GUTI Reallocation Demystified: Cellular Location Tracking with Changing Temporary Identifier

Byeongdo Hong
KAIST
byeongdo@kaist.ac.kr

Sangwook Bae
KAIST
hoops@kaist.ac.kr

Yongdae Kim
KAIST
yongdaek@kaist.ac.kr

***Abstract*—To keep subscribers' identity confidential, a cellular network operator must use a temporary identifier instead of a permanent one according to the 3GPP standard. Temporary identifiers include Temporary Mobile Subscriber Identity (TMSI) and Globally Unique Temporary Identifier (GUTI) for GSM/3G and Long-Term Evolution (LTE) networks, respectively. Unfortunately, recent studies have shown that carriers fail to protect subscribers in both GSM/3G and LTE mainly because these identifiers have static and persistent values. These identifiers can be used to track subscribers' locations. These studies have suggested that temporary identifiers must be reallocated frequently to solve this privacy problem. The only mechanism to update the temporary identifier in current LTE implementations is called GUTI reallocation. We investigate whether the current implementation of the GUTI reallocation mechanism can provide enough security to protect subscribers' privacy.**

To do this, we collect data by performing GUTI reallocation more than 30,000 times with 28 carriers across 11 countries using 78 SIM cards. Then, we investigate whether (1) these reallocated GUTIs in each carrier show noticeable patterns and (2) if they do, these patterns are consistent among different SIM cards within each carrier. Among 28 carriers, 19 carriers have easily predictable and consistent patterns in their GUTI reallocation mechanisms. Among the remaining 9 carriers, we revisit 4 carriers to investigate them in greater detail. For all these 4 carriers, we could find interesting yet predictable patterns after invoking GUTI reallocation multiple times within a short time period. By using this predictability, we show that an adversary can track subscribers' location as in previous studies. Finally, we present a lightweight and unpredictable GUTI reallocation mechanism as a solution.

I. Introduction

A user's identity is inevitably exposed over the air interface of a cellular network depending on the cellular network design. An "IMSI catcher" has been used to track a user's location by catching the International Mobile Subscriber Identity (IMSI), that is, the permanent identity of the user exposed as plaintext on the air interface. Recently, many studies have focused on how to avoid IMSI catching [18], [19], [23], [29], [34], [35], [42]. The 3rd Generation Partnership Project (3GPP) has recognized this problem and designed cellular protocols to use a Temporary Mobile Subscriber Identity (TMSI) instead of a permanent one in 2G/3G, except in unavoidable situations such as the initial attach [2]. In Long-Term Evolution (LTE) networks, a Globally Unique Temporary Identifier (GUTI) is used. However, the 3GPP standard does not specify guidelines for when and how to update the temporary identity, and it leaves the implementation and update frequency to operators.

Recent studies have shown that the absence of a standard guideline has resulted in the problem of reusing temporary identities [20], [30]. Kune *et al.* showed that reusing TMSI, as in existing IMSI catcher attacks, triggers a security threat that can expose a victim's location in Global System for Mobile Communications (GSM) [20]. They noted that if an attacker calls the victim multiple times, he/she can expose the victim's TMSI on the broadcast channel of the air interface. Note that the attacker uses silent calls, in which he/she hangs up before the victim's phone rings to avoid alerting the victim. If the victim is in the same location area (LA) as the attacker, the same TMSI will appear on the channel each time a call to the victim is repeated. Shaik *et al.* showed that the same attack is possible in Voice over LTE (VoLTE) [30]. Both studies suggested frequent reassignment of identity to solve this problem, because it is difficult to track a user's location if the temporary identity is changed.

In LTE networks, GUTI reallocation is the only procedure available for changing the GUTI. If GUTI reallocation changes the GUTI for each voice call, the existing location tracking attack does not work. According to the 3GPP standard, GUTI reallocation can be invoked when (1) a network triggers a non-access stratum "`GUTI Reallocation Command`," (2) the User Equipment (UE) attaches to the LTE, and (3) a `Tracking Area Update` (TAU) occurs [4]. If GUTI reallocation is performed for each call, LTE may become safer against location tracking. However, simply changing the GUTI is not a complete solution against location tracking. Fundamentally, unpredictable GUTI allocation is required to solve the above-described problem.

To investigate this problem, we collected traces of cellular call flows after invoking GUTI reallocation more than 30,000 times for 28 carriers in 11 countries using 78 Subscriber Identity Module (SIM) cards worldwide. Data were collected during our visits to conferences and project meetings and during our vacations. This dataset was mainly collected to determine whether the GUTI reallocation mechanism is securely designed, implemented, and deployed for different carriers. We invoke GUTI reallocation using Circuit Switched Fallback (CSFB), a circuit switched voice call service provided by many LTE

Network and Distributed Systems Security (NDSS) Symposium 2018
18-21 February 2018, San Diego, CA, USA
ISBN 1-891562-49-5
http://dx.doi.org/10.14722/ndss.2018.23349
www.ndss-symposium.org

carriers worldwide. Because most operators require detachment from and attachment to LTE before and after CSFB, we cause GUTI reallocation by implementing auto-call, that is, we repeatedly place calls and hang up automatically.

We analyzed this dataset carefully for each carrier. First, we note that every pattern was consistent within a carrier. In other words, the patterns we found for GUTI reallocation across different SIM cards from a single carrier remain the same. Out of 28 carriers, we discovered simple patterns in 19 carriers. These patterns include varying length of constant bytes as well as monotonically increasing the sequence of bytes. With such predictable patterns, we could track a victim's location as in the previous work [20]. For nine carriers, we initially could not find any pattern. However, we hypothesized that these remaining carriers might still have problems, because previous studies have revealed that the telecommunication industry's implementations are often ad-hoc in nature [39], [40]. For further investigation, we visited four of these nine carriers after implementing a stress test, in which a voice call was rushed within a short period. Depending on the number of calls, all four carriers showed interesting patterns. For example, a carrier skipped GUTI reallocation for less than 10 rushing voice calls.

In our global-scale measurement analysis, we did not find a single carrier that implemented GUTI reallocation securely. The 3GPP standard body also seems aware of the importance of GUTI reallocation as noted in its technical report [1], which discusses this problem through two issues. Issue #7.1 reviews the study by Shaik *et al.* and notes that poor Mobility Management Entity (MME) implementations or carrier misconfigurations may result in the same GUTIs being assigned. Issue #7.4 covers relatively less important issues, such as the fact that poor implementations of temporary identifiers may lead to subscriber identification. For example, $TMSI = ISMI||Counter$, where $||$ is the concatenation operation; in this case, TMSI reveals IMSI. Issue #7.1 examines repeated GUTIs, whereas Issue #7.4 examines information leakage due to the choice of GUTIs. Our analysis shows that most carriers have already implemented a solution for both Issue #7.1 (having different GUTIs after reallocation) and Issue #7.4 (seeming lack of relation to IMSI). We show that Issue #7.1 is insufficiently handled in the current GUTI reallocation problem.

Possibly owing to the lack of a detailed guideline as well as requirements, carriers and manufacturers have used insecure implementations thus far. In fact, operators tend to skip implementations not specified in the standard for network performance optimization. After analyzing associated 3GPP standards and our dataset carefully, we first present detailed requirements to safely implement the GUTI reallocation mechanism. Based on this requirement, we present a light and unpredictable GUTI reallocation mechanism. The technical report also introduces high-level solutions to this problem, specified as Solutions #7.23 and #7.24 [1]. The solution generates a random GUTI using one of the subscriber keys used for authentication. In Section VII, we explain why such a solution is unnecessary.

This paper is organized as follows. Section II provides background information related to cellular networks. Section III describes related work. Section IV outlines the GUTI reallocation rules for each carrier through global-scale measurement analysis of ID management in cellular networks. Section V examines the impact of stress-testing on identity reallocation. Section VI describes the actual attack performed based on information obtained from the measurement analysis. Section VII provides a solution for privacy leakage related to identity management. Finally, Section VIII concludes this paper and discusses future works.

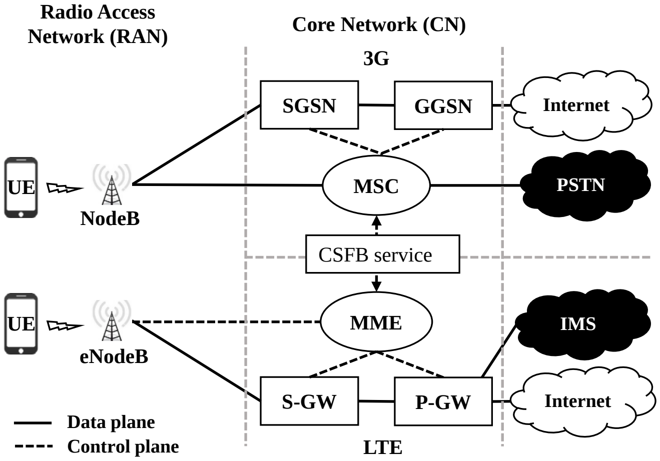


Fig. 1: Cellular Network Architecture

II. Preliminary

In this section, we briefly review the cellular network architecture as well as identities and procedures associated with location leaks.

A. Overview of Cellular Network Architecture

Figure 1 shows the overall architecture of 3G and LTE networks. Both cellular network systems can be divided into three components: (1) UE, (2) a Radio Access Network (RAN), and (3) a core network. The UE represents the user's device used for subscribing and communicating to the network. The RAN comprises a number of base stations, called eNodeB in LTE (NodeB in 3G), that are responsible for radio communication between the UE and the core network. The core network has multiple components that serve voice calls, handover, and data service. The MME is responsible for tracking the location of the UE and managing the connection. The crucial difference between 3G and LTE in the core network is the way in which they deliver data and voice calls. In both networks, data services such as the Internet are provided through packet-switched domains of each network. 3G handles voice calls through a circuit-switched domain and the Public Switched Telephone Network (PSTN), whereas LTE does so using the VoLTE packet-switched domain that is served by an IP Multimedia Subsystem (IMS).

When an LTE carrier does not support VoLTE, it uses CSFB to support a circuit-switched voice call in 3G. When a UE wants to make a voice call in such an LTE network, it first detaches from the LTE network by releasing its resources at the LTE network. Then, it connects to the 3G network through which the call is served. Following the voice call, the UE reattaches its connection to the LTE network.

With regard to the geographical architecture, a service area managed by an eNodeB is called a "Cell." One eNodeB covers

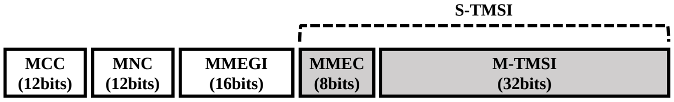


Fig. 2: Structure of GUTI

a group of cells, and the area covered by a group of eNodeB is called the “Tracking Area” (TA; Location Area in 3G). The TA has a unique code called the Tracking Area Code (TAC), and the MME manages the subscriber location by combining the TAC with the MME code (MMEC).

B. *Identifiers in Cellular Network*

The IMSI is a subscriber’s permanent and unique identifier in a cellular network [2]. It is stored in the SIM, and exposing it can lead to security issues such as location tracking and eavesdropping [31], [32]. Therefore, instead of delivering IMSI through the open air interface, carriers use a temporary identifier to hide a subscriber’s identity. Systems older than LTE used TMSI for device identification, whereas LTE uses GUTI. GUTI consists of two parts: a Globally Unique Mobility Management Entity Identifier (GUMMEI) and an MME-Temporary Mobile Subscriber Identity (M-TMSI) (Figure 2). The GUMMEI comprises multiple identifiers for network identification: Mobile Country Code (MCC), Mobile Network Code (MNC), and Mobility Management Entity ID (MME ID). The M-TMSI, consists of a temporary and unique 32-bit value that is used to identify a UE within an MME. The MME assigns a GUTI to a UE when the latter attaches to the network (`ATTACH`) or updates its tracking area (TAU). Thereafter, the UE and MME use the allocated GUTI for identification and communication between the UE and the MME instead of the IMSI. To hide information pertaining to the mapping between subscribers and GUTI, the MME often reallocates GUTI. Note that because the 3GPP standard does not specify the frequency or rules for this reallocation, it is performed using operator-specific configurations [2]. For example, an MME system implemented by Cisco provides two options for triggering GUTI reallocation: time and frequency of access attempts [11]. It can be configured to perform the GUTI reallocation procedure for every N `ATTACH` or TAU requests or periodically every T minutes.

C. *Paging*

Paging is a procedure used when the network wakes up the UE to set up a connection for data service, incoming calls, or Short Message Service (SMS). In such events, the MME lets the eNodeB send an S1AP Radio Resource Control (RRC) paging message to the target UE. We refer to the RRC paging message as simply paging message hereafter. Because the MME does not know the exact eNodeB that covers the target UE when it is idle, it sends S1AP paging messages to all eNodeBs in the TA (paging for voice call). On receiving the S1AP paging message from the MME, an eNodeB broadcasts paging messages through a Paging Control Channel (PCCH). A paging message contains an identifier to designate the UE in the *ue-Identity* field, in which there are two options for UE identification: S-TMSI and IMSI [7]. A UE listens to the paging channel periodically and decodes *PagingRecords* in the paging message. It checks whether the paging message is targeted to itself by comparing identifiers in the *ue-Identity* field. Note that because the paging message is not encrypted, identifiers in the paging message are available to others listening on the same paging channel and located in the same TA. Once a UE in an idle state receives its paging message, it initiates the `Random Access Procedure` to establish a connection with the LTE network. The network then provides the relevant data service or notifies the UE of an incoming call. In the case of paging initiated by an incoming call, the UE receives a paging message and sends a `Service Request` message (`Extended Service Request` in case of CSFB calls).

III. Related Work

This section describes previous works on preserving the privacy of mobile subscribers.

A. *Failure to Maintain Confidentiality of Identity*

Even if the TMSI replaces the IMSI to hide a subscriber’s identity, several studies have shown that an adversary can still compromise the user’s privacy [9], [20], [30]. Attackers exploit the problem whereby carriers do not change the TMSI frequently enough to hide a subscriber’s identity. Given that the TMSI is persistent even after a voice call in GSM, Kune *et al.* suggested that an adversary can locate the subscriber [20]. They made a silent call to generate a paging message from the base station; the recipient is unaware of this call as it is concluded before the phone rings. The adversary monitors the TMSIs in the paging messages by listening to the paging channel (PCCH). If the recipient of the call is attached to the same LA as the adversary, he/she is among the TMSIs monitored in the paging messages once the attacker makes the silent call. The attacker calls several times to check whether the TMSI is monitored at every silent call. Shaik *et al.* showed that the same attack can be mounted in LTE because the GUTI is not allocated often enough [30]. Myrto *et al.* also claimed that TMSIs are not updated frequently [9]. As a solution to this vulnerability, they suggested that carriers should reallocate a new temporary identity to a subscriber more frequently. Broek *et al.* introduced the Pseudo Mobile Subscriber Identifier (PMSI) to defend against the IMSI catching attack; they replaced the IMSI with changing pseudonyms based on SIM information, which were called PMSI [42]. They also tried to keep the subscriber’s identity confidential; however, they did not address a policy of temporary identity reallocation. Furthermore, implementing PMSIs require changing the SIM of all devices. While our work focuses on privacy violations from poorly randomized identifiers in cellular networks, other technologies (including WiFi) are also affected by these issues [24].

B. *Control Plane Analysis in Cellular Networks.*

Lee *et al.* analyzed the impact of each control plane procedure on the 3G core network [21]. Traynor *et al.* showed how a large number of botnets could cause a signaling Denial of Service (DoS) in a cellular network [38]. Arapinis *et al.* introduced a method for tracking the user’s location through `Authentication Failure` [8], and Golde *et al.* introduced a method for hijacking a user’s session through a signaling race condition [15]. Tu *et al.* considered the impact of CSFB calls on network performance in several ways [39], [40], and Li *et al.* introduced a signaling analysis tool applicable to mobile devices [22].

TABLE I: Summary of our dataset of 28 carriers across 11 countries (Carriers indicated by asterisks (∗) are selected for the VoLTE test)

Country	Operator	Date	# of calls	# of USIM	Country	Operator	Date	# of calls	# of USIM
U.S.A.	US-I∗	Nov 2014	601	10	South Korea	KR-I∗	Apr 2015	2,713	8
		Feb 2015	121				Nov 2015	1,041	
		Apr 2015	746				Jun 2017	200	
		Jul 2017	1,700			KR-II∗	Apr 2015	636	4
	US-II	Apr 2015	998	6			Jun 2017	160	
		Jul 2017	1,400			KR-III∗	Nov 2016	100	2
	US-III∗	Jul 2017	200	6	Switzerland	CH-I	Jan 2017	1,000	1
France	FR-I	Dec 2014	99	4		CH-II	Jan 2017	1,500	1
		Sep 2015	418			CH-III	Jan 2017	500	1
	FR-II	Sep 2015	1,055	2	Belgium	BE-I	Feb 2017	800	1
Germany	DE-I	Dec 2014	98	6		BE-II	Feb 2017	600	1
		Aug 2015	982			BE-III	Feb 2017	600	1
		Sep 2015	2,305		Austria	AT-I	Oct 2016	2,000	1
	DE-II	Dec 2014	108	7		AT-II	Oct 2016	2,000	1
		Apr 2015	49			AT-III	Oct 2016	2,000	1
		Aug 2015	497		U.K.	UK-I	Oct 2015	269	1
		Sep 2015	1,297		Spain	ES-I	Jul 2015	282	1
	DE-III∗	Apr 2015	500	4		ES-II	Jul 2015	142	1
		Sep 2015	2,416		Netherlands	NL-I	Dec 2016	2,000	1
	DE-IV	Sep 2015	100	2		NL-II	Dec 2016	2,349	1
Japan	JP-I∗	Apr 2015	337	2		NL-III	Dec 2016	2,349	1

In this study, along with the above-mentioned problems, we demonstrate that carriers are still vulnerable to location tracking even if they change the GUTI frequently. A key requirement of GUTI is generating an unpredictable value to hide the subscriber's identity [5], [6]. We conclude that current operators fail to offer this protection.

IV. Global Measurement of Identity Management

Cellular networks manage the confidentiality of subscribers' identities by allocating a TMSI. As the TMSI is transmitted through a paging channel and is exposed as plain text, the network should refresh it frequently to prevent subscribers from being identified. As mentioned above, this identity management is operator-specific because the 3GPP standards do not specify a detailed mechanism for it. In this section, we investigate operational policies for identity management used by carriers worldwide to check whether they securely manage subscribers' identities. We analyze the large-scale dataset we collected to determine whether the currently deployed GUTI allocation logic is adequate to protect the confidentiality of subscribers.

A. Dataset

Table I shows a summary of the dataset we collected and used in this paper. We gathered signaling messages [1] invoked during the CSFB/VoLTE call procedure to monitor the GUTI value following its reallocation. The data was collected during our visits to conferences and project meetings and on our vacations. Our dataset consists of GUTI allocation data for 39,268 voice calls managed by 28 carriers from 11 countries by using 78 SIM cards during a period of 2 years 9 months (from Nov. 2014 to Jul. 2017). To specifically focus on identity management, all data is recorded without any mobility. We used a simple auto-call tool to automatically dial and disconnect for efficient data collection. This tool can be called and disconnected via a chipset-specific command or an Android debug bridge command, and it is configured to specify the call duration and idle time. Throughout the paper, we denote each carrier by abbreviated symbols for the relevant country and a roman numeral.

B. Methodology

To examine GUTI reallocation, we use the CSFB voice call. On serving a CSFB call at the cellular network, the network switches the UE to the target network system (3G) from the previous network (LTE) and releases all resources belonging to the latter. After serving the CSFB call, the UE proceeds with the `Attach request` procedure, and the carrier reallocates GUTI to the subscriber while proceeding with an `Attach request` through the TAU procedure. As a result, our approach of invoking a CSFB voice call provides an opportunity to test the GUTI reallocation logic of the carriers. Note that for backward compatibility, CSFB is used by many operators and in many countries; therefore, our approach successfully provides hints to examine the GUTI reallocation logic of all operators and countries we investigated. In addition, both CSFB that we use in this work and another method that invokes GUTI reallocation could be used for inferring the reallocation logic.

We analyze the variation in GUTI values in the dataset (Section IV-A) that recorded all GUTI values in TAU messages by continuously generating CSFB calls. Among the various

[1] Control-plane messages

TABLE II: TMSI allocation pattern of carriers

Allocation Pattern	Operators
Assigning the same GUTI	BE-III, DE-II, FR-II, JP-I
Three bytes fixed	CH-II, DE-III, NL-I, NL-II
Two bytes fixed	BE-II, CH-I, CH-III, ES-I, FR-I, NL-III
One byte fixed	AT-I, AT-II, AT-III, BE-I, DE-I

GUTI components, we only focus on M-TMSI, that is, the last four bytes of the GUTI. The remaining parts are the MMEC, public land mobile network code (MCC and MNC), and MME Group ID, which can be considered constant [2]. We also ignore MMEC because monitoring M-TMSI is enough to locate the victim. Note that carriers that use the MME pool might change the MMEC in GUTI reallocation. Even in this case, we noticed that the M-TMSI pattern is not changed. As a result, once the GUTI reallocation pattern is identified, a victim can be tracked by using M-TMSI alone.

C. Identity Allocation Pattern

As proposed in [20], [30], invoking the GUTI reallocation procedure more frequently appears to be a solution to the problem arising from the persistence of GUTI for protecting subscriber confidentiality. However, we observe that the identity allocation logic of carriers is vulnerable in terms of preserving subscriber confidentiality even if the temporary identity is reallocated following each voice call. We verify that most of the carriers considered (19 of 28) have certain noticeable patterns for allocating GUTIs. Note that even if a carrier allocates different GUTIs after every voice call, it can be problematic if the newly allocated GUTIs are sequential or predictable. Therefore, an adversary can track a victim's location by easily inferring his/her identity. Table II shows the GUTI allocation patterns of 28 carriers categorized into four cases.

1) Assigning the same GUTI: BE-III, DE-II, FR-II, and JP-I reassign the same GUTI when reallocating GUTI. They reuse the previously allocated GUTI, but within different patterns from one another. First, FR-II has a procedure to reallocate GUTI to the subscriber; however, it reuses the same value as the previous one, which is retained for every CSFB voice call. Second, BE-III and DE-II periodically allocate the same GUTI values to subscribers. In our experiments, BE-III allocated the same GUTI value from CSFB calls 3–15 times, and DE-II allocated the same values from calls 23–104 times. Finally, JP-I does not reallocate GUTI at all as it adopts Idle-state Signaling Reduction (ISR) technology for control plane optimization. [3]

2) Allocating three bytes as fixed value: We observe that NL-I, NL-II, CH-II, and DE-III allocates different GUTI values; however, the values of three of four bytes in M-TMSI are fixed. We confirm that the positions of the three bytes are fixed in all four carriers: the first, third, and fourth bytes of the four-byte M-TMSI. Figure 3 shows the results of tracing M-TMSI at each voice call in the NL-I dataset [4]. In this case, the first, third, and fourth byte values of the M-TMSI (four bytes) are fixed at `0xF6`, `0xCD`, and `0xB4`, respectively. We also confirm that the other carriers follow the same pattern for over 1,500 CSFB calls. In the case of DE-III, we further verified that the value of the second byte is always smaller than 16. The first four bits of the second byte in the M-TMSI are fixed at `0000` and the remaining bits are less than or equal to `1111`. This implies that the value of 28 bits (three bytes and four bits) and their positions in the M-TMSI of DE-III are fixed. We have not yet found an assignment rule for the second byte. However, the information from the three fixed bytes and their positions is sufficient to reveal the subscriber's identity (Section VI).

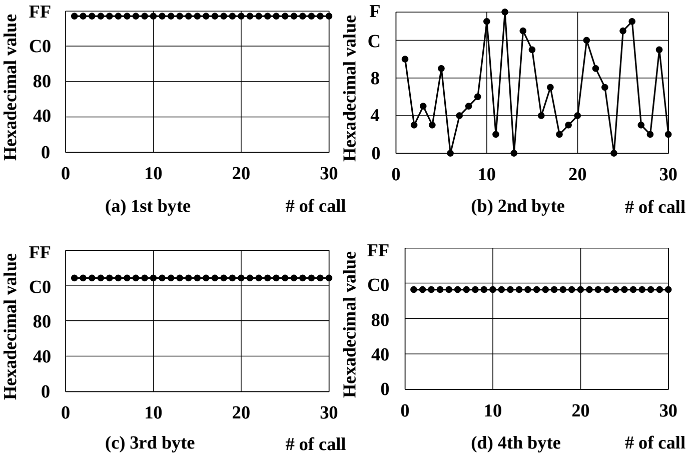


Fig. 3: M-TMSI value of NL-I by number of voice calls

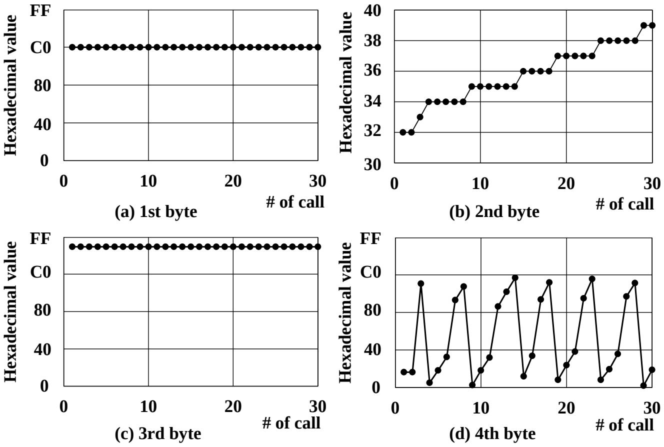


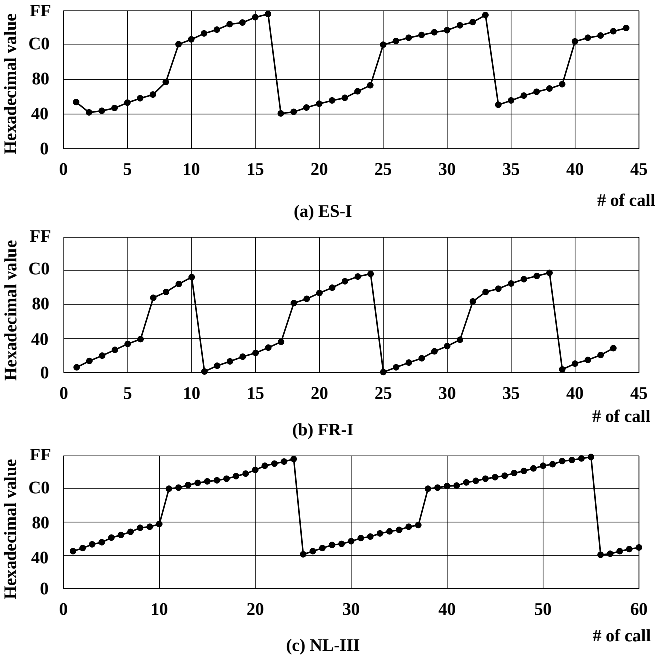
Fig. 4: M-TMSI value of BE-II by number of voice calls

3) Two Fixed Bytes: As in the second case above, BE-II, CH-III, ES-I, FR-I, and NL-III allocate GUTI values, with the values and positions of two bytes in the M-TMSI byte being fixed (first and third bytes). As an example, Figure 4 shows the allocation pattern of the traced M-TMSI values in the BE-II dataset. Note that the fixed value varied across the test cases and carriers; however, the fixed positions are the same. We also make two interesting observations concerning the nonfixed bytes, namely, the second and fourth bytes in the M-TMSI.

[2] The MME Group ID can be changed; however, it is a management code that is usually wider than the tracking area. Therefore, this value does not change in a tracking area.

[3] Note that the attach procedure invokes GUTI reallocation as described. However, the ISR maintains a connection between the user device and the network when the CSFB call is terminated, and therefore the UE maintains the same GUTI.

[4] The figure shows that the range of the x-axis is limited to 30; however, the same rule holds for numbers larger than 30 as well.



(a) ES-I

(b) FR-I

(c) NL-III

Fig. 5: Values of fourth bytes of ES-I, FR-I, and NL-III

First, in the BE-II dataset, the value of the second byte shows a monotonically increasing pattern with 0 or 1 added; the other three carriers (CH-III, ES-I, and NL-III) show the same pattern. Second, the value of the fourth byte shows regularity. As shown in Figure 4(d), a similar pattern is repeated, and the allocated values oscillate. The fourth bytes of ES-I, FR-I, and NL-III are also similar to those of BE-II. Figure 5 shows that the patterns of these three carriers are almost the same. We also note that they have an upper bound while increasing monotonically. After reaching the upper bound value, each provider began assigning the lower bound value. Finally, in addition to the above five carriers, CH-I shows a slightly different tendency, where the first and second bytes are fixed but are changed after being fixed 10–40 times. These new values are maintained for the next 10–40 calls.

4) One Fixed Byte: AT-I, AT-II, AT-III, BE-I, and DE-I use one fixed M-TMSI byte per GUTI reallocation after a voice call. The position of the fixed byte varied across carriers: AT-I, BE-I, and DE-I fix the value of the fourth byte, and AT-II fix that of the first byte. In case of AT-III, we find that the variation in the values of the other bytes is limited. AT-III uses the M-TMSI, the value of the third bytes in which is fixed and the first byte is assigned one of three values.

Summary: Through an analysis of our global dataset, we reveal the internal GUTI allocation logic of operators (Section VI). The overlap of one or two bytes in terms of GUTI value might not seem a significant threat. However, reducing the number of possible GUTI values increases the chances of leaking a subscriber's identity (Section VI-C). In other words, as long as the attacker knows if a carrier follows a particular pattern, by making a number of calls, he/she can identify the victim's GUTI. Section VI discusses and analyzes the effects of our findings on the effectiveness of attacks on user identity.

D. Unresolved Issue of Identifier Reuse in VoLTE

As our calls include VoLTE calls, we also verify whether carriers deploy a defense mechanism against the location leakage attack noted by Shaik *et al.* [30]. The GUTI allocation procedure is triggered in three cases: (a) the UE processes the attach or the update location procedure, (b) the MME of the UE changes, and (c) the GUTI reallocation command is issued [3], [4], [30]. Along with the basic conditions, lessons from previous studies suggest that GUTI (or TMSI) should be altered and reallocated after each voice call [20], [30]. If not, an adversary can perform cell-based user location tracking when combined with paging techniques such as SMS and other data from messenger applications. To examine whether the same vulnerabilities exist in VoLTE calls, we run simple but wide-ranging experiments. For the seven carriers marked with an asterisk ($*$) in Table I, we periodically invoke on average 1,951 VoLTE calls between cellphones and monitor the exposure of their GUTI values to paging messages. Note that for each cellphone, we wait for its RRC connection to become idle to monitor the GUTI.

By examining messages over the control-plane generated by VoLTE calls from these operators, we confirm that the GUTI is still not changed in LTE after all voice calls. Note that our finding is consistent with prior studies that have also shown the consistency of GUTI values [30]; however, we extend the test vector to show that many carriers are still using procedures that are vulnerable.

Remarks. This privacy leakage is mainly caused by a lack of specifications. The relevant standards only mention the case of location change but not cases arising after voice calls. Although VoLTE is being deployed rapidly at present, it tends to consider performance rather than security. We later cover detailed experiments and possible attack scenarios in this context in Section VI.

V. Stress Testing

In the basic experiment described in Section IV, we did not find any noticeable rules for GUTI allocation for nine carriers: DE-IV, ES-II, KR-I, KR-II, KR-III, US-I, US-II, US-III, and UK-I. Among these nine carriers, we physically revisited four carriers and conducted deeper investigations. As a result, we uncovered several vulnerabilities in these four carriers (KR-I, KR-II, US-I, and US-II) through stress testing, where we invoked CSFB calls continuously with a short time gap between calls [5]. We performed two types of stress testing categorized by the gap between CSFB calls: weak and hard. The results of each type of testing and the underlying reasons are different for each carrier. During the stress test, we noticed that the carriers continually allocate the same GUTI values. Table III shows a summary of the stress test results.

A. Weak Stress Testing - Waiting for RRC Idle Mode

We first examined the scenario in which a mobile device receives a paging message as soon as it disconnects from the base station (RRC idle mode). We make CSFB calls and disconnect each before the device rings. We then wait until the UE goes into the RRC idle state, and we generate another

[5] We did not explore the results of stress testing on the remaining carriers.

TABLE III: Stress test results of four selected carriers (✓indicates that GUTI values are fixed in stress test, and ✗ indicates failure to fix GUTI values)

Carrier	Weak Stress Testing	Hard Stress Testing
KR-I	✓	✓
KR-II	✗	✓
US-I	✗	✓
US-II	✓	✓

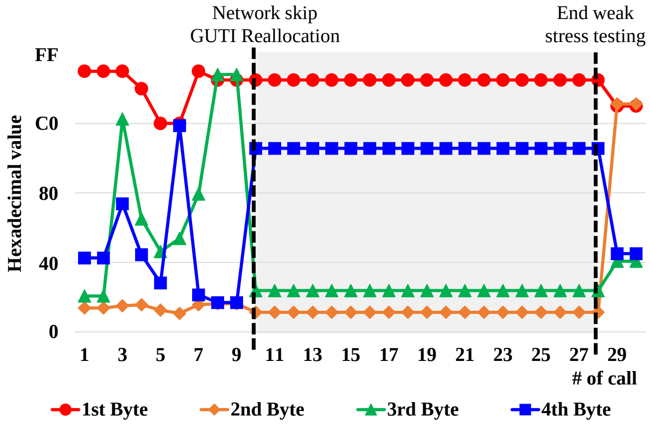


Fig. 6: No GUTI reallocation in weak stress testing in KR-I

CSFB call to trigger a paging message that is exposed to the broadcast channel. We call this method a “weak stress test.” As described in Section IV, a mobile device was in the RRC idle state for a long time because the time interval between each call was long (∼30 s). However, a weak stress test minimizes the period for which a device goes into the RRC idle state. This test makes the base station connect to a device immediately after disconnecting the device.

The hypothesis underlying this experiment is that the MME might try to reduce the control-plane processing to handle overheads to avoid repeating the GUTI reallocation procedure. To verify this, we examine how the network handles our stress test by analyzing the dataset obtained from the weak stress test.

In weak stress testing, US-II and KR-I were found to omit GUTI reallocation altogether after a few unpredictable assignments, causing them to reuse old GUTIs. Figure 6 shows a sample distribution of GUTIs according to the number of CSFB calls in KR-I. When weak stress testing is performed, the first nine GUTI values are allocated without any noticeable pattern. However, from the tenth CSFB call onward, the network did not reassign GUTI but used the same values. Note that this does not mean that the same GUTI values are reallocated by the network but that the `GUTI Reallocation Command` is omitted. This can be easily verified by examining whether the `GUTI Reallocation Command` is included in the `Attach Accept` message from the MME. We confirm this persistent omission of GUTI reallocation by running the weak stress test over ten times. Note that each of stress test requires 30 consecutive calls, as shown in Figure 6. The start time of the omission varies with each trial; however, we observed a persistent GUTI value from the tenth CSFB call onward in all tests. This implies that even without a GUTI allocation rule, we can map the temporary identity to a victim through stress test.

US-II intermittently skips GUTI reallocation; however, it does not omit it two consecutive times. We also confirm this omission of GUTI reallocation by running the weak stress test over ten times, as in KR-I. It omits GUTI reallocation once on the seventh CSFB call. It also dropped the RRC connection immediately after omitting the GUTI; this causes the next paging message to be exposed through the broadcast channel.

B. Hard Stress Testing - Paging without Waiting

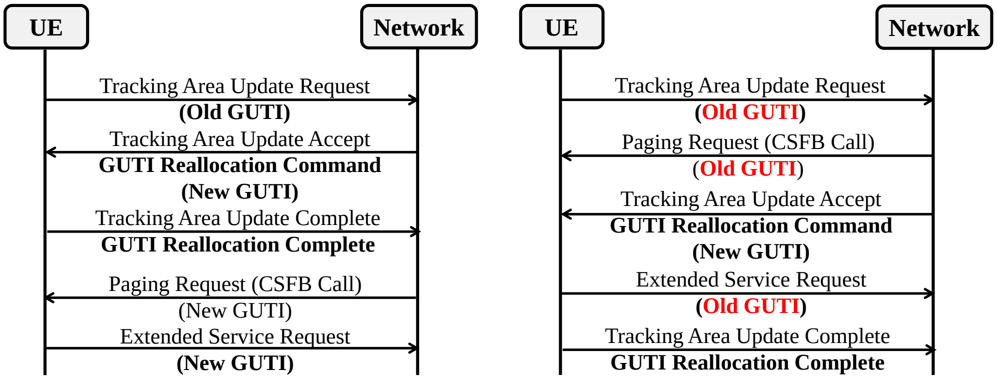
We also run a “hard stress test” that invokes paging without considering the connection between the mobile device and the base station. As in weak stress testing, we call a mobile device and end the call before the target phone rings. The only difference is that the gap between the calls is smaller than in weak stress testing. On making the CSFB call, we wait for the network to send a paging message to the target for 3–8 s and hang up. Note that the waiting time varied with carriers because the time taken to send a paging message to the target is different for each carrier. As soon as we hung up, we started another call without waiting. The hard stress test focuses on dialing quickly without waiting for the RRC idle mode (disconnection between mobile device and base station), and examines how the network reacts in this case.

As in weak stress testing, US-II and KR-I do not reallocate the GUTI in the hard stress test. US-I and KR-II reallocate the same GUTI to the UE, unlike in weak stress testing where different GUTIs are allocated for every CSFB call. However, they have different periods for allocating the same GUTI. Under hard stress testing, the KR-II device continues to use the same GUTI. On the other hand, the US-I device could be made to use the same GUTI two times in succession. These situations arise in the following two cases: (a) procedure omission and (b) a signaling race condition.

1) Omission of GUTI Reallocation: The first reason why UEs have persistent GUTIs is that the carriers omit the GUTI reallocation procedure. We performed 10 hard stress tests, in which the network skipped GUTI reallocation in fewer than 10 calls. If the network omits GUTI reallocation, the device uses the same GUTI value for all services. For example, KR-II omits GUTI reallocation from 3–15 times when hard stress calls are made in our experiment.

One reason for this omission might have been the overhead owing to the processing of signalings on the network side. As some past studies have noted, the network tries to reduce the work needed to handle control messages resulting from a signaling storm [28], [30], [37].

2) Signaling Race Condition: The second reason for allocating the same GUTI value consecutively during the stress test is a racing condition between control messages during the TAU procedure; this consists of the `Extended Service Request` (ESR) and `TAU Accept` message. GUTI reallocation is normally performed with the TAU procedure while handling the CSFB procedure. Figure 7(a) shows the overall flow of TAU and GUTI reallocation. Once the network receives the `TAU Request` message from the UE, it sends the `TAU Accept`


UE
Network
Tracking Area Update Request
(Old GUTI)
Tracking Area Update Accept
GUTI Reallocation Command
(New GUTI)
Tracking Area Update Complete
GUTI Reallocation Complete
Paging Request (CSFB Call)
(New GUTI)
Extended Service Request
(New GUTI)
UE
Network
Tracking Area Update Request
(Old GUTI)
Paging Request (CSFB Call)
(Old GUTI)
Tracking Area Update Accept
GUTI Reallocation Command
(New GUTI)
Extended Service Request
(Old GUTI)
Tracking Area Update Complete
GUTI Reallocation Complete

(a) Normal GUTI Reallocation with CSFB call

(b) GUTI Reallocation and CSFB call collision

Fig. 7: Usage of GUTI in Tracking Area Update and CSFB call

message containing the `GUTI Reallocation Command` with a new GUTI to the UE; this notifies the network of the completion of the procedure by adding `GUTI Reallocation Complete` to the `TAU Complete` message. The UE and the network use the newly assigned GUTI after GUTI reallocation.

However, anomalies occur in US-I and KR-II when the order of the `TAU complete` message and the ESR message is reversed during hard stress testing. If another user calls the UE before the network has sent a `GUTI Reallocation Command` message to the UE, the network sends a paging request to the UE using the old GUTI. The UE receives the paging request and sends an ESR, a message to the network to request a CSFB call service, to the network using the old GUTI, and then, it sends a `TAU complete` message. Moreover, even if the network has assigned a new GUTI to the UE, both the UE and the network continue to communicate using the old GUTI.

This race condition occurs owing to the use of different standards. The standard [4] associated with this process does not provide a clear method to handle this collision: it simply states that both procedures must be carried out but does not specify the order of processing. Therefore, preparation for the collision is implemented differently by each carrier. For example, KR-I set the priority of TAU higher than those of other service requests. Therefore, when the network received a `TAU request` from the UE, it first completed it. In other words, the scenario described in Figure 7(b) does not occur in KR-I because the network that received the `TAU request` does not forward the `Paging request` to the UE. In other words, KR-I avoids this race condition by not forwarding the `Paging request`, because it must be handled only after the UE sends the `TAU accept` message. However, in the case of US-I and KR-II, a signaling race condition happens occasionally because the priority of TAU is not set higher than that of ESR. If another user had known the time at which the UE had sent a TAU request, it could have made a voice call to cause the UE to use its old GUTI.

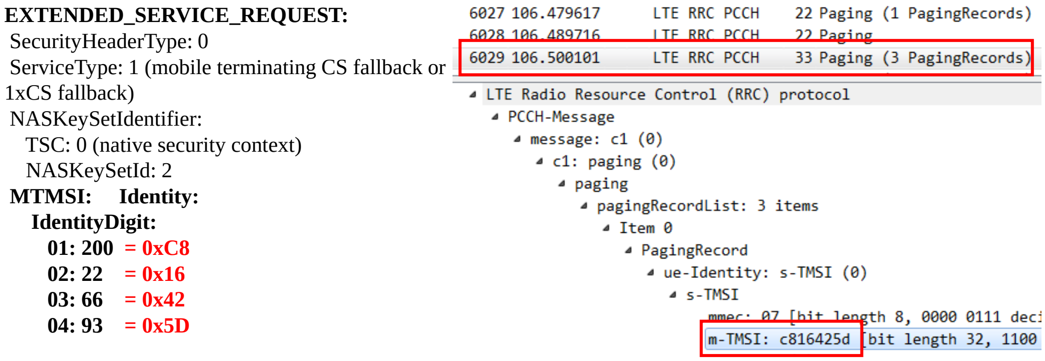
C. Listening to Paging in Stress Testing

One challenge is listening to paging messages during the stress test. If the connection between the mobile device and the network is maintained, the paging message is delivered over the connected session channel rather than the broadcast channel. Note that stress testing is conducted by generating CSFB calls with a short gap before the UE releases the connection. Therefore, one might wonder whether the paging messages generated by stress testing are exposed to the broadcast channel. Carriers want to minimize unnecessary sessions between the network and the mobile device to reduce the load at the core network. For instance, many carriers drop the connection by sending an `RRC Connection Release` message to the mobile device immediately after completing the TAU. This allows us to monitor the GUTI in paging messages during stress testing. In our dataset, US-II, KR-I, and KR-II also drop the RRC connection when they completed `TAU accept` during the stress test; therefore, we could monitor the GUTI value in paging messages through the broadcast channel.

We monitored the control plane message through the diagnostic port in the mobile device. The broadcast channel was monitored using Universal Software Radio Peripheral (USRP) B210 [41] and srsLTE [36], an open-source 3GPP LTE library. Through the stress test, we set up an environment to collect broadcast messages when the network did not reallocate the GUTIs to the mobile device. We placed the LTE broadcast channel receiver (USRP + srsLTE) in a TA to which the mobile device belonged. In this environment, as we continued to call the mobile device, it continued to make `Service Requests` using the same GUTI. Figure 8 shows the GUTI monitored on the (a) mobile device and the (b) broadcast channel during our stress test. The GUTI of the device was fixed at `0xC816425D` (see Fig. 8 (a)), and we captured the paging message delivered to the device in the broadcast channel through our LTE broadcast channel receiver. Figure 8(b) shows that the GUTI with M-TMSI `0xC816425D` broadcasted on the air interface was captured.

D. Ethics

Throughout the stress test experiments along with the global measurement of GUTI reallocation, we care about the possible negative impact on the network and other subscribers. First, one might care about the signaling storm due to stress testing. However, the generated signaling messages are limited to 30 calls between only two devices, which is negligible for the network. We also confirm that the test does not affect other users through an interview with an operator. Second, we only collect GUTIs in the broadcast channel only, which do not provide any private information.



```
EXTENDED_SERVICE_REQUEST:
 SecurityHeaderType: 0
 ServiceType: 1 (mobile terminating CS fallback or
1xCS fallback)
 NASKeySetIdentifier:
   TSC: 0 (native security context)
   NASKeySetId: 2
 MTMSI:    Identity:
   IdentityDigit:
     01: 200  = 0xC8
     02: 22   = 0x16
     03: 66   = 0x42
     04: 93   = 0x5D
```

```
6027 106.479617     LTE RRC PCCH     22 Paging (1 PagingRecords)
6028 106.489716     LTE RRC PCCH     22 Paging
6029 106.500101     LTE RRC PCCH     33 Paging (3 PagingRecords)
◢ LTE Radio Resource Control (RRC) protocol
  ◢ PCCH-Message
    ◢ message: c1 (0)
      ◢ c1: paging (0)
        ◢ paging
          ◢ pagingRecordList: 3 items
            ◢ Item 0
              ◢ PagingRecord
                ◢ ue-Identity: s-TMSI (0)
                  ◢ s-TMSI
                      mmec: 07 [bit length 8, 0000 0111 deci
                      m-TMSI: c816425d [bit length 32, 1100
```

(a) M-TMSI monitored by Device

(b) Paging Message in Broadcast Channel (USRP)

Fig. 8: GUTI exposed to broadcast channel during stress testing.

VI. Attack

In this section, we present our location leakage attack for using our findings from our measurement study. We show how the location of the victim can be leaked even in an environment in which temporary identities are frequently changed by our attack method. First, we describe the overall flow of our attacks by categorizing the voice call technology that the victim uses. We then present the attack procedures and how the characteristics we found are used for the attack. We also verify the exposure of privacy through the experiments. Lastly, we analyze the effectiveness of our attack through the probabilistic analysis and the experimental study.

A. *Methodology*

Figure 9 shows the flow of our proposed attack scenario. The target's phone number is required as a prerequisite for the attack; it can be acquired easily through yellow pages, business cards, or personal homepages [6]. We then need information concerning the target's voice call technology. It is easy to determine through session initiation protocol (SIP) packets whether the target subscriber uses the VoLTE service. For example, if the target device uses VoLTE, it sends a SIP packet for call connection. An adversary can take one of two approaches according to the technology. First, for the VoLTE user, an attacker can exploit the characteristics that GUTI values persist even after the voice call or after establishing an RRC connection. If the attacker fails to leak the location of the victim because the GUTI is not persistent, he/she can perform our attack, called the smart tracking attack.

1) Location Tracking Attack on VoLTE User: In our experiment, all carriers supporting VoLTE do not reallocate GUTI after a voice call. Therefore, an adversary can perform the same attack as that proposed in previous studies [20], [30]. First, the adversary generates a VoLTE silent call that allows the eNodeBs to broadcast a paging message that goes unnoticed by the victim. In our experiments, US and KR carriers take 4 and 2 s, respectively, to ring the phone, implying that the call would not have been noticed by the victim if it had been terminated within these periods; yet, they can trigger a paging message. Once the adversary makes the silent call, he/she listens to the broadcast channel to monitor paging messages and records all GUTIs. Considering the time needed to generate a paging message, the possible GUTIs of the victim after one silent call are limited to a time window within 1–2 s of the call. After several silent calls, if one GUTI value is observed constantly at every call, the attacker can conclude that the GUTI belongs to the victim. Therefore, the victim is located in the same TA [7] as the adversary. Otherwise, if any GUTI appears in every silent call, the victim is not in the TA where the attacker is monitoring the broadcast channel.

2) Smart Tracking Attack: The attack exploiting the persistence of the GUTI does not work with a CSFB call because the GUTI of the UE changes in every voice call in most carriers. Moreover, if the carrier adopts the solution recommended in the literature, namely, reallocating GUTI at every voice call, the previous attack method will be prevented as well. We consider the case in which an adversary can predict the UE's GUTI or have knowledge of the carrier's GUTI allocation pattern. As explored in Sections IV and V, 19 carriers in our dataset have a noticeable GUTI allocation pattern, and stress testing in four carriers showed that GUTI is fixed.

The "smart tracking attack" involves two methods: (1) using the fingerprinted allocation rule and (2) fixing GUTI through the stress test. In this attack, knowledge of the victim's carrier is a prerequisite, as the adversary chooses different methods according to the victim's carrier.

Exploiting GUTI Allocation Rule: For carriers that reveal their GUTI allocation rules (19 of 28 carriers), the attacker uses fingerprinted information. The overall attack scenario is similar to that of the previous attack, in which the attacker makes several silent calls to invoke a paging message through the broadcast channel and monitors GUTIs during the silent call. In the attack, the attacker analyzes all candidate GUTIs for the victim in the paging message at each silent call rather than focusing only on the constantly monitored GUTI. The attacker forms a set of candidate GUTIs by using the reallocation rule. If the monitored GUTI differs from those in the set, but the hex value at the position where the byte value is fixed by the allocation rule is identical to that of the GUTI monitored in the previous silent call, the attacker regards the GUTI as the possible GUTI of the victim and leaves it in the set. Otherwise, the attacker removes the GUTI from the set. The silent call is made repeatedly until the number of remaining candidate

[6] This is the same assumption made by previous works [20], [30]

[7] There are some exceptions that are not TA. For example, KR-I (see Section VI-D).

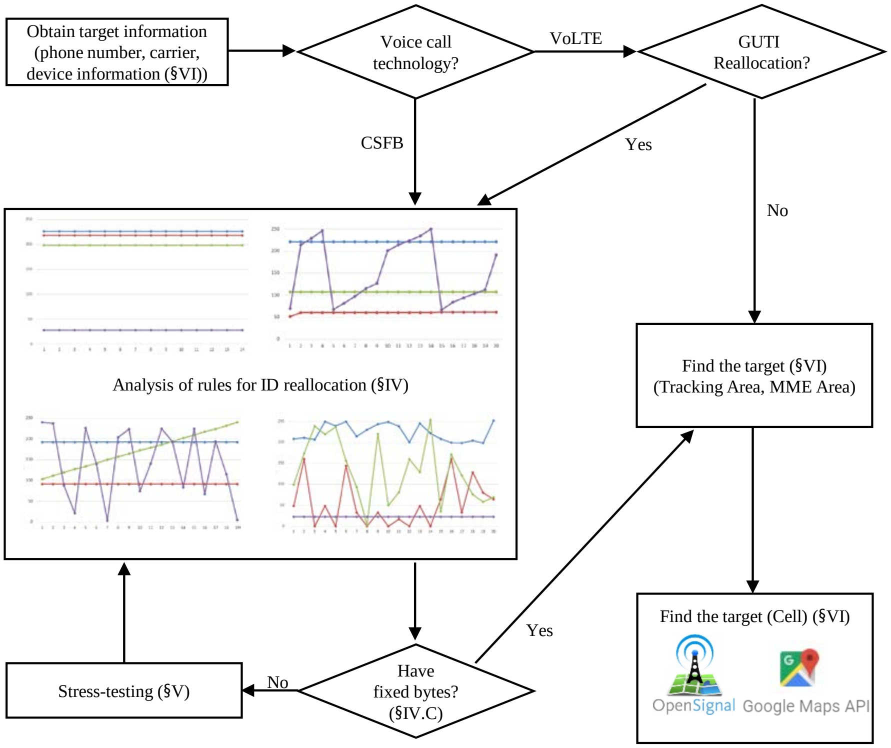

Obtain target information (phone number, carrier, device information (§VI))
Voice call technology?
VoLTE
GUTI Reallocation?
CSFB
Yes
No
Analysis of rules for ID reallocation (§IV)
Find the target (§VI) (Tracking Area, MME Area)
Yes
Stress-testing (§V)
No
Have fixed bytes? (§IV.C)
Find the target (Cell) (§VI)
OpenSignal
Google Maps API

Fig. 9: Attack flow for privacy leakage

GUTIs is one.

Consider the example when the victim's carrier is BE-II, the GUTI allocation rule for which is "Values of the first and second bytes in the GUTI are fixed, and that of the third byte shows a monotonically increasing pattern." While making the silent call and monitoring GUTI, the attacker does not ignore the monitored GUTIs whose values in the third and fourth bytes differ from those of candidates GUTIs but those of the first and second bytes are identical to those of the GUTIs in the set. These GUTIs are still considered candidates for the victim's GUTI. Further, the attacker considers the pattern of the values in the third byte and repeats the above procedure at every silent call until the candidate set only has one GUTI.

Attack through Stress Testing: For the remaining carriers for which we fail to reveal GUTI allocation patterns, an attack can still succeed through stress testing (see Section V). Through stress testing, the attacker makes the network (or MME) omit the GUTI reallocation procedure after several voice calls and sends a paging message with the same GUTI as that allocated previously. In the attack, the adversary performs the first launches stress test to cause the victim to have a persistent GUTI when the silent call is made. Once the GUTI of the victim is fixed, the attacker uses the same attack procedure as that used for VoLTE users. Therefore, the adversary can still infer mapping information between the GUTI and the subscriber, which is an opposite result to that specified by the relevant standard [5].

3) Cell Tracking: Once the TA where the victim is located is found, the attacker performs an additional procedure to find the specific cell. Unlike paging messages used to notify voice calls that are transmitted to multiple cells in the TA, paging messages for data service or SMS are delivered to only the cell where the UE is located. This paging procedure is called "smart paging" [13], [25], [30]. Moreover, all carriers in our dataset do not reallocate the GUTI after smart paging. In the attack, an adversary exploits the smart paging mechanism to find the cell where the victim is located. By invoking smart paging and monitoring the paging message, the attacker determines whether the victim is in the same cell if the victim's GUTI is monitored. Note that the victim's GUTI has already been revealed while finding the TA; therefore, invoking smart paging once is sufficient to leak the victim's location. Several methods that would go unnoticed by the victim can be used to invoke smart paging: (1) sending a broken SMS [17] and (2) using social messengers such as WhatsApp [43] or Facebook Messenger [14].

B. Experiment

We implement an LTE broadcast receiver on a laptop with a quad-core Intel CPU (i7 7500U) connected with a software-defined radio peripheral (B210), as described in Sections IV and V. Our proof-of-concept implementation shows (a) a location leakage attack on a VoLTE user and (b) the same attack on a CSFB user for two carriers that reallocate the GUTI at every voice call. In all experiments, we locate the victim and the attacker in the same TA, and we check whether our attack method could successfully locate the victim.

1) Attack on VoLTE User: We first perform location tracking attack on a VoLTE user subscribing to US-I. Note that a silent call does not change the GUTI of the target (call recipient). We generate a VoLTE call two times with a 6 s time gap. After the first silent call, we record all GUTIs in the paging message. By intersecting two sets of monitored GUTIs, one GUTI is obtained, and we confirm that it belongs to the victim. Having

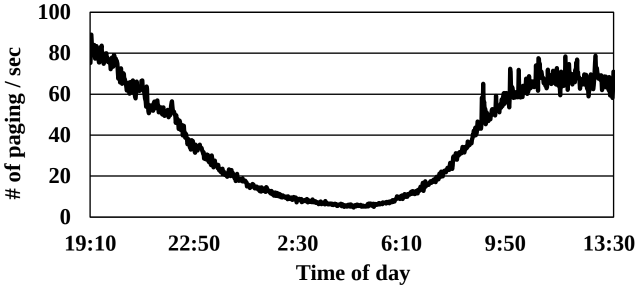


Fig. 10: Paging request per second in KR-I

found the GUTI and confirmed that the victim is located in the same TA as the attacker, we perform a broken SMS attack and succeed in specifying the victim's cell.

2) Experiments on Smart Tracking Attack: To show that the deployed GUTI reallocation remains vulnerable to location tracking attacks, we selected two carriers (US-II and KR-I), that do not show a noticeable GUTI allocation pattern and verify that our proposed attack works on both. We first perform a weak stress test on KR-I to fix the GUTI value. During the stress test, we monitor the GUTI to check for the existence of duplicated GUTIs. At the third CSFB call, we notice that one GUTI was constantly monitored on the paging channel; this is the victim's GUTI.

We used a different procedure for US-II. As observed in Section V, US-II shows duplicate GUTIs two times during the stress test. We perform stress testing with 30 sequential CSFB voice calls and listen to the paging channel to check for the existence of duplicated GUTIs. In our experiment, a GUTI with the value `0xC25BBDAE` appears at the third and fourth silent calls consecutively. To confirm that this is the victim's GUTI, we perform the same procedure again and confirm the existence of the victim in the same TA. Once GUTIs have appeared consecutively, we stop the second stress test and perform a cell tracking procedure by sending an SMS to the victim. As the GUTI has not changed upon receiving the SMS, we confirm the user's cell by checking for the existence of the GUTI monitored in the previous stress test.

C. Analysis of Effectiveness of Attack Method

One possible counterexample that can confuse our attack method consists of the cases in which GUTIs that have identical values in fixed bytes of the victim's GUTI constantly show up whenever we make a silent call. In this case, to locate the victim, the attacker should make the silent call several times until only one GUTI remains; however, it takes time to make calls, and this increases the likelihood of failure because the victim might move to another TA. Complicating the attack is that the attacker does not know the exact number of silent calls required to determine the existence of the victim in this case. To solve this problem, we measure the number of silent calls required to leak the victim's location through (a) probabilistic analysis and (b) simulation of a real environment. To reflect a real environment, we use the recorded traffic pertaining to paging messages and GUTI values on the paging channel of KR-I. Figure 10 shows the observed traffic rate for paging messages over 18 h.

1) Probabilistic analysis: First, we derive the probability of success of the smart tracking attack according to the number of silent call trials. We consider the case in which the GUTI reallocation rule has a fixed number of k bytes for each user, where $k \in \{1, 2, 3, 4\}$. Note that the fixed byte rule applies to each user, and this does not mean that all bytes of a given location in the broadcast channel are fixed. As described in section VI, we assume that the attacker listens to the broadcast channel for 1 s after the silent call. Let t be the number of paging messages appearing on the broadcast channel per second. For simplicity, we only consider byte positions that have fixed values and treat them in a concatenated form. For example, if the second and fourth bytes of GUTI are fixed and the monitored GUTI is `0x12345678`, we only consider the extracted value `0x3478`. We define $A_i = \{a_{i,1}, a_{i,2}, ..., a_{i,t}\}$ for the i-th call where k bytes ($8k$ bits), and $a_{i,j}$ represents the j-th value in the time window extracted using the above method.

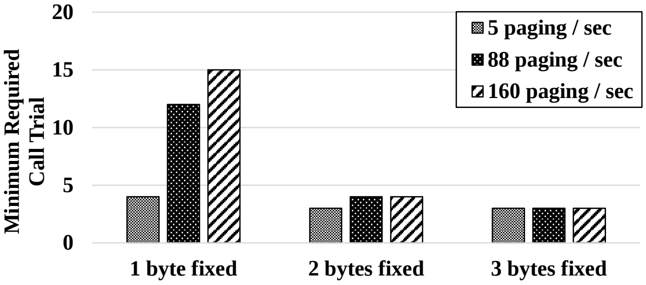
Along with these assumptions, we compute the probability that the intersection of these sets has at least one element during N calls. In other words, the probability indicates the existence of the same GUTI as the victim's one that shows up every time when the attacker makes N silent calls. The probability is calculated as shown below. We assume that the monitored values of k bytes ($a_{i,j}$) in the paging channel follow a uniform distribution[8]. Biases in the actual distribution may increase the expected number of calls; we cover this issue in the next experiments.

$$\begin{aligned}
Pr(\bigcap_{i=1}^{N-1} A_i \neq \emptyset) &= Pr(\bigvee_{a=0}^{2^{8k}-1} (a \in \bigcap_{i=1}^{N-1} A_i)) \\
&\leq \sum_{a=0}^{2^{8k}-1} Pr(a \in \bigcap_{i=1}^{N-1} A_i) \\
&= 2^{8k} Pr(a \in \bigcap_{i=1}^{N-1} A_i) \text{ for some } a \\
&= 2^{8k} \prod_{i=1}^{N-1} Pr(a \in A_i) \text{ for some } a \\
&= 2^{8k} \prod_{i=1}^{N-1} (1 - Pr(a \notin A_i)) \text{ for some } a \\
&= 2^{8k} \prod_{i=1}^{N-1} (1 - (\frac{2^{8k}-1}{2^{8k}})^t) \text{ for some } a \\
&= 2^{8k} (1 - (\tfrac{2^{8k}-1}{2^{8k}})^t)^{N-1}
\end{aligned}$$

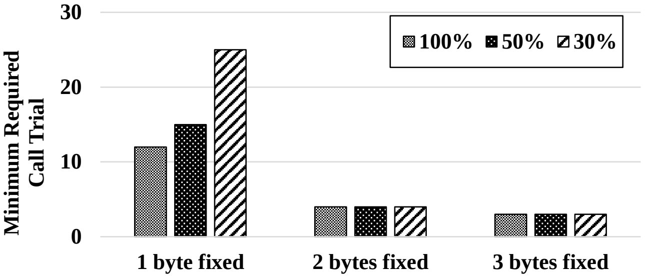
To determine the number of silent calls the attacker needs to make, we derive N_{min}, the minimum N value that makes the derived probability be less than 1%. Note that we calculate the upper bound of the probability. It implies that the attacker can designate the victim's GUTI within N_{min} times with 99% success rate. We measure N_{min} for varying paging message rates and GUTI utilization. Figure 11 shows N_{min} for the carriers adopting three types of GUTI allocation rules.

We first examine the case in which the attacker tries to locate the victim with varying paging message rates. For example,

[8] Note that the distribution of reallocated M-TMSI values at "one" target device is deterministic and predictable. Herein, the assumption of uniform distribution is for the value of fixed bytes in M-TMSI for the devices in the paging channel.



(a) Required number of call trial on varying paging message rate



(b) Required number of call trial on varying GUTI utilization

Fig. 11: Required number of calls covering 99% success rate

during the day when many people are active or in an area where the population density is high, the paging message rate is high, and therefore inferring the victim's GUTI becomes difficult. As Figure 10 shows, the highest and lowest paging message rates were 88 and 5 paging messages per second. Note that this experiment was ran on KR-I, whose MME sends paging messages to all eNodeBs (see Section VI-D for more details). Therefore, we believe this rate must be higher than that of other carriers. For example, US-II during the pick time has around 40 paging messages per second. Figure 11(a) shows N_{min} when we adopt these values to the derived equation, including the two times higher paging rate. As expected, when the paging message rate is high, the attacker needs to make more silent calls. For carriers using the *one byte fixed* rule, the attacker can locate the victim using only 5 silent calls during the night, whereas 12 silent calls are needed during the day. In addition, if the carrier uses the *two* or *three bytes fixed* rule, N_{min} is less than 5 in all cases.

Next, we apply the constraints that reflect the practical environment for the derived probability. By monitoring the paging channel, we observe that the GUTI values are not distributed uniformly. They are duplicated with each other at the byte level, and some are even reused. We consider this by defining the GUTI utilization as the ratio of actually used GUTI values to all possible GUTIs. For example, if the number of actually used GUTIs in the TA is 2^{30} among 2^{32} values, the utilization is 0.25. Figure 11(b) shows that low utilization makes it difficult for the attacker to determine the victim's GUTI. We set the paging message rate as 88 messages per second in this evaluation. Interestingly, the cases of *two* and *three bytes fixed* rules still show low N_{min}.

Through probabilistic analysis, we confirm that knowledge about the partially static GUTI is a big threat. Except when carriers use the one byte fixed rule, the attacker can easily locate the victim with a small number of silent calls that take only 2 min.

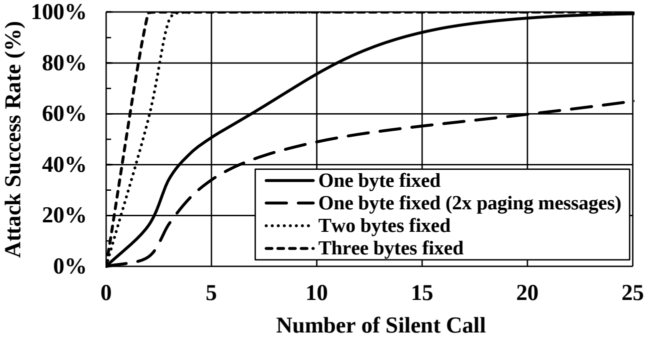


Fig. 12: Worst case simulated attack success rate in KR-I environment

2) Practical analysis: Next, we analyze the effectiveness of each attack with three GUTI allocation rules in a real-world environment. As in probabilistic analysis, we measure the number of silent calls required to leak the victim's location. Toward this end, we perform the attack every second on the recorded dataset to reflect a practical environment. We then measure the number of silent calls required where the number of candidate GUTIs is zero in each attack; this represents the number of calls needed to successfully expose the user's location. Note that this simulation checks the worst-case condition. In a real attack, the probability that the repeated pattern matches the duplicated case is extremely small. We run the simulation for three carriers using three different GUTI allocation rules (*one*, *two*, and *three bytes fixed*) by assuming that the dataset follows all rules. The total number of experiments is 64,471 over 18 h for each method with three GUTI allocation rules.

Figure 12 shows the success rate over the number of silent call trials for attacks with three GUTI allocation rules. The success rate for a given number of silent calls is the ratio of successful cases to the total number of attempts within the given number of silent calls. The result implies that the attacker could locate the victim with less than five silent calls for carriers using the *two* or *three bytes fixed* rule. For the carrier using the *one byte fixed* rule, 90% of attacks are successfully conducted within 15 silent calls, taking less than 7 min to locate the victim.

We note that the paging message traffic influences the success rate. We run the same experiment as the previous one, but monitor two times the number of paging messages relative to before. Carriers using the *two* or *three bytes fixed* rule are not significantly affected by the paging request rate [9]. In the worst case, when the victim's pattern matches the remaining ones, attackers might face difficulties in locating victims whose GUTI have been allocated by the *one byte fixed* rule. However, in practice, the number of calls required must be much smaller. Note that we ran the experiment in which the carrier does not use the *one byte fixed* rule.

D. Impact of Paging Coverage on Location Tracking

The commonly shared assumption on location tracking attacks [20], [30] is that MME sends the paging message to the

[9] Owing to the slight change, we omit the results in Figure 12

TA where the victim is located. Interestingly, we have observed that the paging coverage for voice calls varies across carriers. For example, the MME of KR-I sends the paging message to all eNodeBs, implying that paging messages are broadcasted to all TAs managed by the MME. This is mainly because the carrier wants to improve the Quality of Service (QoS). This was confirmed through interviews with KR-I. We believe that this is an exceptional case, as paging eNodeBs in all TAs is quite an expensive operation. The interviewee also believes that KR-I provides exceptional QoS compared with other carriers. This is an issue about the trade-off between the QoS of subscribers and the network overhead. Even if increased signaling messages on the network incur a processing overhead, configuring the very large paging coverage increases the probability of success of the first trial of paging request and reduces the call setup time. Owing to this configuration, we saw the victim's GUTI in the TA where the victim is not located when the voice call is triggered. As a result, this QoS configuration makes our attack face difficulties in locating the victim because it increases the number of false positives.

In addition, the large paging coverage restricts the precision of the location tracking attack. Because of this coverage size difference, subscribers who use carriers that send a paging message to one TA can be located by the TA (normally $< 30km^2$ in city) in the first step of the attack. However, in the case of carriers such as KR-I, the attacker can only determine the subscriber's location to a larger range than a city ($> 600km^2$ in South Korea).

E. Impact of Victim's Mobility on Location Tracking

Another shared assumption in location tracking attacks is that the victim should be stationary within the paging coverage. Of course, if the victim is moving inside the paging coverage, an attacker can specify the TA where the victim is located. However, if the victim is moving across multiple TAs, the attacker cannot locate the victim because the paging message cannot be seen at the TA where the victim existed previously. One promising result is that our attack model can locate the victim within a few minutes. Therefore, unless the victim is moving with high speed or crossing TA boundaries repeatedly, the attack procedure can be finished before the victim moves to another TA.

One possible workaround for this mobility issue could be having multiple paging message listeners over multiple TAs if the GUTI allocation pattern is maintained after moving to another TA [10]. In this approach, the attacker should monitor GUTIs over multiple TAs simultaneously; however, this requires additional cost to setup paging message listeners. Moreover, the number of monitored GUTIs increases in proportion to the number of listeners; this requires making more silent calls.

VII. Solution

In this section, we present a secure GUTI allocation logic that hides the binding between the subscriber and the temporary identifier. Note that we only focus on the reallocation of GUTI (more precisely, M-TMSI). In other words, there could be side channels other than the identifier itself. We believe this is beyond the scope of this work and leave this issue for future work. We first outline the five requirements for the logic derived from the above results discussed in previous sections and from 3GPP standards. We then present our design that meets these requirements. Lastly, we discuss the solution (Solutions #7.23 and #7.24) contained in the report from 3GPP [1].

A. Requirement

By analyzing the dataset we collected and associated 3GPP standards, we identified five key requirements for secure GUTI reallocation.

R1: Frequent refreshing of temporary identifier. As described in Section VI and in previous studies [20], [30], static or unchanging temporary identifiers allow an attacker to launch location leakage attacks on a victim. If the network reallocates temporary identifiers frequently, the attacker faces difficulties in tracking it and mapping it to a subscriber. This requirement is not different from that suggested in previous studies [1], [20], [30].

R2: Unpredictable Identity Allocation. To prevent an attacker from mapping the subscriber to his/her temporary ID, the next temporary identity that is assigned should be unpredictable. Specifically, note that all bits should be unpredictable. For example, if the victim's next identity value has a bit value overlapping with the previous identity, the attacker can find the mapping between the user and the identity by using only a few paging triggers (see Section VI).

R3: Collision Avoidance. The assigned identity should differ from the identities of other subscribers. This is because GUTI must be unique for each MME according to the 3GPP standard (see Figure 2). GUTIs from two MMEs must be different, because at least the MMECs are different. The network should check the use of identities when assigning a new one to a subscriber. Note that GUTI is reassigned only when GUTI reallocation is performed. Therefore, when a UE goes offline without sending a `Detach request`, its GUTI must not be assigned to others.

R4: Stress-testing Resistance. Temporary identity reallocation should not be omitted even if the mobile device or network is stressed. As described in Section V, performing a stress test led the UE to reuse the same GUTI, as GUTI reallocation was omitted.

R5: Low Cost Implementation. The solution must not incur significant computational and memory-related overhead to fulfill the above requirements.

B. A Secure GUTI Allocation Logic

Our main approach is to generate unpredictable secure pseudorandom bits. Many pseudorandom number generators (PRNGs) are known to not pass statistical randomness tests [33]. A cryptographically secure PRNG (CPRNG) does not have a polynomial time algorithm capable of predicting the k + 1-th bit with probability greater than 50% even if given a random sequence of k bits. This means that it is difficult to predict the next number even if the attacker knows the preceding sequence of numbers. For CPRNG, provably secure algorithms such as Hash_DRBG could be

[10] Theoretically, we expect that the GUTI allocation pattern will not be changed; however, we do not investigate it in this paper. This will be examined in future work.

TABLE IV: Notations for Identity Generation Algorithm

Hash	Selected secure hash function
HashGen	Hash generating function for Hash_DRBG [26]
V	Updated value during each call to the DRBG
C	Constant value that depends on the seed
c	Counter indicates the number of requests for pseudorandom bits

Algorithm 1 Identity block generation process

```
Input: Initial values for $V$, $C$, $c$
Output: Next candidates block for temporary identity
returned_bits ← HashGen($V$)
$H$ = Hash(0x03 || $V$)
$V = V + H + C + c$
$c = c + 1$
$realloc\_identity\_block = returned\_bits$
return $realloc\_identity\_block$
```

used for temporary ID reallocation. Hash_DRBG is a CPRNG standardized as NIST SP 800-90A [26]. A deterministic random bit generator (DRBG) generates a sequence of bits from a secret initial value called a seed. If the seed is not known and it has sufficient entropy, the cryptographic DRBG has a property that the output is unpredictable [10]. Kan *et al.* showed that the Hash_DRBG is secure if an appropriate hash function is used for the algorithm (e.g., SHA 256 could be used.) [16]. The Algorithm 1 briefly shows the identity generation process using Hash_DRBG [26] (see Table IV for notations). If the block size is 256-bit, it can generate 8 outputs of 30-bit length with only one operation.

The M-TMSI part of the GUTI has a total length of 32 bits; however, the two most significant bits are fixed for mapping with legacy networks [2]. Therefore, we need to obtain 30 bits from Hash_DRBG. By using CPRNG, we can generate a random GUTI. However, we have to avoid collision following R3. As the M-TMSI is only 32 bits long, to check the preemption of a randomly generated temporal ID value, we can simply use a bitmap structure. Considering the number of available M-TMSI, MME needs 2^{30} bins to check the usage of the M-TMSI value. When using a bitmap structure that can denote 8 bins with 1 byte, 128 MB memory is required. In addition, the occupancy can be checked by performing simple bitwise operations. Because Hash_DRBG could be based on SHA-256, the computational overhead is negligible [12]. Nevertheless, the time-memory trade-off to generate a few M-TMSIs in advance may reduce the online M-TMSI generation overhead. In the case of an MME that has 40 million subscribers [11] and that generates four M-TMSIs in advance, the probability of generating a preempted value is negligible [12]. We believe that requirements R1 and R4 are implementation- and operator-specific, and we do not discuss how to satisfy them in this study. However, these two requirements are crucial for avoiding location tracking.

C. Temporary Identifier Allocation in 3GPP

We discuss the requirement and the solution (Solutions #7.23 and #7.24) contained in the report from 3GPP [1]. As the reports study the current security issues in LTE, the security problems examined in this study as well as the requirements and solutions are written in an ad-hoc manner. We extract key requirements and solution ideas here. 3GPP also recognizes R1, R2, R3, and R5 as requirements. However, the solution examples provided in the reports are quite different from ours.

Unlike our solution, solutions #7.23 and #7.24 both bind K_{ASME}, the key shared between the MME and the subscriber, to generate the GUTI. The authors use K_{ASME} to generate temporary identifier that because K_{ASME} is a secret value known only to the MME and the subscriber. However, the authors did not discuss other reasons for this binding. In addition, as we discussed previously, to generate an unpredictable GUTI, a random seed long enough to be secure against exhaustive search is sufficient, and such binding is unnecessary, as CPRNG is secure.

For collision avoidance, solutions #7.23 and #7.24 suggest different mechanisms. The former suggests using the MAC (Message Authentication Code) to verify the identity. The latter suggests to increasing the length of M-TMSI to 64 or 80 bits to avoid the Birthday paradox. We believe that both are too expensive and unnecessary by sacrificing 128 MB of memory for checking duplicates.

These solutions are difficult to apply to current LTE networks because they require major changes in UE and MME; by contrast, our solution only needs a small update to MME. With regard to Solution #7.23, supporting the MAC for each paging message results in additional implementation in both the UE and the MME. Extending the length of the temporary identifier (Solution #7.24) requires the subscriber to replace the SIM card in use. In addition, #7.24, which suggests synchronized temporary identifier, should sufficiently cover side-effects such as synchronization problems [19] due to unexpected identity updates. Therefore, it can be a candidate solution for the next generation but is not suitable for current networks.

VIII. Conclusion

Mobile network operators and standards have invested a considerable amount of effort into identity management logic in cellular networks to ensure the confidentiality of their subscribers' identities. Unfortunately, subscribers are not safe from privacy leakage owing to incomplete specifications in the relevant standards and incorrect operation of identity management schemes by carriers. In this study, we examined the identity management systems of 28 carriers over 11 countries and showed that currently deployed systems fail to protect subscribers' temporary identity.

We identified three vulnerable implementations that allow an adversary to easily obtain the victims' location: rarely changed temporary identity, easily predictable identity allocation logic, and lack of resilience to exceeding allocation requests. We implemented three smart attacks to efficiently locate the victim and showed that they worked, even under the assumption that the previous solution is adopted as well. To prevent the threat

[11] Maximum capacity of commercial MME [27]

[12] Probability of collision with generated one M-TMSI is $(\frac{4*10^7}{2^{30}})$; therefore, that when preparing four M-TMSIs in advance would be $(\frac{4*10^7}{2^{30}})^4 \approx 1.92 \cdot 10^{-6}$

of leaking the victim's location, we described the requirements of an identity management logic and presented a solution that involves allocating an unpredictable temporary identity and updating it frequently. Our solution can be deployed with a small overhead, and we believe that carriers can support the confidentiality of their subscribers securely by using our solution.

Acknowledgment

We would like to thank the anonymous reviewers and our shepherd, Brad Reaves, for their insightful comments and suggestions to improve the paper. This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-2015-0-00403) supervised by the IITP (Institute for Information & communications Technology Promotion) and supported by Samsung Research, Samsung Electronics. We would like to thank Shinjo Park, Hyunwook Hong, Dongkwan Kim, Hongil Kim, Eunkyu Lee, Kibum Choi and Youngseok Park on a collection of data.

References

[1] 3GPP. TR 33.899, "Study on the security aspects of the next generation system," 2017.

[2] 3GPP. TS 23.003, "Numbering, addressing and identification," 2017.

[3] 3GPP. TS 24.008, "Mobile radio interface Layer 3 specification; Core network protocols; Stage 3," 2017.

[4] 3GPP. TS 24.301, "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3," 2017.

[5] 3GPP. TS 33.102, "3G security; Security architecture," 2017.

[6] 3GPP. TS 33.402, "System Architecture Evolution (SAE); Security aspects of non-3GPP accesses," 2017.

[7] 3GPP. TS 36.331, "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification," 2017.

[8] M. Arapinis, L. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon, and R. Borgaonkar, "New privacy issues in mobile telephony: fix and verification," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2012.

[9] M. Arapinis, L. I. Mancini, E. Ritter, and M. Ryan, "Privacy through pseudonymity in mobile telephony systems." in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2014.

[10] E. Barker, "NIST Special Publication 800-57 Part 1 Revision 4-Recommendation for Key Management (Part 1: General)," 2016.

[11] Cisco, "MME Administration Guide," http://www.cisco.com/c/en/us/support/wireless/asr-5000-series/products-installation-and-configuration-guides-list.html.

[12] "Crypto++ Benchmarks." [Online]. Available: https://www.cryptopp.com/benchmarks.html

[13] David Nowoswiat, "Managing LTE Core Network Signaling Traffic." [Online]. Available: https://insight.nokia.com/managing-lte-core-network-signaling-traffic

[14] "Facebook." [Online]. Available: https://www.facebook.com/

[15] N. Golde, K. Redon, and J.-P. Seifert, "Let me answer that for you: Exploiting broadcast information in cellular networks." in *USENIX Security Symposium*, 2013, pp. 33–48.

[16] W. Kan, "Analysis of underlying assumptions in nist drbgs." *IACR Cryptology ePrint Archive*, vol. 2007, p. 345, 2007.

[17] S. M. Karsten Nohl, "Wideband GSM sniffing," in *Chaos Communication Congress*, 2010. [Online]. Available: http://events.ccc.de/congress/2010/Fahrplan/events/4208.en.html

[18] M. S. A. Khan and C. J. Mitchell, "Improving air interface user privacy in mobile telephony," in *International Conference on Research in Security Standardisation*. Springer, 2015, pp. 165–184.

[19] ——, "Trashing IMSI Catchers in Mobile Networks," in *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2017.

[20] D. F. Kune, J. Koelndorfer, N. Hopper, and Y. Kim, "Location leaks on the GSM Air Interface," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2012.

[21] P. P. Lee, T. Bu, and T. Woo, "On the detection of signaling dos attacks on 3g wireless networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, 2007, pp. 1289–1297.

[22] Y. Li, C. Peng, Z. Yuan, J. Li, H. Deng, and T. Wang, "Mobileinsight: extracting and analyzing cellular network information on smartphones." in *Proceedings of the ACM Annual International Conference on Mobile Computing & Networking (MobiCom)*, 2016.

[23] Z. Li, W. Wang, C. Wilson, J. Chen, C. Qian, T. Jung, L. Zhang, K. Liu, X. Li, and Y. Liu, "Fbs-radar: Uncovering fake base stations at scale in the wild," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2017.

[24] J. Martin, T. Mayberry, C. Donahue, L. Foppe, L. Brown, C. Riggins, E. C. Rye, and D. Brown, "A study of mac address randomization in mobile devices and when it fails," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 268–286, 2017.

[25] Melih Tufan, "Packet Networks Portfolio," 2011. [Online]. Available: https://www.ericsson.com/ericsson/investors/doc/2011/ap_forum/ericsson_apac_forum_150911_packet_networks.pdf

[26] NIST, SP, "800-90a revision 1," *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*, 2015.

[27] Nokia, "The Alcatel-Lucent 9471 Wireless Mobility Manager," https://resources.ext.nokia.com/asset/156819.

[28] Nokia Network, "Signaling is growing 50% faster than data traffic," 2012.

[29] SecUpwN, "Android IMSI-Catcher Detector," 2012, https://cellularprivacy.github.io/Android-IMSI-Catcher-Detector/.

[30] A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J.-P. Seifert, "Practical Attacks Against Privacy and Availability in 4G/LTE Mobile Communication Systems," *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2016.

[31] "Fake Mobile Phone Towers Operating In The UK," http://news.sky.com/story/fake-mobile-phone-towers-operating-in-the-uk-10356433.

[32] "The body-worn "IMSI catcher" for all your covert phone snooping needs," http://news.sky.com/story/fake-mobile-phone-towers-operating-in-the-uk-10356433.

[33] J. Soto, "Statistical testing of random number generators," in *Proceedings of the 22nd National Information Systems Security Conference*, vol. 10, no. 99. NIST Gaithersburg, MD, 1999, p. 12.

[34] SRLabs, "CatcherCatcher," 2013, https://opensource.srlabs.de/projects/mobile-network-assessment-tools/wiki/CatcherCatcher.

[35] SRLabs, "SnoopSnitch," 2014. [Online]. Available: https://opensource.srlabs.de/projects/snoopsnitch

[36] "srsLTE." [Online]. Available: https://github.com/srsLTE/srsLTE

[37] STOKE, "Charting the Signaling Storms," 2013, http://s1.q4cdn.com/427257256/files/doc_downloads/Stoke_Documents/150-0032-001_IndInsights_ChartingSignalingStorms_Final.pdf.

[38] P. Traynor, M. Lin, M. Ongtang, V. Rao, T. Jaeger, P. McDaniel, and T. La Porta, "On cellular botnets: measuring the impact of malicious devices on a cellular network core," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2009.

[39] G.-H. Tu, Y. Li, C. Peng, C.-Y. Li, H. Wang, and S. Lu, "Control-plane protocol interactions in cellular networks," in *Proceedings of the 2014 ACM conference on SIGCOMM*. ACM, 2014, pp. 223–234.

[40] G.-H. Tu, C. Peng, H. Wang, C.-Y. Li, and S. Lu, "How voice calls affect data in operational LTE networks," in *Proceedings of the ACM Annual International Conference on Mobile Computing & Networking (MobiCom)*, 2013.

[41] "USRP B210." [Online]. Available: https://www.ettus.com/product/details/UB210-KIT

[42] F. van den Broek, R. Verdult, and J. de Ruiter, "Defeating IMSI catchers," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2015, pp. 340–351.

[43] "WhatsApp," https://www.whatsapp.com/.