual International Conference on Privacy Security and Trust (PST), 2010. p. 62-71.
[6]. Bilogrevic I., Jadliwala M. and Hubaux J. P., Security and privacy in next generation mobile networks: LTE and femtocells. In 2nd International Femtocell Workshop, Luton, UK. Citeseer, 2010.
[7]. Bou A. J., Chaouchi H. and Aoude M., Ensured Confidentiality Authentication and Key Agreement Protocol for EPS. In 3rd Symposium on Broadband Networks and Fast Internet, 28-29 May 2012.
[8]. Xiehua, Li, and Wang Yongjun, Security Enhanced Authentication and Key Agreement Protocol for LTE/SAE Network. In 7th International Conference on Wireless Communications, Networking and Mobile Computing, IEEE, 2011.
[9]. Arapinis M., et al., New privacy issues in mobile telephony: fix and verification. In ACM Conference on Computer and Communications Security, 2012.p. 205–216.
[10]. Muxing Z., Yuguang F., Security Analysis and Enhancements of 3GPP Authentication and Key Agreement Protocol," IEEE Trans, vol. 4, 2005.p. 734-742.
[11]. Køien G. M., Mutual entity authentication for LTE. In 7th International Wireless Communications and Mobile Computing Conference, IEEE, 2011.
[12]. Køien G. M., Privacy enhanced mutual authentication in LTE. In IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2013. p 614–621.
[13]. Fabian van den Broek, RoelVerdult and Joeri de Ruiter, Defeating IMSI Catchers. In CCS '15 Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ACM New York, NY, USA 2015.
[14]. B. Blanchet. Proverif: Cryptographic protocol verifier in the formal model. http://www.proverif.ens.fr/.
[15]. KadhimShubber for Wired magazine. Tracking devices hidden in London's recycling bins are stalking your smartphone. http://www.wired.co.uk/news/ archive/2013-08/09/recycling-bins-are-watching-you. Last accessed May 2015.
[16]. MyrtoArapinis, Loretta Ilaria Mancini, Eike Ritter, and Mark Ryan. Privacy through pseudonymity in mobile telephony systems. In NDSS, 2014.
[17]. SirajDatoo for The Guardian. How tracking customers in-store will soon be the norm. http://gu. com/p/3ym4v/sbl. Last accessed May 2015.
[18]. Balasaheb N. Jagdale,Nileema*B.* Gawande,"Hybrid Model for Location Privacy in Wireless Ad-Hoc Networks", IJCNIS, vol.5, no.1, pp.14-23,2013.DOI: 10.5815/ijcnis.2013.01.02
[19]. Muthana A., Saeed M., "Analysis of User Identity Privacy in LTE and Proposed Solution", I. J. Computer Network and Information Security, 2017, 1, 54-63 Published Online January 2017 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijcnis.2017.01.07
[20]. Forsberg D., Leping H., Tsuyoshi K., and Alanara S., (2007) "Enhancing security and privacy in 3gpp e-utran radio interface," in Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on. IEEE, pp. 1–5.
[21]. Tuan, Ta., and Baras, John. (2013) "Enhancing Privacy in LTE Paging System Using Physical Layer Identification.", Data Privacy Management and Autonomous Spontaneous Security. pp.15-28. Springer Berlin Heidelberg, 2013.

International Journal of Engineering Science Invention (IJESI) is UGC approved Journal with Sl. No. 3822, Journal no. 43302.

Abdulrahman Muthana. "Enhancing Privacy of Paging Procedure in LTE" International Journal of Engineering Science Invention (IJESI), vol. 07, no. 02, 2018, pp. 42–50.