


Dynamic ID randomization for user privacy in mobile network

Arijet Sarker¹  | SangHyun Byun¹ | Manohar Raavi¹ | Jinoh Kim² |
Jonghyun Kim³ | Sang-Yoon Chang¹

¹Department of Computer Science,
University of Colorado Colorado Springs,
Colorado Springs, Colorado, USA

²Department of Computer Science and
Information systems, Texas A&M
University-Commerce, Commerce, Texas,
USA

³Electronics and Telecommunications
Research Institute, Daejeon, Republic of
Korea

Correspondence

Arijet Sarker and Sang-Yoon Chang,
Department of Computer Science,
University of Colorado Colorado Springs,
Colorado, USA.
Email: asarker@uccs.edu

Funding information

Ministry of Science and ICT, South Korea,
Grant/Award Numbers: 2021-0-02107,
2021-0-00796; National Science
Foundation, Grant/Award Number:
1922410

Abstract

Mobile and telecommunication networking uses temporary and random identifiers (IDs) to protect user privacy. For greater intelligence and security of the communications between the core network and the mobile user, we design and build a dynamic randomization scheme for the temporary IDs for mobile networking, including 5G and 6G. Our work for ID randomization (ID-RZ) advances the existing state-of-the-art ID re-allocation approach in 5G in the following ways. First, ID-RZ for ID updates is based on computing, as opposed to incurring networking for the re-allocation-based updates, and is designed for lightweight and low-latency mobile systems. Second, ID-RZ changes IDs proactively (as opposed to updating based on explicit networking event triggers) and provides stronger security (by increasing the randomness and frequency of ID updates). We build on the standard cryptographic primitives for security (e.g., hash) and implement our dynamic randomization scheme in the 5G networking protocol to validate its design purposes, which include time efficiency (two to four orders of magnitude quicker than the re-allocation approach) and appropriateness for mobile applications.

KEYWORDS

cellular networking, low latency, mobile computing, 5G, 6G, temporary ID, user privacy

1 | INTRODUCTION

Mobile and wireless devices are increasingly being connected in networking. To protect the transaction and location privacy of mobile devices (and, in turn, the people using them), networking employs temporary identifiers (IDs) to disable the unauthorized tracking of user devices. This issue is especially of high risk for mobile and wireless devices are at risk to this issue because their wireless channels are inherently open and publicly accessible, making them vulnerable to radio-

equipped attackers, including software-defined radio (SDR)-based attacks. In standard cellular communication and telecommunications, using temporary IDs to prevent unauthorized location tracking has been proposed in 2G [1]. Furthermore, the 3rd Generation Partnership Project (3GPP) has implemented temporary IDs as part of their standards. The existing state-of-the-art approach in 5G networking uses temporary ID re-allocation to update temporary IDs. The temporary ID change using re-allocation is reactive and occurs after the initial registration, mobility and periodic registration updates, and the

service request in response to a paging message, among others [2]. However, against a passively observing attacker trying to compromise user privacy using temporary ID tracking, the state-of-the-art reactive re-allocation scheme is insufficient to protect privacy because the updates are not frequent enough [3] and are predictable as some bits are fixed/sequential [4]. An attacker who succeeds in temporary ID tracking can use it to exploit other attacks breaching user privacy, such as location tracking using a user's phone number of a user [4–6], identity mapping, website fingerprinting, DNS spoofing [7], video identification [8], blind DoS, and remote de-registration [9].

ID-RZ addresses these issues and enables frequent and unpredictable temporary ID changes. Randomization differs from the state-of-the-art re-allocation approach because it updates and changes the temporary ID without explicit communication, proactively changes the temporary ID (as opposed to reactive or condition/event driven changes), and increases the rate of the temporary ID change (to increase tracking difficulty and thus privacy).

ID-RZ supports low-latency wireless applications by avoiding incurring communication overhead per ID update and making it compatible with the existing mobile-networking protocol in 5G, including ultra-reliable and low-latency communication (URLLC), which requires latencies below 50 ms [10, 11]. Particularly, ID-RZ builds on the current networking protocol to exchange the permanent ID, utilizing it as the seed to compute and generate a sequence of unpredictable temporary IDs using a hash-chain-based construction to enable randomization.

The rest of this paper is organized as follows. Section 2 describes related works, including the state-of-the-art re-allocation approach. We describe the relevant 5G networking background in Section 3, from which we build the ID-RZ system model in Section 4 and explain the threat model, the threat scenarios, and the ID-RZ requirements in Section 5. Section 6 describes and explains ID-RZ, comparing it with the state-of-the-art approach, while Section 7 analyzes ID-RZ, including the security and collision. Our proposals for next-generation cellular networking are described in Section 8, while we implement ID-RZ and analyze the performance and costs/latency in Section 9 to highlight the lightweight and low-latency design. Section 10 concludes our paper.

2 | RELATED WORK IN 5G USER PRIVACY

In this section, we describe the relevant research on securing user privacy in 5G networking and define the state-of-the-art re-allocation scheme. Subsequently, we compare

the re-allocation scheme to ID-RZ to highlight the research contributions. Gorrepati and others [12] discuss the privacy impact assessment by service providers and emphasize the need to make assessments from a subscriber's perspective. They provide an overview of privacy risks concerning LTE and 5G subscribers due to exploitable vulnerabilities in new technologies. Further, they discuss the need to update the temporary ID frequently after paging in LTE and 5G. Hong and others [4] investigate the temporary ID reallocation problem in 28 carriers across the world (11 countries). They find that most carriers fail to update the temporary IDs frequently, risking the privacy of users. ANOTEL, which provides location management services [13], introduces pseudonym and location provider entities for maintaining the location privacy of users. Nevertheless, the scheme induce significant performance overhead and reliance on two new trusted third parties. For maintaining user location privacy, Nicanfar and others [14] propose updating temporary ID using the handover mechanism during the user movement between location areas. However, this approach does not provide a solution against location privacy when the user stays in the same area. Other studies have proposed using blockchain [15] or permanent equipment ID [16] to protect user privacy in 5G, which are orthogonal to our work.

Particularly relevant to ID-RZ are those with the same goal of dynamically updating temporary IDs for cellular networking. The state-of-the-art re-allocation approach is based on *allocations* triggered by events and requiring explicit networking per update, as described by the 5G New Radio (NR) standard [2, 17]. While the 5G NR standard does not provide/construct the lower-level details or the implementation flexibility for the network managers, other research on temporary ID updates and allocation uses the hash function to generate the pseudo-random number ID [4].

We refer to the technique that uses event- and networking-based allocation and hash-function-based random ID generation as *state of the art*. Compared to the state-of-the-art re-allocation mechanism, ID-RZ updates the temporary ID more dynamically and frequently and is more lightweight with lower latency. Subsequently, we will compare ID-RZ with the state-of-the-art re-allocation scheme regarding the design analyses and the implementation-based evaluation and analyses.

3 | BACKGROUND IN 5G NETWORKING AND CONTROL COMMUNICATION

The 5G New Radio architecture involves the user, base station, and core network and includes control

communication to set up the data communication channels. Figure 1 provides an overview of the control communication and the temporary ID allocation in 5G. The control communication involves the following steps: Channel Setup, Registration Request, (Security) Setup Establishment, Setup Suspend, and Setup Release.

First, the user requests a radio setup and gets a temporary identifier, C-RNTI ID, Cell Radio Network Temporary Identifier (C-RNTI), from the base station (*Channel Setup*) [18]. There is no security setup (e.g., key establishment) between the user and base station/core network in this stage [19]. However, each user can still encrypt an already assigned unique permanent identifier ID₀ with the home network public key (securely provisioned offline). The user sends the encrypted ID₀ with the initial registration request message to the network (*Registration Request*), unlike in earlier versions (i.e., 4G and 3G) where ID₀ is sent in plaintext [20]. Upon verifying the user with the ID₀, the base station and core network establish a security setup with the user (messages are encrypted and integrity protected when security setup is established), and the core network provides a temporary ID_t (*Setup Establishment*) to the user. The user uses the ID₀ to authenticate itself with the core network only after rebooting or special cases (i.e., the core network is unable to map ID_t to ID₀). Otherwise, ID_t is used for authentication in other purposes (e.g., mobility registration, periodic registration, and service request) with the core network. The user also uses this ID_t to transmit and receive data after the security setup is established with the core network (*Data Communication*). To reduce battery consumption, network signaling load, and latency, a user can go into two states when there is no ongoing data transmission: (i) *inactive state* (no security setup connection with the base station, but the security setup context

remains established with the base station and core network) and (ii) *idle state* (no security setup connection with the base station and core network). The base station and core network allocate an Inactive- Radio Network Temporary Identifier (I-RNTI) and ID_t before the user moves to the *inactive (Setup Suspend)* and *idle states (Setup Release)*, respectively. If any new data arrive for the user in the *inactive* and *idle states*, then the base station and core network can search for that particular user (paging message) using already assigned with the I-RNTI and ID_t, respectively, assigned during the previous *Setup Establishment* stage. Conversely, if the user needs to send a service request message in the *inactive* and *idle states*, it uses the I-RNTI and ID_t, respectively, assigned during the previous security setup stage. The paging and service request messages in the *inactive* and *idle states* are sent in plaintext.

4 | SYSTEM MODEL

ID-RZ builds on the current 3GPP standard in 5G NR networking described in Section 3.

4.1 | Variables and notations

In this section, we describe the notations and variables used in ID-RZ. Table 1 lists the relevant terminology used

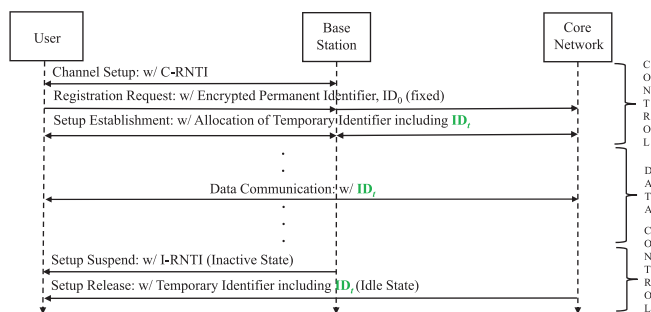


FIGURE 1 Overview of temporary ID allocation to users in 5G. In the current 5G NR, ID_t changes when there is a new allocation of the temporary ID and does not change in this diagram. However, in ID-RZ, *t* increments and ID_t changes for every communication step (or the arrows in this figure) and data communication

TABLE 1 Variables and acronyms used in this paper versus those used in the 3GPP standard for 5G NR [17, 19, 21]

Var./Acr.	Our work	3GPP standard (5G NR)
ID _t ^a	Temporary identifier	Short-Temporary Mobile Subscriber Identity (S-TMSI)
I-RNTI	I-RNTI	Interactive-Radio Network Temporary Identifier (I-RNTI)
C-RNTI	C-RNTI	Cell-Radio Network Temporary Identifier (C-RNTI)
	Base station	gnodeB (gNB)
	Core network	Access and Mobility Management Function (AMF)
ID ₀	Permanent identifier	Subscription Permanent Identifier (SUPI)
	Encrypted permanent identifier	Subscription Concealed Identifier (SUCI)

^aID_t is the user-specific part of the S-TMSI.

in our paper against those in the 3GPP standard for 5G NR. t corresponds to the discrete time in communication steps; that is, t is a positive integer, which increments as communication occurs. ID_0 is the permanent ID previously established at user registration, and ID_t refers to the temporary ID of the user at the time t ($t \geq 1$). The hash function, seed, and Boolean flag (to determine if the temporary ID is verified) are denoted by H, s , and B , respectively.

ID-RZ randomizes the user ID. For ID-RZ, n corresponds to the length of the hash chain and the number of hash function computations given the setup and establishment of the channel and permanent ID, thus referring to the number of temporary IDs (ID_t) generated in one setup contrary to the state-of-the-art re-allocation approach requiring networking per temporary ID. W is the input of the hash chain, which combines the permanent ID ID_0 and the random seed s . The f function corresponds to the processing of the hash function output to make it suitable for the temporary ID format in 5G. For example, we use XOR and LSB for f subsequently. While ID-RZ builds on the correctness and security of the cryptographic primitives (e.g., the one-way and collision-resistance properties of the cryptographic hash function), we analyze the performance and efficiency of ID-RZ by measuring the time cost overheads. We define T_C for the computing latency (to generate ID_t) and T_V for the verification latency (to verify ID_t of the user by the core network). T_N is defined as the networking latency for sending ID_t in the one-way direction (e.g., from the user to the core network or vice versa). Thus, $2T_N$ is the networking latency in both directions. Since T_N excludes the intra-node computing (which we separately analyze), the total latency is T ; that is, $T = T_C + 2T_N + T_V$.

4.2 | SCOPE OF OUR WORK

ID-RZ focuses on the ID_t is the user-specific part of the Short-Temporary Mobile Subscriber Identity (S-TMSI)—which is the Temporary Mobile Subscriber Identity (TMSI) and has a length of 32 bits—and excludes the core-network part or the AMF-specific part of the ID (16 bits for the AMF Set ID and AMF Pointer in 5G NR) (Figure 2). S-TMSI is a part of the global unique temporary identifier (GUTI), which includes the user-specific ID, and is transmitted/networked between the user and the core network (not all parts of GUTI are transmitted when networking).

ID_t does not change during the steps depicted in Figure 1 but rather after the diagram at during the time of re-allocation of the temporary ID in the current state-of-the-art approach in 3GPP [17, 19, 21]. In contrast,

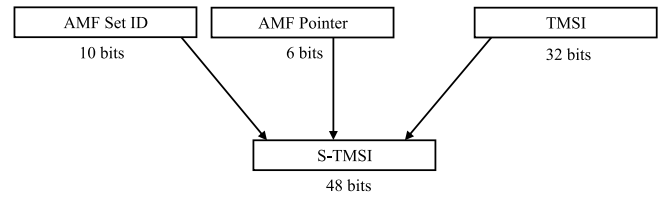


FIGURE 2 Short temporary mobile subscriber identity

ID-RZ increments t and updates ID_t for every control communication step and data communication.

ID-RZ focuses on the S-TMSI between the user and core network, rather than I-RNTI and C-RNTI. This is because ID-RZ is implemented between the core network and user (separated by multiple hops and greater physical distance), while I-RNTI and C-RNTI focus on the wireless channel setup, including the medium access control between the user and base station (one hop in networking). Since of our scope focuses on S-TMSI, we use the variable ID_t for S-TMSI and not I-RNTI or C-RNTI. Table 1 lists the relevant terminology used in our paper vs. in 5G NR.

5 | THREAT MODEL AND ID RANDOMIZATION REQUIREMENTS

5.1 | Threat model

ID-RZ considers and defends against an attacker monitoring and observing communication to breach the location and transaction privacy of a user [4]. In the ID-RZ threat model, the attacker can monitor communications between the user and base station/core network, knows the 5G protocols, including the ID_t design, and implementations using Kerckhoff's principle, and has access to the software and hardware needed to listen and decode control-communication messages (such as paging and service request messages in the *inactive* and *idle states*).

The attacker can also process the observed information, such as finding the pattern or correlation among the number of ID_t s allocated to a particular user; for example, incrementing ID_t s for the same user.

ID-RZ focuses on attackers who do not require the compromise of the networking service provider and thus pose an increased threat. Thus, the base station and core network providing the networking services are outside of the scope of our threat model. Moreover, we focus on protecting the information associated with the user ID by dynamically changing/randomizing it.

5.2 | Threat scenarios

ID-RZ defends against threats that gain intelligence/information from the unauthorized tracking of user ID, as described in Section 5.1. Without ID-RZ, the attacker can use such information for active threats, which can have a concrete and devastating security impact on the user. These include modifying the unencrypted part of the message exchanged between the user and base station/core network. Simultaneously, it keeps the header checksum the same as the original message payload, acts as a malicious relay between the user and base station [7], and sets up a rogue base station [9]. Utilizing this active threat model and the information gained from unauthorized ID_t tracking, an attacker can track a user using their phone number [4–6]; map identities to obtain data link layer information to perform website fingerprinting, DNS spoofing [7], and video identification [8] attacks; and disconnect a legitimate user from the base station/core network by performing blind DoS and remote de-registration [9] attacks.

Since ID-RZ prevents the attacker for getting such information, ID-RZ prevents and disables these threats (see Section 7.3).

5.3 | Requirements for temporary ID randomization

We design ID-RZ to meet the five key requirements for secure ID_t updates established by previous research [3–5]: frequent updating of ID_t by the core network, unpredictable ID_t reallocation, allocation of unique ID_t s to the users, resistance against stress-testing, and low computation and memory overhead. Regarding the uniqueness requirement of the temporary IDs, using hash function in ID-RZ yields hash output collisions with small probabilities (e.g., a couple of collisions for every million packets). However, to ensure the uniqueness, we resolve the collisions by coupling the collided ID (ID_x , i.e., the collision occurred at time $t=x$) with the subsequent ID (ID_{x+1}), as we will discuss in Section 7.2.

In addition to the previous research requirements, we ID_t randomization. The first, especially important for low-latency applications, is *low communication costs*, specifically low latency costs. ID-RZ fulfills this requirement by enabling the temporary ID update/change without networking while sharing the seed input for the temporary ID generation offline (i.e., before the ID randomization and networking). Therefore, ID-RZ only requires the computing overhead, in the order of hundreds of microseconds at most, and not the networking/communication

overhead, in the order of tens of milliseconds (two orders of magnitude greater than computing).

The second requirement to make it *compatible and modular* with the existing networking standards, which is 5G in our case, as described in Section 3, enables wider deployment of the solution for dynamic temporary IDs [Editor4]. While building on the rest of the technologies used in 5G, a modular solution for updating the temporary IDs on both the authorized transmitter and receiver (e.g., holding the secret key or seed) can facilitate the deployment of the solution by keeping most of the networking protocol intact.

6 | DYNAMIC TEMPORARY ID RANDOMIZATION SCHEME, ID-RZ

In this section, we describe and explain ID-RZ and its integration into the 5G networking protocol. Section 6.1 describes temporary ID generation based on the local computing on the user and the core network, while Section 6.2 focuses on the networking protocol integration of ID-RZ.

6.1 | Temporary ID generation

We generate multiple random temporary IDs using a hash chain given the input of $W = s||ID_0$ provided from networking, as depicted in (Figure 3). The generation of the temporary IDs (ID_t) is from left to right. Conversely, the actual use of the ID_t for networking between the user and the core network is from right to left. We use the hash function H for two purposes, which are well established in cryptography. First, H provides pseudo-random outputs, which makes its output unpredictable. Second, the one-way property of the hash function H (i.e., it is easy to compute from input to output, but it is computationally difficult to compute from output to input) makes that the future ID_t (e.g., the attacker observing ID_t cannot derive ID_{t+1}). The formal security proofs for the pseudo-random outputs and hash functions are provided in Rukhin and others [22] and Appel [23].

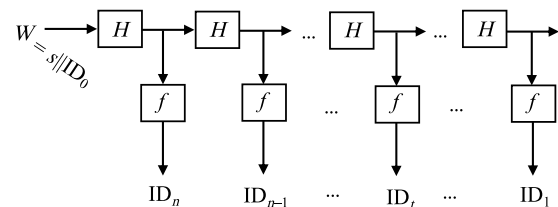


FIGURE 3 Temporary ID generation process

Due to these security properties, similar hash-chain-based pseudo-random bit generation constructions have been used in secure networking, including one-time password generation using S/Key [24], IP address randomization in SDN [25], and the wireless sensor network broadcasting authentication of TESLA [26].

To make the hash function outputs compatible to 5G networking, we apply the f function. In the 3GPP 5G NR standard, the user-specific ID is a 32-bit part of the 48 bit-long S-TMSI, as described in Section 3. Therefore, f takes the hash output and compresses it to be 32-bit long.

6.2 | Integration into the 5G networking protocol

In this section, we describe how the proposed ID-RZ can be incorporated to the 5G networking protocol; particularly, how ID_t s. Since ID-RZ supports multiple ID_t generation based on seed networking, as described in Section 6.1, ID_t changes as t increments in every step during control and data communications.

6.2.1 | Exchange of s and n

Initially, the core network generates s and n for the user, where s and n are dynamic. Since s needs to be secret between the user and the core network, the core network exchanges s and n with the user after Setup Establishment between them so that these messages are encrypted and integrity protected. n (which does not need to be kept secret from the attacker to secure ID-RZ) is the number of temporary IDs per seed exchange and is a parameter for controlling the overhead/frequency of exchanging the hash chain seed s . The value of the control parameter n depends on the 5G implementation, which varies across companies and regions (in general, the 5G standardization leaves such parameter value selections to the developers and operators). Afterward, the user and core network have the input for the hash chain, $W = s || ID_0$, and thus generate ID_t as described in Section 6.1. The ID_t s are used in the reverse order of hash generation; for example, $ID_1 = f(H^n(W))$, $ID_2 = f(H^{n-1}(W))$, ..., $ID_t = f(H^t(W))$, ..., $ID_n = f(H^1(W))$.

6.2.2 | Exchange of ID_t 's

After generating ID_t s, the core network or user can use them (each ID_t can be used once) in a monotonically increasing pattern, such as $ID_1, ID_2, \dots, ID_t, \dots, ID_n$, for

their respective purposes (i.e., Setup Establishment, Data Communication, and Setup Release, etc.) regardless of who (core network or user) is using it. For example, if the core network does Setup Release using ID_3 for a particular user, the user sends the next Setup Establishment request with ID_t s generated, they can easily verify the received ID_t list. In ID-RZ, the user and core network can continuously use new ID_t set for every communication (until n is exhausted) without the explicit networking-based reallocation of the ID. In contrast, in the current 5G NR, the core network needs those events to allocate new ID_t s to the user after the Setup Establishment and before the Setup Release (otherwise the same ID_t is used for subsequent communication). In ID-RZ, the core network needs to exchange new s and n with the user only when n is exhausted to allocate a new set of ID_t s.

Algorithms 1 and 2 show the pseudocode for ID_t generation and verification process, respectively. ID_t generation (Algorithm 1) is implemented in both the user and core network, whereas ID_t verification (Algorithm 2) is implemented and installed in the core network. The inputs of Algorithm 1 are s and n . s is concatenated with ID_0 to generate W . Subsequently, H is applied to W till n and each hash value are kept in $r[t]$ in Algorithm 1. Afterward, f is applied to each $r[t]$ to generate the ID_t s. f can be either bit-wise XOR or LSB, taking the less significant bits of $r[t]$. The input of Algorithm 2 is the ID_t s, $ID_t[0:n-1]$, for the user. If it is found, then Algorithm 2 returns $B = 1$, meaning ID_t is found. Otherwise, it returns $B = 0$, meaning ID_t is not found.

For multiple users, the core network saves these ID_t s for each active user; for example, each user requires 4 kB of storage for $n = 1000$; if there are one million active users, then the storage overhead of the core network is 4 GB (the storage of a small USB flash drive).

Algorithm 1: ID_t generation

Input: s, n
Output: ID_t
 $W = s || ID_0$
 $t = 0$
while $t < n$ **do**
 $r[t] = H(W)$
end while
for each $r[t]$ **do**
 $ID_t = f(r[t])$
end for

Algorithm 2: ID_t verification

Input: $ID_t, ID_t[0:n-1]$
Output: B
 $B = 0$
if ID_t in $ID_t[0:n-1]$ **then**
 $B = 1$
 Return B
else
 $B = 0$
 Return B
end if

6.3 | Comparison of ID_t re-allocation to our approach

The 3GPP standard for the 5G NR does not specify how to generate temporary ID and when to update it, leaving the implementation details and update frequency to the network operators' implementation [17, 19]. However, the network providers are reluctant to update the temporary ID frequently (due to expensive protocol interactions and cryptographic operations). A predictable pattern is found in temporary ID allocation by network operators, such as fixed/sequential bits for a particular user [4]. In ID-RZ, a set of unpredictable temporary IDs can be generated (the core network can vary the number of temporary IDs to be generated in one set according to its requirement) with significantly fewer protocol interactions and cryptographic operations (providing low-latency and low-cost communications) between the user and core network. Hong and others [4] propose a hash-based deterministic random bit generator (HASH_DRBG) for generating temporary IDs where the seed is constant. In our approach, the core network updates the seed for a user while updating the set of temporary IDs.

7 | ANALYZING ID-RZ

7.1 | Resolution of ID_t collision

ID-RZ is compression based and generates a shorter output than the input by first using a hash function. Subsequently, it compresses the output further to 32 bits to fit the 5G NR standard, excluding the separate 16-bit field for the core network/AMF. Therefore, the ID_t s can *collide*; that is, multiple users can have the same temporary ID sometimes. If the collision occurs in different cells, then the core network can use the other cell-specific identifier fields generated by the base station (e.g., Cell ID) to distinguish between the collided users. If the collision occurs in the same cell, the following ID_{t+1} can be used to distinguish between the users/streams since the probability of consequent collisions decreases exponentially (i.e., less than or equal to $(10^4/2^x)^x \approx (2.33 \cdot 10^{-6})^x$ for x consequent packets). When a collision occurs, the base station can temporarily store the packet corresponding to the $collID_t$ and resolve it later when the next packet is communicated at $t+1$. Such collision resolution rarely occurs, as the collision probability in a cell and per base station is low. Given that a base station can support up to thousands of users at a given time, depending on the cell size in 5G (e.g., up to 1200 [27]), the probability of collision is upper-bounded by $10^4/2^{32} \approx 2.33 \cdot 10^{-6}$. Alternatively, the ID

collision will occur a couple times or less for every million packets, since the pseudo-random hash output generates the ID_t randomly in a uniform distribution.

7.2 | Security analyses

7.2.1 | Secure exchange of s

The security of our temporary ID randomization scheme, ID-RZ, relies on the secrecy and the secure exchange of the ID-RZ seed s . s acts as a symmetric key/input between the user and core network. In ID-RZ, we share the impending session's s from the previous session, in which the Setup Establishment includes 5G authentication and key management (5G-AKA) [2] and the security key exchange enabling the confidentiality-protected exchange of s (as well as its existing purpose for encryption on the data communication).

7.2.2 | Randomness and unpredictability of ID_t

We improve the security of the current 5G NR technology in two ways. First, contrary to the current 5G practice that does not fully use 32 bits for the random temporary ID, rather introducing fixed bytes, we use the full 32 bits and maximize its entropy, building on the pseudo-random property of a hash function. Second, we increase the frequency of ID_t changes to increase the tracking difficulty more than the current reallocation-based temporary ID updates.

7.2.3 | Orthogonal to the existing security

The current 5G NR does support security protection for confidentiality and integrity, including the Setup Establishment involving the security setup exchanges (e.g., 5G-AKA [2]). ID-RZ is orthogonal to these security mechanisms and does not interfere with them. Thus, we can add ID-RZ to these other security mechanisms. Improving the data communication confidentiality or the integrity (e.g., against an active attacker injecting communications) is beyond the scope of this paper.

7.2.4 | Man-in-the-middle attack

A man-in-the-middle, mID_t , to simply relay the communication between the user and core network to eavesdrop or inject fake messages into the unencrypted control messages.

However, our approach defends against man-in-the-mID_{*t*} before observation. Therefore, a man-in-the-mID_{*t*} for the authentication.

7.3 | ID-RZ Against threat scenarios

An attacker that can track the ID_{*t*} information to perform various powerful attacks on user privacy and DoS. One of such threats is mentioned in Hong and others [4], Hussain and others [6], and Kune and others [5], where an attacker can frequently place silent phone calls, SMS, or messages with instant messengers (e.g., WhatsApp) to the user in a short period of time to observe the ID_{*t*} appearing frequently in the paging messages, it can track the location of the user, concluding that the ID_{*t*} belongs to the user. ID-RZ prevents this attack by randomizing the ID_{*t*} frequently so that when the attacker launches this attack, it observes different ID_{*t*} to a particular user.

Utilizing ID_{*t*} tracking and the man-in-the-middle attack to track the location of a user, an attacker can launch considerably powerful attacks, as described in Bae and others [8], Kim and others [9], and Rupprecht and others [7]. These attacks [7, 8] are focused on user privacy and can link the user ID_{*t*} to its radio identity. That is, C-RNTI utilizes the data link layer information to obtain information about the websites visited and videos watched by a particular user. Moreover, the authors in Rupprecht and others [7] showed that an attacker can change the IP address of the DNS request of a particular user, redirecting it to a malicious website to launch DNS spoofing attack. However, these attacks (assuming the ID_{*t*} is long-lasting) need to map the ID_{*t*} of the user to its radio identities to obtain the user's data link layer information. Since ID-RZ randomizes ID_{*t*} frequently for every packet, the attacker cannot link a specific user ID_{*t*} to its radio identities and obtain the corresponding data link layer information. The attacks described in Kim and others [9] showed DoS attacks, (e.g., blind DoS, and remote de-registration), where an attacker can disconnect a legitimate user from the base station and core network by capturing the packets between the user and base station, and by spoofing the connection using the captured ID_{*t*}. However, ID-RZ prevents these attacks because the captured ID_{*t*} cannot be reused for subsequent authentication.

8 | PROPOSALS TO NEXT-GENERATION CELLULAR NETWORKING

As described in Section 6.2, ID-RZ builds on the 5G networking and is designed to incur minimal additional

overheads. For the 5G existing protocol, we recommend using the 5G-AKA protocol to establish the security setup and the ID field/information for the ID-RZ randomized ID delivery. Implementing ID-RZ requires sending *s* after the Setup Establishment (e.g., 5G-AKA [2]) to keep it secret from the attacker monitoring the communication between the user and base station. Provided a protocol similar to 5G-AKA is adopted in 6G for confidentiality and message integrity, ID-RZ can be implemented in 6G. ID-RZ also builds on the ID delivery in the cellular control communication and the inclusion of the information field in the transmissions, which occur before the payload-data communications and the connectivity provisions.

While ID-RZ does not incur any additional networking (no additional bits to transmit beyond the existing 5G protocol), it does incur additional computing overheads. ID-RZ requires ID_{*t*} generation and verification, as described in Algorithms 1 and 2, for the user and the core network. We show that the computing performance has minimal overhead for ID_{*t*} generation and verification in Section 9.1.2.

We expect ID-RZ to be easily applicable to the next-generation cellular networking in 6G, since the ID-RZ-relevant parts of the security setup and ID transfer have been consistently implemented in 2G since 1991. In addition to keeping and adapting the relevant parts from 5G (e.g., 5A-AKA), we recommend that 6G includes an option to enable ID-RZ computing and the automatic/dynamic ID updates on the user and the core network.

9 | IMPLEMENTATION ANALYSES

9.1 | Computing for ID randomization

9.1.1 | Computing implementation

Our implementation and the parameter choices prioritize efficiency. We use SHA-256, a NIST-standardized hash function, popularly used in security applications, for the hash function. For *f* to take the 256-bit hash output and transform it to the 32-bit ID_{*t*}, we use two simple functions: *XOR*, which divides the input to non-overlapping segments of the output length and then applies the bit-wise XOR, or *LSB*, taking the least significant bits of the input. We implement a proof-of-concept of ID-RZ, running scripts in Python 3.7 on one physical machine with Intel(R) Core(TM) i5-1135G7, 2.40 GHz, 4-core CPU, 20 GB, and 3200 MHz RAM.

9.1.2 | Computing performance

We measure the computing latency, T_C , and resource cost of ID-RZ and present the results, which are processed over 1000000 samples, in this section. Figure 4A shows the computing latency, T_C , with a breakdown over H, f , and the overhead. The *overhead* includes the remaining scheme implementations, such as memory access, variable assignments, and other arithmetic operations. This overhead applies to all computing operations, including generation and verification. Our experiments reflect our design choices to prioritize the mechanism efficiency; for example, when the RAM computation is below 0.4%, the storage takes $4 \cdot n$ bytes (4 kB when $n = 1000$), and the overall T_C ranges from hundreds of microseconds to one microsecond. Temporary ID generation using the LSB for f is 126.61 times faster than using XOR. When f based on XOR is used, it becomes the dominant time cost factor, accounting for 66.7% of the time cost ($0.667 = 144.177/216.001$). However, when LSB is used for f , its time cost only accounts for 5.1% of the overall time cost.

9.2 | Networking gains from ID randomization over ID re-allocation

9.2.1 | Networking implementation

Our application-specific setup

We build a cloud server to run as a core network for our experiment. The configuration of the cloud server is an E2 general-purpose model in Google Cloud Services with Ubuntu 16.04 server version 64 bit, 2 v CPUs (Intel Xeon scalable processor 1st generation 2 GHz), and 8 GB memory. The cloud server performs ID_t generation and verification. We run our experiments on a computer (the client) connected through 5G cellular networking to communicate with the cloud server and collect our application-specific networking latency $2T_N$ and verification latency T_V using our- application -specific scripts in Python 3.7. We use the two most popular cellular operators in the country for the experiment, T-mobile and AT&T.

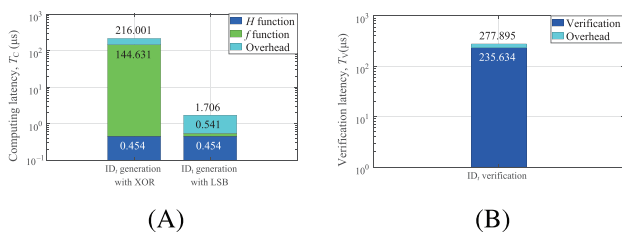


FIGURE 4 Our experimental results: (A) computing latency, T_C for ID_t generation; (B) verification latency, T_V

Networking measurements

We use *traceroute* for popular applications through cellular connections to identify the core network nodes in the real world. We collect 5000 traceroute samples for the 10 most used websites [28] and identify the core network nodes using the following two steps. First, we utilize the IP address assignment information from the Internet Assigned Numbers Authority [29] to identify the IP ranges of the cellular provider. Second, we use the traceroute samples to find the common nodes that fall in the above IP ranges. We measure the latencies for reaching and receiving responses from these nodes.

For our networking measurements, we distinguish between the baseline and ID-RZ. The *baseline* measurements inform us of our networking testing environment without ID-RZ. After the baseline measurements, we implement ID-RZ in the same networking environment. RTT is measured using traceroute, where RTT is the latency of sending packets using internet control message protocol (ICMP) from the sender to the receiver and receiving the response from the sender. Conversely, $2T_N$ is measured by sending our application-specific data (i.e., ID_t) using transmission control protocol (TCP) from the sender to the receiver and receiving the response from the sender. $2T_N$ does not include T_V . ICMP is designed for testing the network with minimal overhead and a connectionless protocol contrary to TCP, which includes overheads for connection-oriented protocol; thus, ICMP is much faster than TCP [30].

9.2.2 | Networking performance (baseline)

Figure 5B shows the probability distribution of RTT for reaching the core network using traceroute. The average RTT for reaching the core network is 175.165 ms, and the measurement range is [34.701–502.131] ms. The number of hops required 7 and 10, respectively.

9.2.3 | Networking performance (ID-RZ)

Figure 5A shows the probability distribution of our application-specific networking and verification latency measurements $2T_N + T_V$. The average $2T_N + T_V$ is 204.493 ms, while the measurement range is [125.552 – 673.295] ms for cellular operator 1. The average $2T_N + T_V$ is 184.586 ms, while the measurement range is [140.569–622.368] ms for cellular operator 2. The verification latency T_V is 277.894 μs, of which the overhead is 42.261 μs (shown in Figure 4B). T_V is significantly smaller/negligible compared to $2T_N$; in our measurements, $2T_N$ is ≈ 735 (cellular operator 1) and ≈ 665 (cellular operator 2)

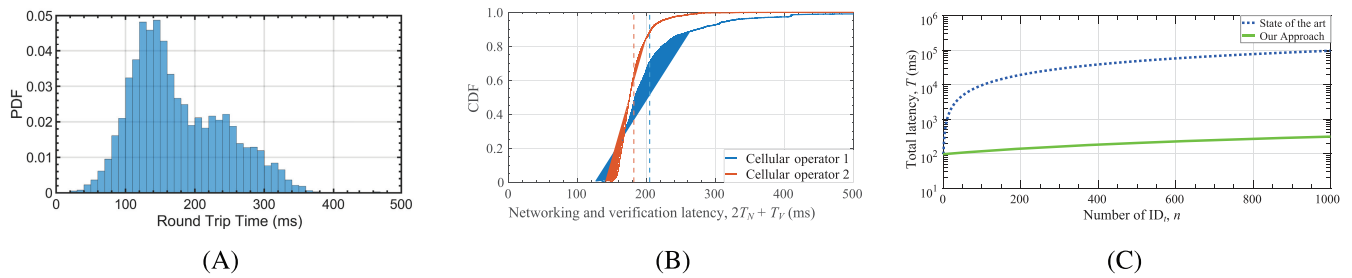


FIGURE 5 Experimental results: (A) round-trip time (RTT) measurement, (B) our application-specific networking and verification latency, $2T_N + T_V$ for cellular operators 1 and 2, and (C) comparison of total latency, T , between our proposed approach and the state-of-the-art approach

times greater than T_V . Our application-specific latency $2T_N + T_V$ is slightly higher (16.743% and 5.378% for cellular operators 1 and 2, respectively) than the RTT because of the implementation and protocol differences, including whether the TCP and ID-RZ are implemented.

We also measure the total time T , where $T = T_C + 2T_N + T_V$ (Section 4.1) and compare it with the state-of-the-art approach (Section 2) by collecting real-life measurements using the setup mentioned in Sections 9.1.1 and 9.2.1. The total latency, T , for the state-of-the-art approach is $(2T_N + T_V) \cdot n + T_C \cdot n$, as networking is needed between the core network and the user for each ID_i update. In contrast, T is $(2T_N + T_V) + T_C \cdot n$ for ID-RZ, as we eliminate the need for constant networking between the core network and the user to update ID_i . Figure 5C shows the impact of T for $n = 1000$. Our approach reduces T by two orders of magnitude for $n = 1000$ compared to the state-of-the-art approach, as it removes the need for additional communication overhead to transfer ID_i s. In ID-RZ, the increase in T as n increases is due to the increase in the computing latency, T_C , for generating ID_i s.

10 | CONCLUSION

To enhance user privacy in 5G and future mobile networking, we propose a lightweight and dynamic randomization scheme, ID-RZ, for temporary IDs. We prioritize improved security/privacy, efficiency, and compatibility with the existing 5G NR standard in designing ID-RZ. We analyze the security of ID-RZ, building on the well-established cryptographic primitives for pseudo-random ID generation and the existing security protocols in 5G. We implement ID-RZ for validation and measure the overheads of our randomization scheme in networking and computing to highlight the lightweight performances.

We implement ID-RZ in the latest 5G NR standard networking to provide a concrete design and application. However, our work can be applied to other networking

systems utilizing temporary IDs for mobile user privacy, including future 6G networking, which is yet to materialize, in principle. Thus, we envision our work to inform both current and future networking designs for securing the privacy of mobile networking.

ACKNOWLEDGMENTS

We thank the editors and anonymous reviewers for their helpful feedback. This work was supported by National Science Foundation under Grant No. 1922410 and by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. 2021-0-00796, Research on Foundational Technologies for 6G Autonomous Security-by-Design to Guarantee Constant Quality of Security; No.2021-0-02107, Collaborative Research on Element Technologies for 6G Security-by-Design and Standardization-Based International Cooperation).

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

ORCID

Arijet Sarker  <https://orcid.org/0000-0002-7911-6625>

REFERENCES

1. G. S. M., 3.20 version 3.3.2, *European Digital Cellular Telecommunication System (Phase1)*, 1991.
2. 3GPP. TS 33.501, *Security architecture and procedures for 5G System*, 2021.
3. A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J.-P. Seifert, *Practical attacks against privacy and availability in 4G/LTE mobile communication systems*, 2015. arXiv preprint arXiv:1510.07563.
4. B. Hong, S. Bae, and Y. Kim, *GUTIreallocation demystified: Cellular location tracking with changing temporary identifier*, Network and Distributed Systems Security Symposium, San-Diego, CA, USA), 2018. <https://doi.org/10.14722/ndss.2018.23349>
5. D. F. Kune, J. Koelndorfer, N. Hopper, and Y. Kim, *Location leaks on the GSM air interface*, (Network and Distributed Systems Security Symposium, San-Diego, CA, USA), 2012.

6. S. R. Hussain, M. Echeverria, O. Chowdhury, N. Li, and E. Bertino, *Privacy attacks to the 4G and 5G cellular paging protocols using side channel information*, (Network and Distributed Systems Security, San Diego, CA, USA), 2019. <https://doi.org/10.14722/ndss.2019.23442>
7. D. Rupperecht, K. Kohls, T. Holz, and C. Pöpper, *Breaking LTE on layer two*, (IEEE Symposium on Security and Privacy, San Francisco, CA, USA). IEEE, 2019, pp. 1121–1136.
8. S. Bae, M. Son, D. Kim, C. Park, J. Lee, S. Son, and Y. Kim, *Watching the watchers: Practical video identification attack in {LTE} networks*, (31st Usenix Security Symposium (Usenix Security 22), Boston, MA, USA) 2022, pp. 1307–1324.
9. H. Kim, J. Lee, E. Lee, and Y. Kim, *Touching the untouchables: Dynamic security analysis of the lte control plane*, (IEEE Symposium on Security and Privacy, San Francisco, CA, USA), 2019, pp. 1153–1168.
10. 3GPP. TR 21.915, Release 15, 2021. <https://www.3gpp.org/release-15>
11. 3GPP. TR 21.916, Release 16, 2021. <https://www.3gpp.org/release-16>
12. U. Gorrepati, P. Zavarsky, and R. Ruhl, *Privacy protection in lte and 5G networks*, (2nd International Conference on Secure Cyber Computing and Communications, Jalandhar, India), 2021, pp. 382–387.
13. T. Dittler, F. Tschorsch, S. Dietzel, and B. Scheuermann, *Ano- tel: Cellular networks with location privacy*, (IEEE 41st Conference on Local Computer Networks, Dubai, United Arab Emirates) 2016, pp. 635–638.
14. H. Nicanfar, J. Hajipour, F. Agharebparast, P. TalebiFard, and V. ictorC. M. Leung, *Privacy-preserving handover mechanism in 4G*, (IEEE Conference on Communications and Network Security, National Harbor, MD, USA), 2013, pp. 373–374.
15. Y. Wu, H.-N. Dai, H. Wang, and K.-K. R. Choo, *Blockchain-based privacy preservation for 5G-enabled drone communications*, IEEE Netw. **35** (2021), no. 1, 50–56.
16. A. Haque, V. Madathil, B. Reaves, and A. Scafuro, *Anonymous device authorization for cellular networks*, (Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks, 2021, pp. 25–36.
17. 3GPP. TS 23.003, *Numbering, addressing and identification*, 2021.
18. S. R. Hussain, M. Echeverria, I. Karim, O. Chowdhury, and E. Bertino, *5GReasoner: A property-directed security and privacy analysis framework for 5G cellular network protocol*, (Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, United Kingdom), 2019, pp. 669–684.
19. 3GPP. TS 36.321, *Medium Access Control (MAC) protocol specification*, 2021.
20. A. Shaik, R. Borgaonkar, S. Park, and J.-P. Seifert, *New vulnerabilities in 4G and 5G cellular access network protocols: exposing device capabilities*, (Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, Miami, FL, USA), 2019, pp. 221–231.
21. 3GPP. TS 36.331, *Radio Resource Control (RRC)*, 2021.
22. A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, *A statistical test suite for random and pseudorandom number generators for cryptographic applications*, Booz-Allen and Hamilton inc, McLean, VA, 2001.
23. A. W. Appel, *Verification of a cryptographic primitive: Sha-256*, ACM Trans. Program. Lang. Syst. (TOPLAS) **37** (2015), no. 2, 1–31.
24. L. Lamport, *Password authentication with insecure communication*, Commun. ACM **24** (1981), no. 11, 770–772.
25. S.-Y. Chang, Y. Park, and B. B. A. Babu, *Fast IP hopping randomization to secure hop-by-hop access in SDN*, IEEE Trans. Netw. Service Manag. **16** (2018), no. 1, 308–320.
26. A. Perrig, R. Canetti, J. D. Tygar, and D. Song, *The tesla broadcast authentication protocol*, Rsa Cryptobytes **5** (2002), no. 2, 2–13.
27. Quora, *How big of an area and how many people does one cell tower usually cover*, 2022. <https://www.quora.com/How-big-of-an-area-and-how-many-people-does-one-cell-tower-usually-cover> [last accessed March 2022].
28. Alexa, 2022. <https://www.alexa.com/topsites> [last accessed March 2022].
29. IANA, *Internet Assigned Numbers Authority*. <https://www.iana.org/>
30. L. Wenwei, Z. Dafang, Y. Jinmin, and X. Gaogang, *On evaluating the differences of TCP and ICMP in network measurement*, Comput. Commun. **30** (2007), no. 2, 428–439.

AUTHOR BIOGRAPHIES



Arijet Sarker received his BSc and MSc degrees in Information Technology from the Institute of Information Technology, Jahangirnagar University, Dhaka, Bangladesh, in 2013 and 2015, respectively. He was a lecturer in

Bangladesh at Dhaka City College and Daffodil International University from 2015 to 2016. He is currently pursuing his Ph.D in Security at University of Colorado Colorado Springs, Colorado, USA. His research interests include 5G/6G wireless security, blockchain technology, vehicular privacy, software assurance, and wireless sensor networking.



SangHyun Byun received his BS degree in engineering and mechatronics engineering from Dong-Eui University, Busan, Rep. of Korea, in 2012 and MS degree in information assurance from Regis University, Denver, CO, USA, in 2018. He is currently pursuing a PhD degree in security at University of Colorado Colorado Springs, CO,

USA. His research interests include security and privacy of vehicular network, blockchain, machine learning, and privacy of robust aggregation for federated learning.



Manohar Raavi received his BTech in Electronics and Communications Engineering from Jawaharlal Nehru Technological University Kakinada, University College of Engineering, Vizianagaram, Andhra Pradesh, India, in 2015, and his MSc in Telecommu-

nications Management from Oklahoma State University (OSU), Stillwater, Oklahoma, USA, in 2017. From 2017 to 2019, he was a Network Engineer at OSU's IT Department. He is currently pursuing a PhD at University of Colorado, Colorado Springs, Colorado, USA. His research interests include post-quantum cryptography, applied cryptography, 6G wireless security, and network security.



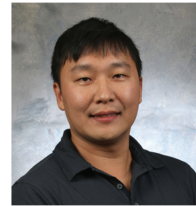
Jinoth Kim received his PhD in Computer Science from the University of Minnesota, Twin Cities. He is currently an associate professor of Computer Science at Texas A&M University-Commerce. He is also an executive team member of Silicon Valley Cybersecurity Institute (SVCSI) and an Affiliate Faculty

Scientist at Lawrence Berkeley National Laboratory. His main research interest lies in the area of networked/distributed systems, with a focus on performance, reliability, scalability, visibility, and security, using machine intelligence and data analytics. His current research projects include 6G security, network traffic monitoring, cybersecurity analytics, and scientific computing and networking.



Jonghyun Kim received his MSc and PhD in Computer Science from the University of Oklahoma, USA in 2000 and 2005, respectively. He was a researcher with Samsung Electronics from 1995 to 1997 and a system consultant with

Samsung SDS in 2000. He is currently a principal researcher with the Electronics Telecommunications Research Institute, Daejeon, Korea. He is currently working as the project leader of the Intelligence Security Group of the ETRI. He is also involved in standardization activities as a vice chair of WP1 and a rapporteur of Q.4 (cybersecurity) with ITU SG17. His research interests include information security, cyber security, cloud security, AI-based malware detection, and 5G/6G security.



Sang-Yoon Chang received his BSc and PhD from the Department of Electrical and Computer Engineering at University of Illinois Urbana-Champaign in 2007 and 2013, respectively. He worked as a post-doctoral fellow at the

Advanced Digital Sciences Center in Singapore from 2013 to 2016. Since 2016, Sang-Yoon has been with the Computer Science Department at University of Colorado Colorado Springs, Colorado Springs, Colorado, USA, where he is currently an associate professor. His research is focused on security, networking, wireless/mobile computing, cyber-physical systems, and applied cryptography.

How to cite this article: A. Sarker, S. Byun, M. Raavi, J. Kim, J. Kim, and S.-Y. Chang, *Dynamic ID randomization for user privacy in mobile network*, ETRI Journal **44** (2022), 903–914. <https://doi.org/10.4218/etrij.2022-0181>