Towards Secure 4G and 5G Access Network Protocols

vorgelegt von

Altaf Shaik (M.Sc.)

ORCID: 0000-0003-2657-6975

von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

DOKTOR DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Thomas Magedanz, Technische Universität Berlin
Gutachter:	Prof. Dr. Jean-Pierre Seifert, Technische Universität Berlin
Gutachter:	Prof. Dr. Ivan Martinovic, University of Oxford
Gutachter:	Prof. Dr. Abhay Karandikar, Indian Institute of Technology Kanpur

Tag der wissenschaftlichen Aussprache: 24. Juni 2020

Berlin 2020

Abstract

The security architecture of 2G and 3G mobile networks has been dramatically improved to accommodate 4G (Fourth Generation, *a.k.a* Long Term Evolution (LTE)) security requirements. As generations evolve, security improvements address previously known vulnerabilities, esp. in terms of user privacy. Thus, there have been substantial efforts to protect user plane traffic by using robust encryption algorithms over the LTE access network. Contrarily, the *control plane remains vulnerable despite its security enhancements*. Especially, the radio and subscriber management protocols have not evolved for the last two decades in mobile networks. By design, these protocols are allowed to operate without any security measures to minimize overheads in the system. Such design choices made by the standard body Third Generation Partnership Project (3GPP) are justified as a trade-off between conflicting requirements such as security and availability and performance. These justifications remain valid, considering that telecommunication systems have traditionally been proprietary, expensive, and efforts to mount attacks against them were challenging.

Today, the *proliferation of inexpensive radio hardware and open-source cellular software* has changed the threat landscape in mobile networks. In this context, our research practically investigates the validness of LTE security trade-offs. For this, we develop a low-cost experimental testbed to mount different types of wireless attacks and evaluate their feasibility and impact on commercial LTE devices and networks. We discover several new vulnerabilities in the access network protocols that jeopardize the privacy and availability aspects of the system. We also identify bad security practices in end-user devices and the operator's infrastructure that catalyze our attacks and further, amplify its consequences.

Our findings indicate that the *equilibrium points in the trade-offs have changed today* compared to where they were when designing the LTE security architecture. Also, the security margins to protect against trade-off changes being too narrow demonstrates the lack of resilience in LTE networks. Unlike jamming or other types of Denial-of-Service (DoS) attacks, ours are *stealthy and remain active on end-user devices for a prolonged period*.

To this extent, we responsibly communicated our research findings to the relevant standard bodies, operators, and baseband vendors. We emphasize that the justification for these trade-offs is no longer valid and violates LTE security requirements. Thus, we propose mitigations to restore privacy and availability aspects of the system, and fortunately, they are *enforced into 4G and 5G specifications* and also into worldwide operational devices and networks. We recommend that safety margins introduced into future specifications should incorporate greater agility and flexibility to maintain a stable trade-off equation.

Zusammenfassung

Die Sicherheitsarchitektur von 2G und 3G Mobilfunknetzen wurde stark überarbeitet und verbessert, um den Sicherheitsanforderungen, die an 4G (Fourth Generation, auch bekannt als LTE (Long Term Evolution)) gestellt werden, gerecht zu werden. Mit jeder neuen Generation wurden die Sicherheitsstandards verbessert, um neu bekanntgewordene Sicherheitslücken zu beheben, insbesondere im Hinblick auf die Privatsphäre der Nutzer. Es wurden erhebliche Anstrengungen unternommen, um die LTE-Userplane bei der Übertragung im LTE Netz mittels robuster Verschlüsselungsalgorithmen zu schützen. Im Gegensatz hierzu ist die Controlplane, trotz einiger Sicherheitsverbesserungen, weiterhin verwundbar. Insbesondere der Betrieb von Funk- und Abonnentenmanagementprotokollen hat sich für Mobilfunknetzte in den letzten zwei Jahrzehnten nicht weiterentwickelt. Diese Protokolle sind ohne notwendige Verschlüsselung konzipiert, um die Systemlast minimal zu halten. Diese Entscheidungen werden durch das Standardorgan 3GPP (Third Generation Partnership Project) getroffen und werden mit einem Kompromiss zwischen den widersprüchlichen Anforderungen der Sicherheit und Verfügbarkeit auf der einen, und der Leistung auf der anderen Seite, begründet. Diese Rechtfertigungen sind weiterhin gerechtfertigt, da Telekommunikationssysteme traditionell proprietär und teuer sind und Angriffe gegen Telekommunikationssysteme eine Herausforderung darstellten.

Heutzutage hat sich angesichts der Verbreitungen günstiger Funk-hardware und Open-Source Mobilfunk Software die Bedrohungslage in Mobilfunknetzen stark verändert. Vor diesem Hintergrund untersucht unsere Forschung praxisnah die Gültigkeit der Trade-Offs im Design der LTE Sicherheitsarchitektur. Hierzu entwick-

eln wir eine günstige experimentelle Testumgebung, um verschiedene drahtlose Angriffe durchzuführen und ihre Auswirkung auf kommerzielle Geräte und Netzwerke zu evaluieren. Wir entdecken eine Reihe an bis dato unbekannten Sicherheitslücken in den LTE Zugangsprotokollen, welche die Vertraulichkeit und Verfügbarkeit der Systeme gefährden. Darüber hinaus identifizieren wir sicherheitskritische Praktiken bei Implementierungen für Endbenutzergeräte und in der Infrastruktur der Netzbetreiber, die von Standard Sicherheitsverfahren abweichen und dadurch unsere Attacken beschleunigen und ihre Konsequenzen erhöhen.

Unsere Ergebnisse deuten darauf hin, dass sich das Gleichgewicht zwischen Sicherheit und Leistung, im Vergleich zu dem Stand, als die LTE Sicherheitsarchitektur entworfen wurde, verschoben hat. Desweiteren stellen sich die Sicherheitsspielräume, die gegen Veränderungen der Trade-Offs schützen sollen, als zu gering heraus. Anders als bei Jamming- und Flooding-basierten Angriffen laufen unsere Angriffe im Generellen unbemerkt ab und können für einen längeren Zeitraum auf dem Endbenutzergerät aktiv bleiben.

Wir haben unsere Forschungsergebnisse verantwortungsbewusst an die relevanten Standardorganisationen, Betreiber und Basisbandanbieter gemeldet. Wir betonen, dass die eingegangenen Kompromisse nicht mehr zu rechtfertigen sind und eine die LTE Sicherheitsanforderungen nicht erfüllen. Wir schlagen daher Maßnahmen vor, um die Aspekte, welche die Vertraulichkeit und Verfügbarkeit des Systems betreffen, wiederherzustellen. Glücklicherweise werden diese Maßnahmen in den 4G und 5G Spezifikationen sowie in den weltweiten Netzen der Betreiber umgesetzt. Wir empfehlen, dass Sicherheitsspielräume, die in künftige Spezifikationen aufgenommen werden, eine größere Agilität beinhalten sollten, um späteren Änderungen des Kompromissgleichgewichts Rechnung zu tragen.

Publications Related to this Thesis

The work presented in this thesis resulted in the following peer-reviewed publications:

- *New Vulnerabilities in 4G and 5G Cellular Access Network Protocols: Exposing Device Capabilities*, Altaf Shaik, Ravishankar Borgaonkar, Shinjo park, & Jean-Pierre Seifert, In the proceedings of 12th ACM Conference on Security & Privacy in Wireless and Mobile Networks, 2019 (ACM WiSec).

- *New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols*, Ravishankar Borgaonkar, Lucas Hirschi, Shinjo Park, & Altaf Shaik, In the proceedings of 20th Privacy Enhancing Technologies Symposium, 2019 (PETS).

- *On the Impact of Rogue Base Stations in 4G/LTE Self Organizing Networks*, Altaf Shaik, Ravishankar Borgaonkar, Shinjo park, & Jean-Pierre Seifert, In the proceedings of 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, 2018 (ACM WiSec).

- *Practical Attacks Against Privacy and Availability in 4G/LTE Mobile Communication Systems*, Altaf Shaik, Ravishankar Borgaonkar, Asokan N, Valtteri Niemi, & Jean-Pierre Seifert, In the proceedings of 23rd Annual Network & Distributed System Security Symposium, 2016 (NDSS).

Contents

1 Introduction

1.1 Motivation and Problem Statement

Mobile communication is an important cornerstone in the lives of the vast majority of people and societies on this planet. During the past two decades, mobile devices such as smartphones have become ubiquitous. The reach of mobile communication systems, starting from the second generation Global System for Mobile Communications (2G/GSM) and the third generation Universal Mobile Telecommunication Systems (3G/UMTS), has extended to every corner in the world. The latest generation in this evolution, the 4G/LTE system, is a dominant technology today, surpassing half of the global mobile connections in 2019 [66]. At the time of writing this thesis, the world is awaiting the commercial launch of fifth-generation (5G) networks that aim for a fully connected digital society.

A mobile communication system primarily constitutes the following entities: mobile device, base station, and core network. The evolution of these systems specified by 3GPP has incorporated not only improvements in functionality but also strengthened security. Early 2G systems were known to have several vulnerabilities. A major design flaw was the lack of mutual authentication between mobile devices and the network, which implied that an adversary could set up a fake base station and convince legitimate mobile devices to connect to it. Interception and DoS attacks are possible once the device is using the fake network. Later, 3G security design addressed [1] the issue of false base station attacks by the use of mutual authentication. In this, both the mobile device and network authenticate each other through the Authentication and Key Agreement protocol (AKA) protocol [1]. Also, the use of stronger and well-analyzed cryptographic algorithms was introduced to enhance user privacy over the access network. 3G and 4G specifications consider user identity and location confidentiality and user untraceability as explicit privacy requirements, as stated in [1, 23]:

[28]: *"Subscription privacy deals with various aspects related to the protection of subscriber's personal information, e.g., identifiers, location, data, etc."* [3GPP]

To achieve these requirements, LTE specifications further strengthened signaling protocols by requiring authentication and encryption in more situations than was previously required. Separate security domains to protect both user and signaling plane traffic was adopted in LTE. Consequently, there is a general belief that LTE specifications provide strong privacy and availability guarantees to mobile users. Previously known attacks, such as the ability to track user movements, were thought to be difficult in LTE.

While designing security systems, a holistic approach is required rather than an isolated one. LTE security design followed the later where user authentication and encryption protocols have received more scrutiny than others that did not evolve at par with the security enhancements made in LTE. Notably, the signaling plane protocols that perform radio and subscriber management have relatively unchanged in the history of mobile networks. There are conscious exceptions in the LTE standard that allow the operation of these protocols without any security procedures. These exceptions tend to violate the above-mentioned LTE security requirements in terms of privacy and availability.

From the design perspective, security exceptions are justified and perceived in two ways. First, to maintain backward compatibility with earlier generation networks and second, as a trade-off design between security and other system requirements such as availability and performance. In any system design, trade-offs are universal and inevitable because there are no best solutions independent of specific needs, objectives, and values. By enforcing security to radio and subscriber management protocols, overheads caused due to poor radio conditions and communication delays may affect the performance and availability of the network. Hence, during the security design, 3GPP has concluded that radio and subscriber management protocols should be operated regardless of the security state between mobile devices and networks to guarantee performance and availability to the subscribers.

Although to a certain degree 3GPP is aware of the threats resulting from such a design [17], mitigation is applied in the standards only when the impact of an attack outweighs the cost of doing it and the cost of implementing countermeasures for it. During the design of LTE security architecture, mobile network systems remained proprietary, and their exploitation is considered challenging. Only law enforcement agencies possessed such attacking equipment (a.k.a International Mobile Subscriber Identity (IMSI) catchers or fake base stations) [74] that is very expensive and difficult to obtain as a commercial off-the-shelf product. Hence, LTE security design safely ignored threats from such devices [17].

The advancements in digital electronics and open hardware solutions have contributed to the developments of low-cost Software Defined Radios (SDRs). This has completely transformed the threat landscape in cellular networks. Thus, today, it is relatively simple to transmit and receive over wireless channels using a programmable SDR. Similarly, on the software side, the open-source community noticed a steady development of cellular protocol stacks right from GSM in 2009 to LTE today. These developments have culminated in various projects such as `OpenBSC` [161], and `OsmocomBB` [123] for 2G technology and `OpenLTE` [41] and `SRSLTE` [149] for 4G technology. They are regarded as the publicly available counterparts of the radio protocol stacks (referred to as baseband) that remain proprietary.

Although, on the one hand, such advancements promote extensive cellular research among the community, on the other hand, *we would like to question the resistance of LTE security architecture against the misuse of these increased hardware and software tools*. In particular, they can be molded into malicious network elements and interfere with end-user devices and network infrastructure. Moreover, such expensive IMSI catchers can now be easily built using widely available hardware [52] or even with Wi-Fi technology [118].

Even though security protocols are set strongly, sometimes, operational, and implementation aspects may not be conceivable during the standard design. Further, new protocols and enhancements to existing ones invite new threats that may not have been conceived during the phase of security analysis [17]. Hence, the exact consequences and impact of the attacks cannot be estimated unless investigated in practice.

In this thesis, we practically investigate the *security implications of manipulating radio and subscriber management protocols using low-cost and readily available tools*. We challenge the effectiveness of the LTE design trade-offs against the evolved and powerful threat landscape in mobile networks. Our research presents new attacks against the LTE access network protocols. It demonstrates that an adversary with low-cost radio hardware and software tools can cause significant damage and loss to not only 4G and 5G subscribers and devices but also to the operators as well. We specifically study compromising the privacy and availability aspects of the system and also discuss potential mitigations to restore them. We strive to enhance the attack resistance of LTE access network protocols that forms the basis of next-generation networks such as 5G and so-on.

1.2 Hypothesis and Methodology

The research presented in this thesis proves and demonstrates the following hypothesis:

"The ease of performing persistent wireless attacks using low-cost tools demands a change in mobile network security protocols and reconsider various design trade-offs to increase the resiliency of future networks."

To prove this hypothesis, we identify new vulnerabilities in the LTE control plane protocols, i.e., the communication between a mobile device, base station, and core network). We characterize mainly two types of threat models, namely active and passive, with attacking capabilities obtained using inexpensive hardware and readily available open-source cellular software. We develop and launch various attacks over the control plane and further apply them in our experimental studies to evaluate their feasibility and impact over the commercial LTE devices and networks. Our study uncovers numerous insecure and leaky implementations in basebands and network infrastructure that fuel the attacks and also amplify their consequences.

Experimental Research Environment: For our research, we leveraged low-cost hardware and open-source software to create a full-fledged LTE network infrastructure with complete control over the access network protocol stack.

The hardware comprises of an SDR to transmit/receive LTE signals/data over the air and is controlled by a host-based software. This software consists of LTE protocol implementation of end-user device, network-side base station, and core network entities. The software stack ranges from the physical layer to the higher management layers. Precisely, we utilized a popular SDR called `USRP B210` from Ettus Research, costing up to 1000 USD to transmit and receive LTE radio signals.

We utilize multiple software in our testbed, namely `OpenLTE` and `SRSLTE`, to operate a fully functional LTE network and communicate with end-user devices and also with the commercial network entities. Further, the open-source nature allows us to introduce modifications/changes into the source code (protocol stack) to develop customized network elements. Our customized elements can impersonate a legitimate mobile network(s) and mount various wireless attacks targeting both end-user and carrier networks. We tested and validated our attacks over a broad spectrum of devices and networks. Precisely, we experimented with devices from 5 different baseband vendors and evaluated our test results in LTE networks in more than 30 countries over three continents (Europe, North America, and Asia).

Our research is supported by various operators and vendors who offered their test-network infrastructure for specific experiments. Further, we extend our tests to the latest LTE based low-powered IoT devices in the market using a custom-built Narrow Band - Internet of Things (NB-IoT) testbed. Our experiments involve transmission over the cellular interface and hence require an operational license. To abide by the telecommunication laws and regulations, we conducted our experiments under the careful guidance of an operator. We also used a Faraday cage (where required) to evaluate some of our operations. For some of our active attacks that require transmission over LTE radio frequencies, we utilized a test license from an operator who is closely associated with this research.

1.3 Scientific Contribution and Impact

Our research contributes to enhancing security and resilience in 4G and 5G over the access network. First, we practically show that the design trade-offs adopted in LTE security architecture are invalid, given the current threat landscape. Second, we emphasize that performing attacks over mobile network infrastructure has become relatively straightforward in the last years. We prove this by demonstrating various attacks over LTE networks using low-cost and readily available tools. The vast majority of our attacks are persistent and possible due to the design shortcomings of 4G and 5G access network protocols. Therefore our study affects devices and networks all over the world that are conforming to the 3GPP standards.

With our investigations, we appeal to the standard bodies for immediate revisions in the LTE security specifications to meet the desired requirements. We urge to address threats caused by both passive and active attackers and reconsider the design trade-offs to improve the attack resilience and, if possible, completely mitigate the attack with a feasible solution. Our contributions in this research include:

Security Analysis of 4G and 5G Access Network Protocols. We demonstrate several new passive and active attacks over the LTE air-interface. Our attacks are based on vulnerabilities we discovered during a careful analysis of LTE access network protocol specifications. Mainly these vulnerabilities compromise two security aspects: privacy and availability. Privacy compromising attacks include a) location privacy of subscribers: identifying if a subscriber is in the range of attacker's fake base station, and obtain the precise location b) monitoring subscriber's mobile activity and usage and c) fingerprinting mobile devices and their applications. Availability attacks include a) persistently denying LTE and all mobile services b) bidding-down to less secure networks such as 2G, c) denying and tampering specific LTE and 5G services

(including NB-IoT), and d) hijacking legitimate services. Furthermore, we highlight new vulnerabilities in automated LTE networks that mainly target the network infrastructure and poison the radio information inside the operator's management system. We prove that attacks over LTE infrastructure reveal more private information about a user or device than earlier 3GPP networks.

Security Risks in Operational LTE Networks & Basebands. We discover configuration faults in LTE network deployments over 30 countries that catalyze our attacks when exploited together with the identified protocol weaknesses. We also exhibit poor Self Organized Network (SON) implementations and their consequences in operational networks. Further, our tests reveal leaky implementations in smartphones and NB-IoT devices covering five popular baseband vendors. We set security guidelines and standard practices for 4G and 5G mobile networks to prevent our attacks despite weaknesses in the specifications.

Implementation & Evaluation. We implement our attacks using inexpensive hardware and software framework based on Universal Software Radio Peripheral (USRP) and `srsLTE`, `OpenLTE` respectively. We confirm their effectiveness using commercial LTE devices from several vendors and real LTE networks of several carriers all over the world. The range of devices and networks covers from LTE specification releases from 8 and up to 14.

Trade-off Analysis & Mitigations. We study possible considerations that may have led to the vulnerabilities. Specifically, we discuss the perceived or actual trade-offs between security/privacy and other criteria like availability, performance, and functionality, as well as recommending mitigations. Remedial actions have been applied to LTE and 5G specifications across various 3GPP releases. Further, several carriers and baseband vendors also updated their configurations and implementations [85, 84, 125] and referred our research contributions.

Impact. We reported our attacks to the manufacturers and carriers concerned as well as to the standardization body 3GPP. We followed the vulnerability disclosure program organized by the GSM Association (GSMA) that takes the responsibility to distribute our research to carriers worldwide. We received positive acknowledgments [65] for our research from all the affected parties and also confirmation about installing fixes in operational networks. Some of the vendor's patch releases are publicly available from [85, 84, 125].

Our research is referred to in several 3GPP documents [17, 22] for its contributions and practical investigations. *3GPP has enforced several updates to the LTE specifications*

to address the issues we raised in this research [34, 35, 36, 8]. A summary of our research, including discovered vulnerabilities, attacks, trade-off considerations, and mitigations that are enforced into standards and operational networks, are presented in the Table 1.1.

1.4 Thesis Structure

Based on the above-described methodology, the thesis is structured into the following chapters:

Chapter 2 presents technical concepts required to understand our research and covers LTE architecture and various network procedures. We also explore the literature work related to our study.

Chapter 3 is dedicated to build an experimental testbed using open source components and mount various LTE network entities over inexpensive hardware. Next, we sketch different adversary models and characterize them with various attacking capabilities. These adversaries perform numerous proof-of-concept attacks and evaluate LTE design trade-offs in the next two chapters.

Chapter 4 presents the LTE radio management protocols that trade security for availability of the network. We expose various vulnerabilities in their design and demonstrate location tracking and DoS attacks on SON. We evaluate their feasibility, impact, and explain the shift in the trade-off equilibrium. We close by understanding the reasons for the vulnerabilities and recommend mitigations.

Chapter 5 presents the LTE subscriber and device management protocols that trade security for the performance of the network. We expose various vulnerabilities in their design and demonstrate fingerprinting and DoS attacks. Also, we highlight a fundamental flaw in the authentication protocols used in the 3G, 4G, and 5G networks. We evaluate their feasibility, impact, and explain the shift in the trade-off equilibrium. We close by understanding the reasons for the vulnerabilities and recommend mitigations.

Chapter 6 concludes this research by briefing our contributions towards improving the security of 4G and 5G access network protocols and leave with open research questions for the future.

Attack		Adversary Type	Vulnerability		Potential fix (applied to 3GPP specifications)
Group	**Description**		**Type**	**Possible trade-off**	
Privacy Leaks	Link location over time (L1)	Passive	Underspecification	Security vs. availability	Policy to guarantee GUTI freshness
	Leak coarse-grained location (L2)	Semi-passive	Application software architecture	Security vs. functionality	Tools [158, 148] detecting suspicious signaling
	Leak fine-grained location (L3)	Active	Specification & implementation	Security vs. availability	Authentication and ciphering for measurement data
	Fingerprinting devices and applications (F1)	Active & Passive	Specification	Security vs. performance	Encrypting device NAS capabilities
	Mobile activity monitoring (F2)	Active & Passive	Specification	Security vs. performance	Symmetrically encrypt SQN
Denial of Service	Downgrade to non-LTE services (D1)	Active	Specification	Security vs. performance	Timer-based recovery
	Deny all services (D2)	Active	Specification	Security vs. performance	Timer-based recovery
	Deny selected services(D3)	Active	Specification	Security vs. performance	Extend “matching conversation" check to all parameters
	Radio Service Hijacking(D4)	Active	Specification & Implementation	Security vs. performance	Authentication and ciphering for RRC capabilities
	NB-IoT Battery Draining(D5)	Active	Specification	Security vs. performance	Extend “matching conversation" check to all parameters
	SON measurement poisoning (S1–S3)	Active	Specification & Implementation	Security vs. availability	Verify measurement information

Tab. 1.1.: Research Summary: LTE attacks, Adversaries, Vulnerability & Trade-off analysis, and Fixes

2 Background

We briefly cover the background information required to understand this thesis, especially the architecture, protocols, and operations of an LTE network. More elaborated information is available in the 3GPP specification documents referred throughout this chapter. We also study the literature work related to this thesis.

2.1 LTE Network Architecture

We describe LTE architecture and its components and their deployment across a geographical region. For this, we consider a simplified LTE architecture involving components required to set up access network protocols between a base station and mobile devices. We hide other details of the architecture which are not relevant for understanding this thesis. Figure 2.1 depicts this simplified architecture, which contains three main components: User Equipment (UE), Evolved Universal Terrestrial Radio Access Network (E-UTRAN), and Evolved Packet Core (EPC). All three components are collectively referred to as an Evolved Packet System (EPS) according to 3GPP terminology. In the interest of simplicity, throughout this thesis, we refer to the whole system as LTE. The three components are detailed below:

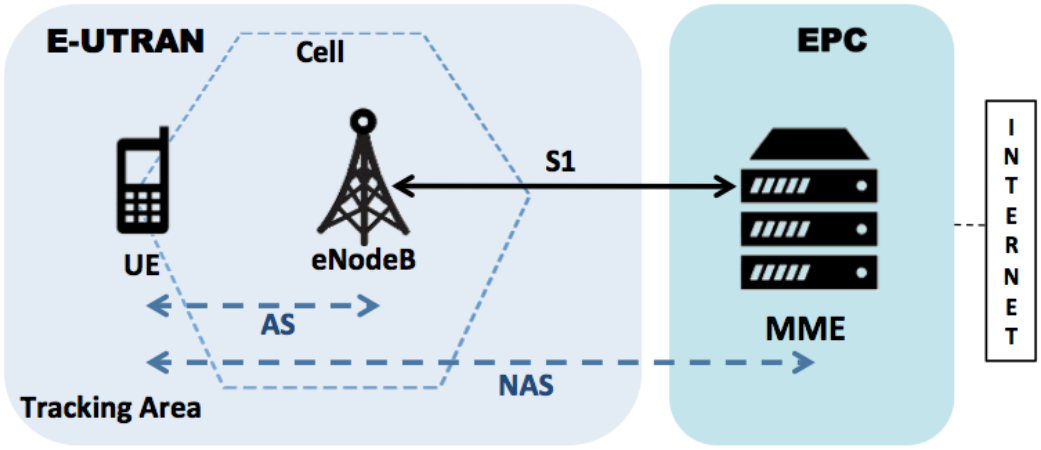

Fig. 2.1.: LTE system architecture

2.1.1 UE

UE refers to the actual communication device, which can be, for example, a smartphone or a Machine to Machine (M2M) communication device [155]. A UE contains a Universal Subscriber Identity Module (USIM) [4], which represents the IMSI and stores the corresponding authentication credentials [31]. IMSI is used to identify an LTE user (generally referred to as "subscriber" in 3GPP terminology) uniquely. The USIM participates in LTE subscriber authentication protocol and generates cryptographic keys that will subsequently be used to protect signaling and user data communication between the UE and base stations over the radio interface. USIM contains a unique, permanent secret *symmetric key* that we indicate as K_{IMSI} (used as a shared secret between a UE and its corresponding operator) and a 48-bits counter, called Sequence Number (SQN), that we denote as *SQN* (used as replay protection, as explained later in this section).

2.1.2 E-UTRAN

E-UTRAN consists of base stations. It manages the radio communication with the UE and facilitates communication between the UE and EPC. In LTE, a base station is technically referred to as evolved NodeB (eNodeB). The eNodeB uses a set of access network protocols, called Access Stratum (AS) for exchanging signaling messages with its UEs. These AS messages include Radio Resource Control (RRC) protocol messages. Other functions of eNodeB include paging UEs, over-the-air security, physical layer data connectivity, and handovers. Each eNodeB is connected to the EPC through an interface named S1. Two eNodeBs (if they are closely located) are directly connected over a X2 interface [6] over which they can perform handovers, load management, and also exchange configuration settings.

2.1.3 MME in EPC

EPC provides core network functionalities by a new all-IP mobile core network designed for LTE systems. It consists of several new elements, as defined as in [15]. However, for our work, we need to describe only the Mobility Management Entity (MME) in detail. MME is responsible for authenticating and allocating resources (data connectivity) to UEs when they connect to the network. Other important functions of MME involve security (setting up integrity and encryption for signaling) [24] and tracking UE's location at a macro level. The set of protocols run between UE and MME is referred to as Non Access Stratum (NAS).

Now, we explain how the system components presented above can be deployed in a geographical region (e.g., in a city) by mobile network carriers (more commonly referred to as "operators" in 3GPP terminology) to provide LTE services. A service area of a mobile operator is geographically divided into several regions known as Tracking Area (TA)s. TAs are similar to Location Areas in GSM networks and are managed by the MME. Further, a TA contains a group of "cells"[1] each of which is controlled by an eNodeB. A cell is identified by a combination of an E-UTRA Cell Global Identifier (ECGI) and a Physical Cell ID (PCI). ECGI is used to identify the cell on a network-wide or global level uniquely. In contrast, PCI is used to scramble over-the-air transmissions by the eNodeB to avoid interference between adjacent cells (that may operate on the same frequency). PCI has a range from 0 to 503, and thus distantly located cells can operate with the same PCI. The eNodeB broadcast operator-specific information such as Tracking Area Code (TAC), Mobile Country Code (MCC), Mobile Network Code (MNC), and ECGI via System Information Block (SIB) messages [8]. By decoding them, UEs can identify their serving network operator and establish a connection to the network.

2.2 LTE Network Procedures

We describe various protocol interactions of the mobility-related LTE procedures such as registration, paging, and handover. Also, security procedures that are responsible for providing authentication, confidentiality, and integrity in the network are presented. At last, we introduce the concept and operation of SON.

2.2.1 UE/Subscriber Registration

UE registration is one of the EPS Mobility Management (EMM) procedures [16]. At power-on, UE begins the cell search procedure where it scans available frequency bands and selects a suitable cell to establish a connection with the eNodeB. Accordingly, UE measures the signal power from the surrounding eNodeBs and chooses the strongest one. By tuning into the frequency (EARFCN) of the eNodeB, UE can listen to the Primary Synchronization Signal (PSS) and the Secondary Synchronization Signal (SSS) of the cell. PSS and SSS allow the UE to synchronize in time and frequency with the eNodeB and further decode the PCI of the cell.

Once the UE is time and frequency aligned, it can decode the broadcast messages periodically transmitted by the eNodeB. In LTE, broadcast signals are referred to as

[1] In LTE, a coverage area of an eNodeB is divided into several sectors known as cells.

System Information Block (SIB) messages, which contain essential information for the UE to identify and learn about the network. For instance, *SIB type 1* message includes MCC, MNC, TAC, and ECGI of the system, and *SIB type 2* message contains the Random Access Channel (RACH) parameters. Using *SIB type 1* message UE detects its home network (based on USIM) and using *SIB type 2* UE can initiate a connection and perform a handshake with the network. After that, UE performs registration with the network over the control plane, as shown in Figure 2.2.

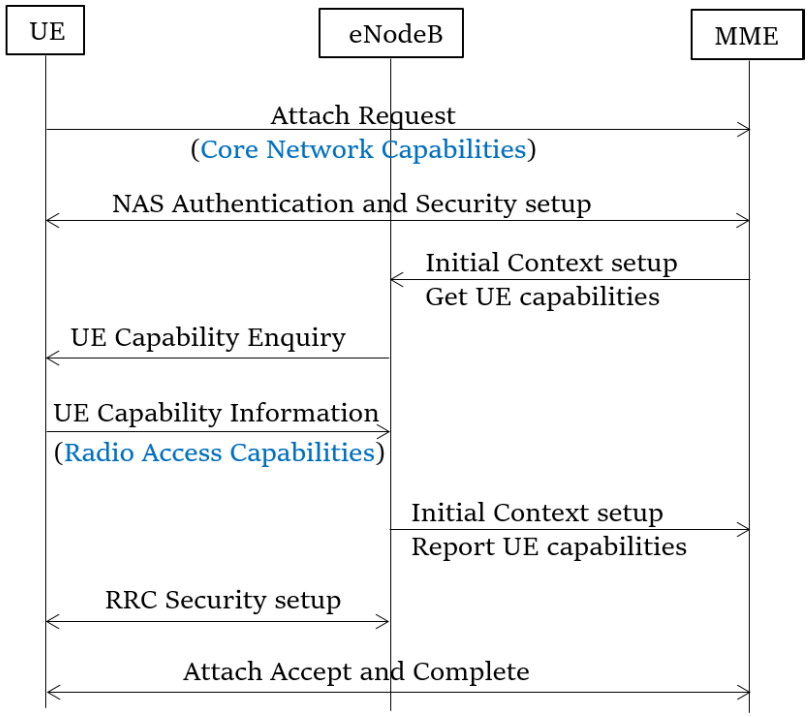

Fig. 2.2.: LTE Registration Procedure

To begin, UE sends a *Attach Request* message to the MME, indicating its request for voice/data services or both. It primarily consists of subscriber identities such as IMSI or Globally Unique Temporary Identifier (GUTI) and UE's core network capabilities. Since *Attach Request* is a first message to the network, it is transferred in plaintext. Upon identifying the subscriber, both UE and network perform mutual authentication and establish the first level of security. In particular, NAS security is established between the UE and the MME to enable encryption and integrity protection of the messages hereafter exchanged between them. Next, the MME instructs the eNodeB to fetch UE's radio access capabilities. Thus upon receiving a *UE Capability Enquiry* message from eNodeB, UE transfers the requested radio access capabilities using *UE Capability Information* message. These capabilities are forwarded to the MME and cached there until UE de-registers from the network. Further, eNodeB and UE establish a second level of security called RRC or AS security. Hereafter the messages exchanged between UE and eNodeB are encrypted and integrity-protected. Both

NAS and AS security are collectively referred as EPS security. Following this, the registration is completed when the UE receives a *Attach Accept* message in which UE is assigned a new GUTI. Now the UE can utilize telephony and data services offered by the network.

Typically a UE holds one of the two states: RRC IDLE or RRC CONNECTED. In *RRC IDLE*, state UE is not communicating with any eNodeB but is merely listening to the broadcast channels. In contrast, in *RRC CONNECTED* state UE is actively exchanging signaling and data messages with the eNodeB.

Cellular IoT UE: Two new categories of UEs known as NB-IoT and LTE-M are defined by the 3GPP in LTE Release 13 specifications to support low-powered, battery constrained IoT devices in mobile networks. An optimized registration procedure is adopted for these categories in which these UEs are required to establish only the NAS level of security and eliminate the RRC security setup. Moreover, data transmission is facilitated using secure NAS control plane messages [32].

2.2.2 Security

Security in the LTE network is achieved through various protocols interactions. Among them, we mainly discuss the authentication protocols in which we discovered logical vulnerabilities. Further, we also consider how permanent identities are protected in LTE networks.

A. Concealing Subscriber Identity

As IMSI is a permanent identifier of a subscriber, LTE specifications try to minimize its transmission in over-the-air radio communication for security and privacy reasons. Instead, a GUTI [31] is used to identify subscribers during radio communication. It is assigned to UEs during $Attach$ and may be periodically changed to provide temporal unlinkability of traffic to/from the same UE.

B. Authentication and Key Agreement(AKA)

3GPP has standardized the AKA protocol to perform mutual authentication between UE and the network and to agree on session keys that provide integrity and confidentiality protection for subsequent signaling and data plane messages [24]. While this protocol was invented in 3G, it has evolved with each generation [1, 23, 18],

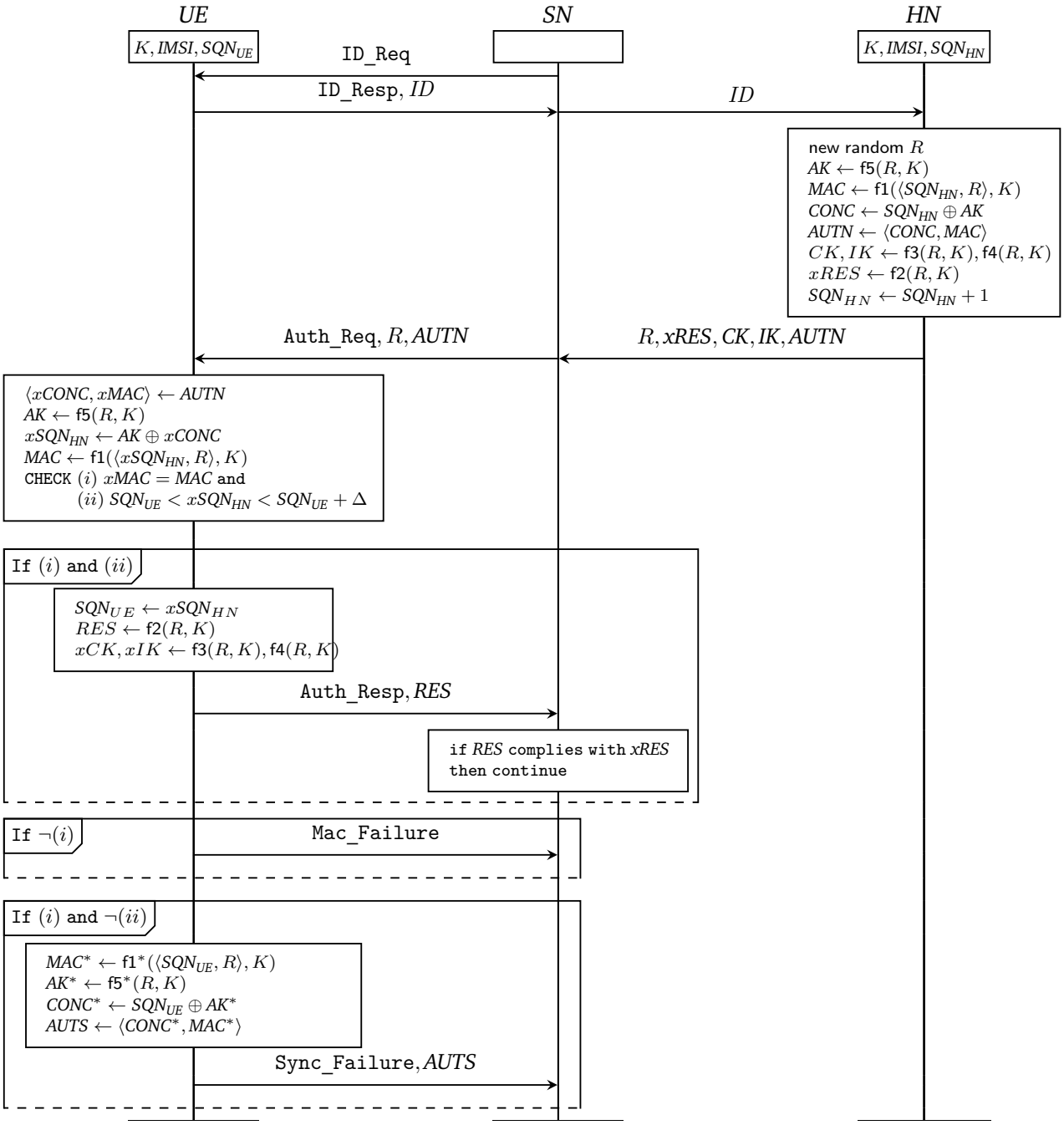

Fig. 2.3.: The AKA protocol. K denotes K_{IMSI}.

but, its core specification remained the same. We describe a simplified AKA protocol that is common to all the variants.

AKA protocol flow with message specifications is depicted in Figure 2.3. A Home Network (HN) (denoted as (*HN*)) contains a database of their subscribers and their corresponding USIM cards and is in charge of their authentication. However, in

cases that UEs are in locations where their corresponding *HN* has no base station, a Serving Network (SN) (denoted as (*SN*)) to which UEs may attach act as a relay to *HN*.

The AKA protocol achieves mutual authentication and key exchange between a UE and its corresponding *HN*, relying on some *SN* that is known by the *HN*. It allows UE and *SN* to establish session keys to be used to secure subsequent communications (*e.g.,* integrity and confidentiality of calls or SMSs). The AKA protocol is made up of 3 main phases: *identification*, *challenge-response*, and *re-synchronization procedure* (that is optional and aims at updating SQN on the *HN* side in case SQN is out-of-sync).

Identification. First, the *SN* identifies the UE. If the current UE's identity is unknown to the *SN*, it may ask for the permanent identity *IMSI* (or encryption thereof in 5G) by sending a *Identity Request* message. The UE then gives its identity in a *Identity Response* message. This identity enables the *SN* to request authentication material to the appropriate *HN* in the next phase. In 5G, UE never reveals its permanent identity in plaintext. It rather sends the randomized encryption thereof, protected with the *HN*'s public key, along with the *HN*'s identity (forming the so-called *SUCI* [18]).

Challenge-response. Upon reception of a request for authentication material from a *SN*, the *HN* computes an *authentication challenge* made of a random nonce R and some message *AUTN* called as Authentication Token (AUTN). In addition, the expected authentication response $xRES = \mathsf{f2}(R \quad \| \quad K_{IMSI})$ – computed using f2 described below–, the encryption key CK, and the integrity key IK are also computed by *HN* (but not sent by *SN* to UE). Note that, in 5G, the message $xRES$ has a slightly different form; this has no impact on our attack.

The functions $\mathsf{f1} - \mathsf{f5}, \mathsf{f1}^*$, and $\mathsf{f5}^*$, used to compute the authentication parameters, are one-way keyed cryptographic functions completely unrelated[2], and $\oplus$ denotes the eXclusive-OR (XOR) operator. *AUTN* contains a MAC (Message Authentication Code) of the concatenation of R with the corresponding sequence number SQN_{HN} stored for this subscriber. An increment of the counter generates a new sequence number.

The sequence number SQN_{HN} allows the UE to verify the freshness of the authentication request to defend against replay attacks, and the MAC proves the authenticity of the challenge. However, SQN_{HN} is not transmitted in clear text to avoid eavesdrop-

[2]Even though the specifications are not clear about the requirements of those functions [1, 23, 18], Milenage [2] and TUAK [21] schemes are used in practice, which is based on block ciphers.

ping. Thus, the specification requires *SQN* to be *concealed*; *i.e.,* XORed with a value, called *Anonymity Key*, that should remain private: $AK = \mathsf{f5}(R \parallel K_{IMSI})$. Formally, the concealed value, also included in *AUTN*, is as follows: $CONC^* = SQN_{HN} \oplus AK$ and allows the UE to extract SQN_{HN} by computing *AK*.

The UE replies with an *Authentication Response* message when the authentication is successful, or *Authentication Failure* message with the cause of failure otherwise. To check whether authentication is successful or not, the UE extracts SQN_{HN} from *AUTN* and checks that: (i) MAC is a correct MAC value w.r.t. K_{IMSI}, replies `Mac_failure` if it is not the case; (ii) the authentication request is fresh (i.e. $xSQN_{HN} > SQN_{UE}$ and $xSQN_{HN} < SQN_{UE} + \Delta$), replies `Sync_failure` $\parallel$ *AUTS* otherwise (*AUTS* is explained next). The quantity Δ is a threshold that is fixed according to an availability vs. security trade-off. If all checks hold then the UE computes the ciphering key CK and the integrity key IK and stores them to secure subsequent messages.

It also computes the authentication response RES and sends it to the *SN* using *Authentication Response* message. Only RES is included in the message; other computed values like CK and IK are not transmitted. The *SN* authenticates the UE by verifying whether the received response matches with $xRES$. If so, the AKA protocol is completed, and subsequent communications can be secured using the secret keys IK and CK.

Re-synchronization procedure. While *SQN* is expected to be synchronized between the UE and *HN*, it may become out-of-sync. We thus use SQN_{UE} (resp. SQN_{HN}) to refer to the SQN value stored in the UE (resp. *HN*). In case of a synchronization failure (case `(i) and ¬(ii)`), the UE replies with `Sync_failure` $\parallel$ *AUTS*. The *AUTS* message's purpose is to allow the *HN* to re-synchronize with the UE by replacing its own SQN_{HN} by the sequence number of the UE (*i.e.,* $SQN_{UE} + 1$).

For reasons explained above, SQN_{UE} is concealed: $CONC^* = SQN_{UE} \oplus AK^*$ where $AK^* = \mathsf{f5}^*(R \parallel K_{IMSI})$. Finally, $AUTS = CONC^* \parallel MAC^*$ where $MAC^* = \mathsf{f1}^*(K_{IMSI}, (SQN_UE \parallel R)$ allowing the *HN* to authenticate this message as coming from the intended UE.

2.2.3 Paging

Paging refers to the process used when MME needs to locate a UE in a particular area and deliver a network service, such as incoming calls. Since MME may not know the

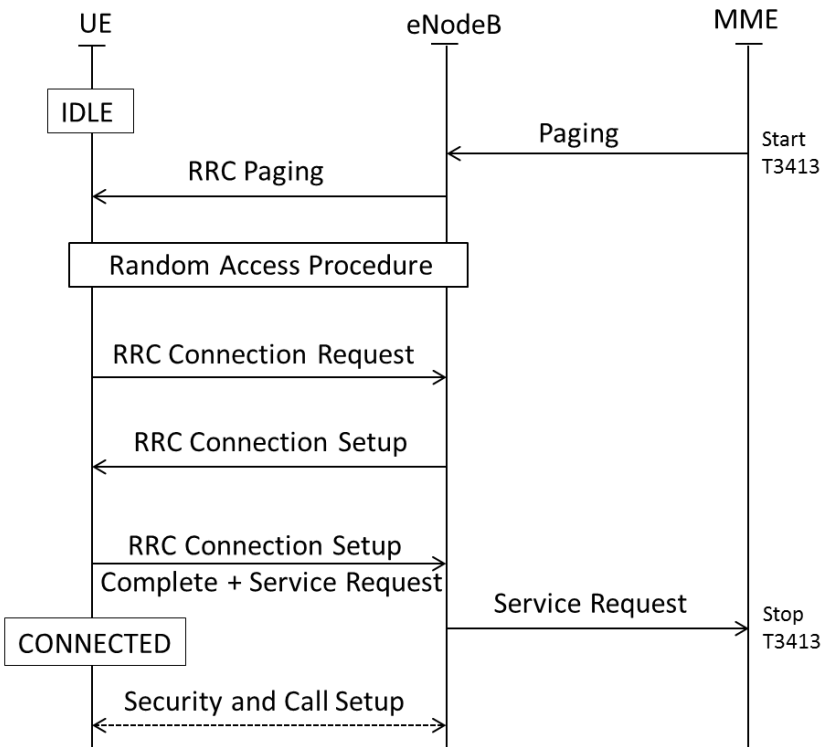

Fig. 2.4.: Paging in LTE

exact eNodeB to which UE is connected, it generates a paging message and forwards it to all eNodeBs in a TA. Simultaneously, MME starts a paging timer (T3413) and expects a response from UE before this timer expires. Thus, all eNodeBs present in the paged TA broadcast an RRC paging message to locate the UE. Paging messages contain identities of UEs such as S-TMSI(s) or IMSI(s). S-TMSI is a temporary identifier (SAE-Temporary Mobile Subscriber Identity). It is part of a GUTI. For the sake of simplicity, we consistently use the term GUTI throughout the rest of this paper, even when referring to S-TMSI. Figure 2.4 highlights LTE paging procedure, described in detail in the relevant LTE specifications [16, 10, 7].

The UE in IDLE state decodes RRC paging messages. If it detects its IMSI, then it initiates a new $Attach$ procedure to receive a GUTI as defined in [16]. If UE detects its GUTI, it acquires a radio channel through the *"Random Access Procedure*" [7] for requesting an RRC connection from the eNodeB. *"RRC Connection Setup*" involves the configuration of radio resources for exchanging signaling messages. Upon receiving this setup message, the UE completes a three-way RRC handshake procedure by sending a *"RRC Connection Setup Complete*" message along with a *"Service Request*" message. At this point, UE leaves the IDLE state and enters into CONNECTED state[3].

[3]CONNECTED means the UE has an active connection with an eNodeB.

The eNodeB forwards the service request message to MME, which in turn stops the paging timer. Further, eNodeB establishes a security context and proceeds to deliver network services to UE.

In LTE, the paging procedure is improved to reduce signaling load and locate the UE faster using a technique called Smart Paging [107, 46, 116]. It is compliant with LTE specifications and consists of directing paging messages selectively via the eNodeB (cell) where the UE was seen the last time. If no response is received, paging is repeated in the entire TA. In our experiments (Section 4.2.1) to study LTE paging procedures in a major city, we observed that several network operators and vendors had implemented smart paging.

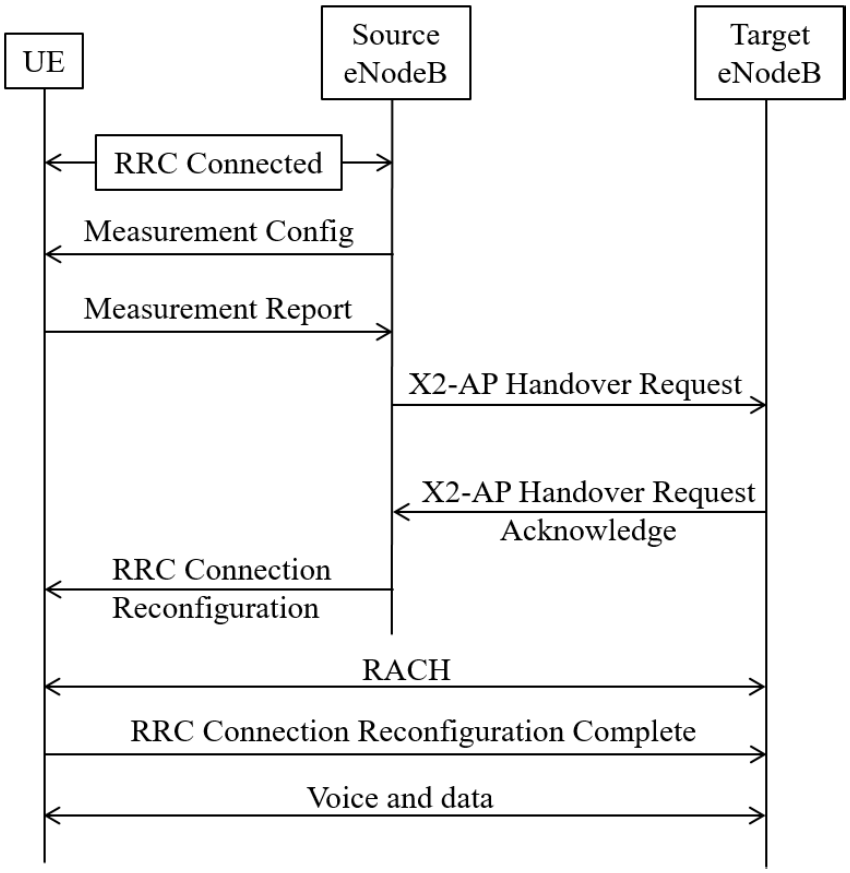

Fig. 2.5.: LTE Handover

2.2.4 Handover

Handover is a technique to ensure that UEs can freely move across cells while enjoying high-quality communication services. The eNodeB is responsible for managing the UE movement across different cells. An overview of the LTE handover procedure

over the X2 interface is shown in the Figure 2.5. Once UE enters into the CONNECTED mode, it receives a *measurement configuration* comprising of EARFCN(s) and certain trigger conditions. When these conditions are satisfied, for instance, UE's received power level from the source eNodeB is falling below a threshold, UE triggers a *measurement report* to the source eNodeB in a periodical fashion. This report contains PCIs and the corresponding power measurements of all the neighboring cells visible to UE on the configured EARFCN(s).

Based on the measurements, source eNodeB chooses the best-reported target cell to initiate a handover. Accordingly, source eNodeB sends a *X2-AP Handover Request* message to target eNodeB and prepare for a handover. This message contains the information required to perform a handover such as UE security context, capabilities, etc. Based on the current load conditions, the target eNodeB reserves resources for the UE and returns a *X2-AP Handover Request Acknowledge* message as an acknowledgment (willing to accept the UE) along with pre-allocated RACH information for the UE. Next, the source eNodeB issues a handover command to the UE with a *RRCconnection reconfiguration* message, which includes the target eNodeB information such as its PCI and EARFCN and RACH information. By using the PCI and EARFCN alone, UE instantly detects and synchronizes with the target eNodeB and acknowledges a successful handover procedure by sending a *RRC Connection Reconfiguration Complete* message to the target eNodeB. Later, UE and target eNodeB resumes the ongoing voice/data session.

2.2.5 Self Organized Network (SON)

LTE-A networks are designed to be self-organized in which the network can self-configure and self-optimize to improve its performance. SON also reduces operational and maintenance costs of the mobile network operators. To support SON in multi-vendor environments, the 3GPP has developed SON related standards for radio-access equipment such as eNodeB [29].

SON standards include a combination of self-configuration, self-healing, and self-optimization techniques for network management. SON functionalities are typically implemented in the Operation, Administration, and Maintenance (OAM) system or the network elements such as eNodeBs [105]. Depending on SON implementation, the SON architecture can be categorized into three types: centralized, distributed, and hybrid. In centralized, SON functionality is implemented in the OAM, whereas in distributed, it is implemented in the network elements. In the hybrid, SON functionalities are shared among the OAM and the network elements. The primary

source of intelligence for SON functions is measurement information received from the deployed base stations and end-users devices such as phones [6].

SON Operation

SON is typically a software package that is either directly installed on each eNodeB or centrally controlled from the OAM. As per [6], SON majorly constitutes of three phases, namely self-configuration, self-optimization, and self-healing [11].

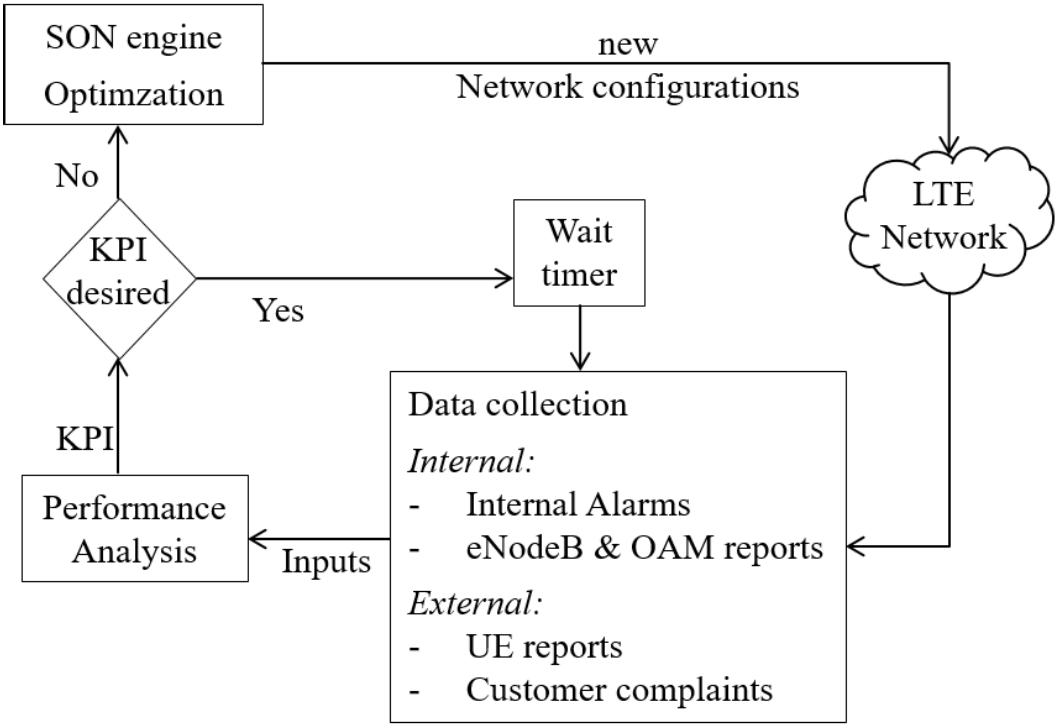

Fig. 2.6.: SON methodology

Self-configuration. It is the process where newly deployed eNodeBs automatically acquire configuration details from the OAM and neighboring eNodeBs and become operational.

Self-optimization. During the operational state, UE, eNodeB, and measurements and other performance measurements are used to auto-tune the network settings.

Self-healing. It is the mechanism that automatically detects and resolves failures in the network elements, e.g., cell outage detection and compensation. After the fault is repaired, all the parameters are restored to their original values.

Once an eNodeB self-configures and becomes operational in the network, its performance is continuously monitored and optimized by a SON software. SON implements a reactive methodology where optimizations are triggered when a problem is detected in the network. A classic SON methodology is depicted in Figure 2.6, where the SON engine controls LTE network configurations through various optimizations. Network problems or failures are detected through multiple internal and external data collection methods. Among them, network alarms, eNodeB, and OAM reports are internally available in the network, and UE reports and user complaints are collected externally. This data is supplied as an input to assess the network performance and generate a Key Performance Indicator (KPI). When the current performance is deviating from the desired KPI, the SON engine automatically issues a decision to perform suitable optimizations and alter the network configurations. For instance, improper network planning can cause voice calls to drop at the cell edges due to the coverage gap between the cells. In such a case, the SON engine detects KPI drops and decides to increase the transmit power to extend the signal coverage and close the gaps. In contrast, when the KPI is above a pre-defined threshold, the performance is recalculated after a specific wait time.

2.3 Related Work

In the last years, several attacks against cellular devices and infrastructure have been published. To the best of our knowledge, our work constitutes the *first publicly reported practical* attacks against LTE access network protocols. Following this, other research works utilized similar open source components to identify vulnerabilities in LTE networks. Some researchers have created frameworks to verify implementation and standard weaknesses in both end-user devices and also network infrastructure. An overall systematic analysis of known attacks on cellular networks together with their underlying causes and mitigations are presented in [136]. In this section, we mainly study two categories of related work a) privacy leaks and b) DoS, and the rest covers the framework based security analysis.

2.3.1 Privacy Attacks

Privacy compromising attacks target on leaking subscriber's personal information such as location, identities (permanent or temporary), and as well as interception of data/voice. For instance, although IMSI transmission in plaintext is allowed in LTE networks, 3GPP restricted International Mobile Equipment Identity (IMEI) transmission only in a security-protected message. However, certain baseband

implementations reveal [134] the device IMEI to rogue base stations. Although the work highlights leaky baseband implementations, we rather focus on standard-based vulnerabilities that have more impact.

Previous works have reported attacks against 2G and 3G access network protocols [96, 98], core network protocols [58, 157, 131, 156], as well as services [50]. In passive attacks, Kune et al. [96] showed that despite the use of temporary IDs, the location of a subscriber's UE in a GSM network could be leaked. In specific, it is shown that an attacker can check if a UE is within a small area, or absent from a large area, without subscriber's awareness. However, their location leaks granularity is lower, and it is improved with our attacks on LTE networks. The 3GPP discusses a set of threats exposed in E-UTRAN [17] during LTE security study. However, the attacks we presented are not identified by the study.

In the active type of attacks, [40] presents a method to determine the presence of a subscriber in a particular area by exploiting a vulnerability in 3G AKA protocol. By leveraging a rogue eNodeB (femtocell), previously captured authentication parameters are replayed to the UE, and the presence is confirmed based on the response from the phone. However, their attack cannot reveal the approximate location of a UE in a given area.

In [138], authors perform DNS hijacking over the LTE air-interface using a Man In The Middle (MITM) relay. In particular, the relay tampers the data traffic that is not integrity protected. Although our experimental setup is similar, our attacks do not involve any cryptanalysis and are easier to perform. Further, this problem is addressed and fixed in 5G networks; hence, they do not apply to 5G networks.

In [108, 117], authors present device type identification techniques using MAC layer information and network interactions for IP-enabled IoT devices or cellular devices connected over wired Ethernet or WLAN interfaces. Further, they also pinpoint vulnerable devices based on the data from vulnerability databases. They perform numerous experiments with real-world off-the-shelf IoT devices. Unlike theirs, our research focuses on devices with cellular capabilities and hence applies to the latest mobile IoT technologies introduced in the last couple of years. Moreover, we do not use private identifiers such as IMEI or MAC addresses for identification but determine the device type using its features. Most importantly, our identification technique also detects a wide range of devices on 5G networks.

2.3.2 Denial of Service Attacks

Based on our literature survey, the possible DoS attacks in LTE networks can be categorized into two types: Jamming attacks that directly disrupt the network signals and signaling attacks that exploit protocol vulnerabilities. Although signal jamming can be one of the most straightforward attacks, its effect is neutralized once the jammer is switched off. Hence, 3GPP did not standardize any mitigations against jamming type attacks and left it for implementations.

Through simulations, the authors in [88] show that botnets can cause DoS attacks by exhausting subscriber traffic capacity over the air interface. A theoretical paper from P. Jover et al. [130] provides an overview of DoS attacks (smart jamming) that extend the range and effectiveness of basic radio jamming. However, according to to [55], both aforementioned flooding and jamming attacks are non-persistent DoS attacks hence not considered a serious threat to address in the LTE architecture. In contrast, our DoS attacks are persistent and targeted towards the UEs (subscribers).

LTE security architecture and a detailed list of security vulnerabilities existing in the LTE networks have been presented in [104]. Our attacks are not shown in this survey. Two recent papers [93, 99] discuss resource stealing and DoS attacks against VoLTE, whereas our focus is against LTE access network protocols. To the best of our knowledge, there was no previous work evaluating practical attacks on LTE access networks in the literature.

In [100, 133], authors propose to disrupt (jam) the PSS and SSS that are essential for any UE to synchronize with the eNodeB. Hence all UEs in the range fail to connect to the network. However, UE can regain network services instantly when the jammer is turned off. Also, in [97], authors perform RF spoofing to cause DoS attacks to UEs. In contrast, our attacks inject unreliable information into the network and target SON operations resulting in a persistent DoS. Moreover, our attacks are controllable and can be targeted to specific UE(s). Prior works [77] demonstrated DoS attacks arising from the exploitation of LTE signaling protocol messages directly exchanged with UEs. Differently, our DoS attacks impersonate a legitimate eNodeB and do not require any exchange of messages neither with the UE nor with the network. This keeps our attack simple, low-cost, and stealthy.

In [156], the authors analyzed the impact of cellular botnets (malicious devices) on the mobile network by generating heavy signaling through compromised UEs. However, such hostile traffic can be detected with the latest techniques being standardized in 5G SON [140]. A study in [70] reveals that the 911 emergency service

system is susceptible to DDoS attacks in which the attackers repeatedly issue calls to 911 with anonymized phones with rogue identities. Based on simulations, the attackers prove that with less than 6K bots, an entire state in the US can be denied emergency calls. In contrast, our attacks are simple to perform and deny regular voice calls and data to all the subscribers. We analyze rogue base station threats and their impact on automated LTE networks.

In [168], the authors perform signal injection where the attacker modifies the SIB messages from a legitimate network over the LTE air-interface. Since they are broadcast in nature, UE cannot detect their modification. The attacks are feasible since the UE decodes the stronger signal when multiple simultaneous wireless signals collide in the air. It is valid for signals with a slight power difference of 3 dB [110]. The attack involves several technical challenges due to strict requirements of transmission timing and frequency for precise overshadowing the legitimate signal. Although the vulnerability is not novel, a proof of concept attack is developed in this work.

2.3.3 Security Testing Framework

LTE protocol stack implementations comprise millions of lines of code and thousands of state machines. Mostly this is deployed on proprietary equipment and lacks security testing. Also, the community lacks any access to such carrier-grade equipment and hence, hinders any security research.

A semi-automated tool called LTEFuzz is designed in [92] that generates test cases to fuzz LTE network infrastructure. This is carried out by customized open-source LTE UE side application. The resulting device-side logs are analyzed to determine the vulnerable behavior of the network. Their analysis is wholly based on the device-side records, and hence it can be regarded as partial vulnerability analysis of the network infrastructure. Similarly, in [137] developed a framework to evaluate the implementations of security functions such as authentication and encryption in LTE end-user devices. This study uncovered vulnerabilities in multiple baseband chipsets that violate 3GPP specifications.

Another framework called LTEInspector [86] was developed to identify vulnerabilities in the LTE mobility management procedures. The tool uses open source components and combines a symbolic model checker and a cryptographic protocol verifier in the symbolic attacker model. The use of such a model checker against cellular protocols is novel. Their attacks reveal the coarse-grained location of targets and DoS attacks. In contrast, our attacks show the fine-grained location of the user.

Some of the attacks are already known in the community and are inherent from older generation networks. Further, they are not persistent, and UE can recover immediately from the attack.

In contrast to these works, our approach identifies design issues in the 3GPP standard and analyses the underlying reasons behind such implementation and also any security-related decision taken by the 3GPP. These frameworks identify vulnerabilities that are implementation-specific to a particular UE or eNodeB vendor. Hence, the impact factor is significantly smaller.

Solutions described in these attacks require significant architectural changes in the cellular network. For example, the adoption of public key infrastructure for LTE broadcast messages. However, this is not a feasible solution for today's mobile networks [17] and will require massive investment for operators and SIM manufacturers. Further, Hence, these trivial attacks that build on exploiting architectural issues in mobile networks are not novel to the industry. They exhibit a unique way of performing the attack but exploit the same root cause every time. Differently, our research identifies problems that are not trivial and also fixable with minimal costs and updates. Further, they are not obvious problems due to broadcast messages. They highlight design issues that correctly have traded off security for various other requirements.

2.3.4 Low-Powered IoT attacks

Capability modification attacks by a MITM are proposed in low-powered wireless networks. Capability exchange during Bluetooth pairing procedure is presented in [71, 72, 152] and LoRa have *spreading factor* which changes bit rate and power consumption [141], but unlike LTE it is static configuration. Besides, Sigfox [143] has a different security model where MITM is not feasible and is not affected by this attack unless a cellular network is used as a backhaul link.

3 Low Cost Tools: LTE Testbed and Threat Modelling

Attacks over the mobile network can originate from different interfaces. For instance, from the operators interconnections such as SS7 [154] and Diameter [75], using compromised Femtocells [58], from the access network using fake base stations (aka. IMSI catchers) [74, 56] or compromised mobile devices (botnet type attacks) [156], and etc. Among them, IMSI catcher based attacks are expensive due to their closed-source hardware and software. Traditionally, they were operated by law enforcement agencies or state-sponsored actors with access to carrier-grade equipment that cost millions of dollars. The notorious IMSI catching attack where the end-user device would reveal its identity (IMSI also IMEI) to a rogue network is applicable from 2G up to 4G network and is a threat to subscriber privacy.

3GPP has studied various mitigations to conceal the IMSI during the LTE security design [17] process. But, implementing a solution did not seem to justify the cost of complexity involved in making the attack. Similarly, other known vulnerabilities, especially over the control plane, are not mitigated in LTE standards. Further, the cost of standardizing a solution for the control plane always outweighed the cost of performing an attack. Hence, network availability and performance have won the trade-off design over the security of the system. Besides, radio jamming is also a severe threat in mobile networks, but 3GPP did not add any countermeasures to it since the attack is not persistent; the network recovers as soon as the attacker stops jamming.

Within a year after the LTE system was standardized, the first IMSI catcher was publicly demonstrated at DEFCON in 2010 by Chris Paget [94] using low-cost hardware and open-source software. This was possible due to the evolution of cheap SDRs, together with a combination of leaked source codes, hardware documentation, and a well-trained open-source community, that has broken the barriers to telecommunications security research. Consequently, this developed into open-source software for various network elements right from the mobile device to core network components. All of them can be operated using cheap and readily available SDRs to set up a GSM network [161].

Thereafter security investigations accelerated in the field of mobile networks. Researchers and enthusiasts have started to learn and apply the tools for practical research activities and discover vulnerabilities in handsets and network infrastructure. In particular, customized `OsmocomBB` [123] tools were leveraged to perform passive interception of voice calls on the GSM network by exploiting weaknesses in A5/1 encryption algorithm [113, 114]. Such practical demonstrations accelerated the operators to update their networks to use A5/3 instead of A5/1 algorithms. Although some base stations today still use A5/1, which may be regarded as a configuration error or operators have difficulties updating the software [62].

On the lines of Osmocom, the open-source community witnessed an LTE network side stack called `OpenLTE` [41]. Commercial products such as Amarisoft [38] also existed in the market; however, they lack source-code availability and hence, not suitable for research. `OpenLTE` is the primary testbed used in this thesis. Later in 2015, another open-source software named `SRSLTE` [149] was released, which today offers both UE and network side implementations, including eNodeB and EPC components. At the time of writing this thesis, it is the most stable implementation of LTE stack available and is utilized by various researchers and organizations across the world. Recent trends and the shift of cellular protocols towards IP-based architecture, telecommunications is not anymore a closed garden. Today, it has become relatively easy to build and operate (with a valid license), a mobile network with cheap hardware, and openly available software just with basic knowledge of programming and networking protocols.

3.1 IMSI Catchers

IMSI catcher is a device to identify and track cellular phone subscribers, traced back to the 1990s with devices such as StingRay from Harris Corporation [74] and GA 900 from Rohde Schwarz [56]. These devices detect the presence of permanent cellular identities such as IMSI and IMEI linked to the subscriber's SIM card and mobile device, respectively. As mentioned earlier, the broader availability of software-defined radio (SDR) and free, open-source software tools have reduced the cost to build such a device [135, 109], therefore potentially increasing the availability of the IMSI catcher for malicious adversaries.

IMSI catcher impersonates legitimate networks and mimics the characteristics of real base stations. As a thumb rule, they transmit with a higher power than the surrounding base station to attract the phones. They operate using distinct area identification codes deviating from any nearby cells to trigger the mobility management

procedure(s) from the phone and steal their identities. Sophisticated and highly capable IMSI catchers can also perform a MITM attack to intercept the cellular traffic. Since GSM specifications have no means for the mobile device to authenticate the network, it suffers from the fake base station problem. Further, the optional use of encryption enables interception attacks once the mobile device selects the rogue network.

Due to the mandatory use of mutual authentication in 3G networks, attacks that were possible over 2G networks are harder to achieve. Many commercially available IMSI catchers first downgrade the user's device to GSM and then perform the attack [124]. Using jammer to downgrade the device is a well-known technique. Differently, researchers demonstrated interception attacks on the 3G network using the compromised small-sized base station known as femtocells [58]. 5G networks defend such IMSI catcher devices [18]; however, downgrading attacks (to force subscribers in using 4G, 3G, and 2G networks) are still possible in 5G, as will be demonstrated in this thesis. Therefore, IMSI catcher devices may be a potential threat in the 5G era as well due to support to the legacy cellular networks. Besides, 3GPP is seriously addressing the issue of the fake base station in the upcoming releases of 5G networks [22].

3.2 LTE Experimental Testbed

We develop an LTE experimental testbed using inexpensive hardware and open-source cellular software stack(s). Our testbed consists of a fully functional and configurable UE, eNodeB, and MME components required to operate a complete LTE network. Further, we can achieve registration, security, paging, and handover procedures using our setup. We leverage this testbed to perform various wireless attacks and evaluate the effectiveness of LTE security trade-offs in the next two chapters.

3.2.1 Hardware

Figure 3.1 depicts the individual components of the testbed. The primary hardware component is a SDR, which is essentially a radio that is wholly or partially configurable by software. Precisely, "*its a radio communication system where components that are typically implemented in hardware (e.g., mixers, filters, amplifiers, modulators/demodulators, detectors. etc.) are instead implemented using a software*" - taken from Wikipedia [164]. The SDR is connected to a host computer, to be

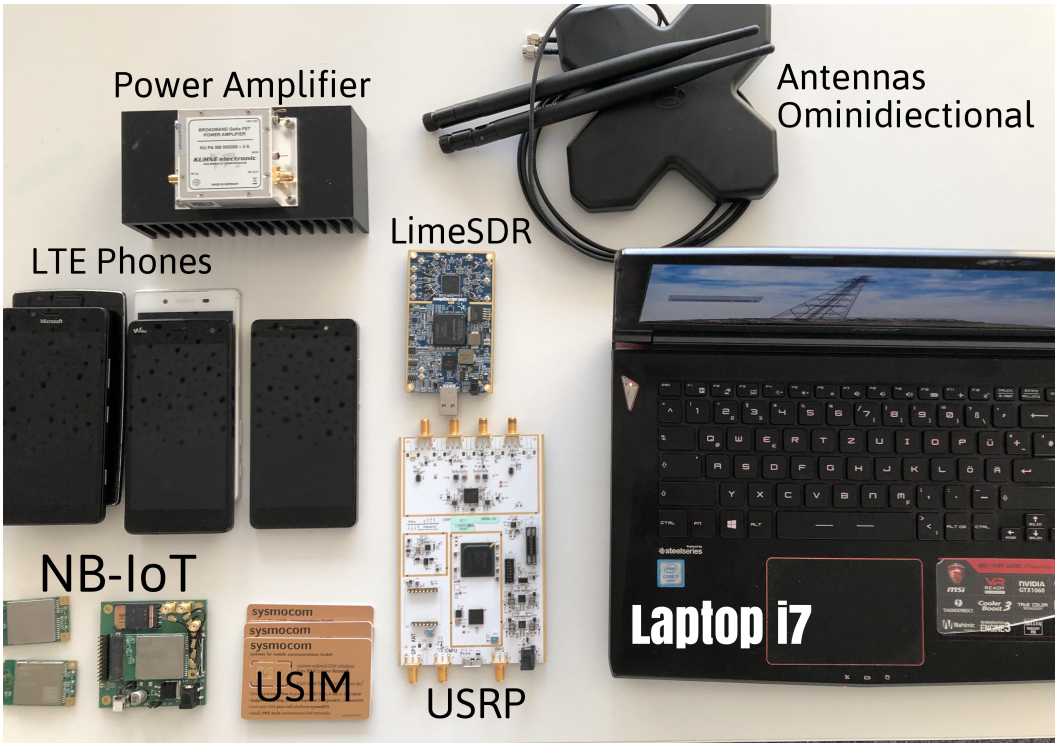

Fig. 3.1.: Low-cost Hardware Components for LTE network operation

used by host-based software to transmit/receive signals/data over the air. Popular software includes GNU Radio and contains a suite of signal processing applications. SDR offers a full-duplex or half-duplex radio module and supports a wide range of electromagnetic frequency bands between several kHz to GHz, making it suitable for numerous applications including, cellular, WLAN, etc. Moreover, it's compact and ranges from a credit card size to the size of a laptop. SDRs have become much more affordable in recent years due to innovation in hardware, and some of the widely known SDR's include USRP, Bladerf, and Hackrf. Recently, Limesdr [101] has also become a favorite choice of researchers as they are low-cost.

LTE is a full-duplex communication system, and hence we require an SDR with full-duplex features. Therefore we chose Universal Software Radio Peripheral (USRP) to operate as a radio unit in our testbed. Further, it also offers a wide range of LTE frequency bands and is widely used in the research community. It has USB3 support for stable and high-speed data transfer over the USB3 link to the host computer. Moreover, early LTE software stack developments like OpenLTE preferred USRP and the related drivers are already available. In particular, we use the `USRP B210` [52] model that was available in 2015 and had two transmitting and receiving radios, and hence, the higher price compared to its counterparts such as `USRP B200mini-i`. The mini is a trimmed version with a single transmit and receive unit costing up to 700 euros. Even though we utilized USRP B210, which costs around one thousand

euros, some of our passive attacks can also be realized practically with more cheaper radio hardware. For example, RTL-SDR [121] dongles, which cost around 15 euros, can be leveraged to listen over the LTE air-interface passively. However, RTL-SDR devices are not as stable as USRP due to hardware limitations.

Next, we require a powerful general-purpose processor to perform highly intensive signal processing using GNU Radio software. In summary, our hardware platform contains a USRP B210 device that connects to a host laptop (Intel i7 processor & Linux-based OS), serves as our radio unit to send/receive LTE signals using appropriate LTE software. The software, i.e., the LTE stack, including that of the eNodeB's RRC layer and MME's NAS layer are operates from the laptop.

We selected popular LTE-capable mobile phones available in the market. Further, we also used modems from tablets, automobiles, laptops, routers, USB data sticks that have LTE capable modems ranging from LTE release 8 to 14 for various experiments. These devices incorporate LTE implementations from five major LTE baseband vendors such as Qualcomm, Samsung, Intel, Mediatek, and Huawei, who collectively account for the vast majority of deployed LTE-capable UEs. Further, we also used NB-IoT chipsets based on LTE release 13 specifications for our experiments. A complete list of end-user devices (UEs) is presented in Table B.1 in Appendix B.

The second type of hardware testbed costing 300 dollars is also developed and utilized in our research. The essential goal of this testbed is to operate a minimal LTE network - transmitting only the broadcast channels SIB type1 and SIB type2. Precisely, we used the UDOO X86 embedded PC as host and a LimeSDR [101] as the radio unit. UDOO is based on the Intel Atom processor and connected to LimeSDR via USB 3 port. The LimeSDR is a full-duplex system costing around 150 $ and operates similar to the USRP. This testbed is used to perform various experiments in the context of self-organizing networks, and is discussed in section 4.3.1. Further, for the same experiments, we utilized dedicated SON equipment (base stations and core network elements) from a particular vendor.

The effective transmission range of USRP when operated at full power can extend up to 50 meters. Further, it heavily depends on the operating frequency and antennas used. To extend our range for certain experiments, we utilized power amplifiers [95] and directional antennas on the transmit side of USRP B210 and LimeSDR. PC/SC [166] capable smartcard reader (we used ACS ACR38 [102]) with commercial USIM cards and programmable USIM cards [153] also play a key role in our testbed.

3.2.2 Software

Cellular Protocols constitute complex state machines with various message transactions between the UE, eNodeB, and MME. In practice, this translates to millions of lines of code embedded into mobile devices and network equipment. At the time of creating the testbed, two implementations of the network side (eNodeB and MME) LTE protocol stack were available as open-source software, namely, `Openairinterface` [119] from the Eurecom Institute and `OpenLTE` from a cellular enthusiast and hobbyist. We deployed our first version of the testbed using `OpenLTE` due to its low-complexity in implementation. While each of the protocol layers is implemented in *C++*, they are well structured and documented. It was capable of performing a complete LTE registration procedure using LTE release 9 devices. Once a commercial UE registers with our test LTE network, we could send and receive data and were successful in maintaining a stable connection for more extended periods. We exclusively used this a customized version of this testbed to operate as a passive listener and a rogue eNodeB presented in section 3.3.2 and section 3.3.2 respectively.

The second version of the testbed is created in 2017 using the `SRSLTE` project. `SRSLTE` is an open-source framework comprising UE (`srsue`), eNodeB (`srsenb`) and EPC (`srsepc`) implementations. It is implemented using *C/C++* language and runs on an off-the-shelf general-purpose processor and supports various LTE bandwidths. Further, it also allows logging at various levels of the LTE stack, notably, RRC and NAS protocols. The protocol support is only available up to release 10 specifications during the research.

Further, required message modifications are programmed by us to support our experiments. We mainly used this version of the testbed to create a MITM relay, as discussed in section 3.3.2. These testbeds enabled us to conduct real-time experiments on LTE devices and networks. Although various functionalities like paging and handover were unavailable in the source code, we did program the additional required features for our test purposes. Further, we also added support for operating LTE release 10 and 11 specifications, later in which we identified vulnerabilities. Besides, we modified the telephony protocol dissector [122] available in Wireshark [165] to decode all messages exchanged between the rogue eNodeB and UE, including RRC, and MAC layers in real-time (rather than logging into files and viewing them later). These modifications are submitted to the Wireshark project and merged into the mainstream application.

3.3 LTE Threat Landscape

We present the LTE threat landscape and characterize various adversaries in it. We also describe our customizations towards open-source software to achieve adversarial capabilities and detail their operation concerning LTE networks.

3.3.1 Adversary Modelling

LTE security requirements as stated in [19] guarantee protection from various threats and attacks over the access network. We design different adversary models to challenge these requirements and evaluate the security and privacy of LTE networks. Our adversary models are constructed by studying the security analysis document [17] created by 3GPP to address shortcomings of 3G and improve the security for 4G networks. We identify specific weaknesses in the control plane protocols that did not evolve with the LTE security architecture. Additionally, our adversary models exploit new vulnerabilities we discovered in the LTE access network.

We assume that the adversary does not possess any private information of the victim, but may hold public identifiers such as the phone number (MSISDN), email address, twitter handle, etc. The goals of the adversary include a) identifying the subscriber and device (UE) uniquely, b) learning the precise location of a subscriber in a given geographical area, c) force subscribers to use less secure GSM or 3G networks thereby exposing them to various attacks such as IMSI catchers [151], d) Deny legitimate LTE services to subscribers, e) disrupt network infrastructure, for instance, launch DoS attacks inside the network. Unlike SS7 or Diameter attacks, these attacks cannot be performed remotely. Hence, the adversary should be in the same geographical area (or near surroundings) as the victim subscriber or target. Based on the learnings from security attacks over 2G and 3G networks and our understanding of LTE protocol vulnerabilities, we characterize the adversary models into three types: passive, semi-passive, and active.

A. Passive

A passive adversary can silently sniff LTE over-the-air (radio) broadcast and dedicated control signaling channels. To achieve this, the adversary has access to a hardware device (for example, USRP) and associated software needed to observe and decode the radio signaling messages.

B. Semi-Passive

A semi-passive adversary is, in addition to passive monitoring, able to trigger signaling messages to subscribers using interfaces and actions that are *legitimately available* in LTE or higher layer systems. For example, a semi-passive adversary can trigger paging messages to subscribers by sending a message via a social network or initiating a voice call. The adversary is assumed to be aware of the social identities of subscribers. For example, these identities can be a Facebook profile or a mobile phone number of the subscriber. A semi-passive adversary is analogous to the 'honest-but-curious' or 'semi-honest' adversary model used for cryptographic protocols [60].

C. Active

Common capabilities required for active adversary include knowledge of LTE specifications and hardware such as USRP. We discuss three types of active adversaries developed and utilized in our research:

1. The first type of active adversary can set up and operate a rogue eNodeB to establish malicious communication with UEs. It involves impersonating subscriber's serving operator network and injecting malicious packets to UEs. An active adversary is analogous to the 'malicious' adversary model in cryptographic protocols [60].

2. The second type of active adversary can operate a rogue UE and establish communication with the legitimate network. The adversary can inject malicious information into the network to achieve a specific purpose or impersonate a victim subscriber(s).

3. The third type of active adversary acting as a MITM can relay the traffic between a victim UE and a legitimate network. Further, the relay modifies/inject information into the unprotected LTE control plane messages. In this context, unprotected refers to no encryption and no integrity protection for the message.

3.3.2 Building Rogue LTE Components

We now model the described hardware and software components into malicious network elements that enable the adversaries to perform various attacks. We design four types of rogue elements: passive listener, `rogue eNodeB`, `rogue UE`, and `relay`.

Fig. 3.2.: `Passive listener` and `Rogue eNodeB`

A. Passive Listener

To sniff LTE broadcast channels, we utilized parts of `SRSLTE`. It is a free library for software-defined radio mobile terminals and base stations. Currently, the project has a UE-side LTE baseband implementation. `SRSLTE` uses the Universal Hardware Device (UHD) library to communicate with the USRP B210 as shown in Figure 3.2. Since all the passive sniffing happens in real-time, it is recommended to have a high-speed host (laptop) to handle the high (30.72 MHz) sampling rates without data loss and also to maintain constant sync with eNodeBs. To build our `passive listener`, we used the `pdsch-ue` application to scan a specified frequency and detect surrounding eNodeBs. It can listen and decode SIB messages broadcast by surrounding eNodeB. Further, we modified `pdsch-ue` to decode paging messages which are identified over-the-air with a Paging-Radio Network Temporary Identifier (P-RNTI). Upon its detection, GUTI(s) and IMSI(s) are extracted out of paging messages.

In a semi-passive attack mode the adversary leverages Facebook [54, 53] and WhatsApp [162] applications over the Internet to initiate communication with the targets.

B. Rogue eNodeB

An active attacker can always succeed in masquerading as a real operator due to the following two reasons: firstly as a thumb rule, UE always prefers to connect to the eNodeB with the strongest signal according to [9] and secondly, UE trusts all the *SIB messages* broadcast by a eNodeB. Hence, it is relatively simple to mount a `rogue eNodeB` shown inFigure 3.2 and attract UEs by transmitting at high power and spoofing the target operator's *SIB messages*. This attack cannot be operated remotely and in certain cases has the risk of detection by the operator. However, a sophisticated attacker can orchestrate the attacks in different ways to avoid exposure.

We built an eNodeB to mount successful active attacks against UEs registered with a real LTE network. The process of building such `rogue eNodeB` is as follows. Generally, UE always scans for eNodeBs around it and prefers to connect to the eNodeB with the best signal power. Hence in the IMSI catcher [151] type of attacks, `rogue eNodeB` is operated with higher radio power than surrounding eNodeBs. However, in LTE, the functionality of the UE may be different in some situations. In particular, when a UE is very close to a serving eNodeB, it does not scan surrounding eNodeBs. It allows UEs to save power. Hence to overcome this situation in our active attacks, we exploit another feature named *'absolute priority-based cell reselection'* and introduced in the LTE release 8 specification [30].

The principle of priority-based reselection is that UEs, in the IDLE state, should periodically monitor and try to connect to eNodeBs operated with high priority frequencies [30]. Hence even if the UE is close to a real eNodeB, operating the `rogue eNodeB` on a frequency that has the highest reselection priority would force UEs to attach to it. These priorities are defined in SIB Type numbers 4, 5, 6, and 7 messages broadcast by the real eNodeB [8]. Using passive attack setup, we sniff these priorities and configure our eNodeB accordingly. Further, the `rogue eNodeB` broadcasts MCC and MNC numbers identical to the network operator of targeted subscribers to impersonate the real network operator. Generally, when UE detects a new TA, it initiates a *"TAU Request*" to the eNodeB.

To trigger such request messages, the `rogue eNodeB` operates on a TAC that is different from the real eNodeB. We discover the highest priority frequency using modified *cell_search* application from `SRSLTE`. It resides on the PC and controls the `USRP B210` (or `LimeSDR`) to sniff LTE broadcast information passively. By default, the application can only detect the PCI of the strongest cell on a given EARFCN. We have modified the application to detect the PCI from all available cells from all operators present in a certain area and further decode their respective *SIB type 1*

information. Further, we programmed `LTE_Fdd_enodeb` to include LTE RRC and NAS protocol messages to demonstrate active attacks.

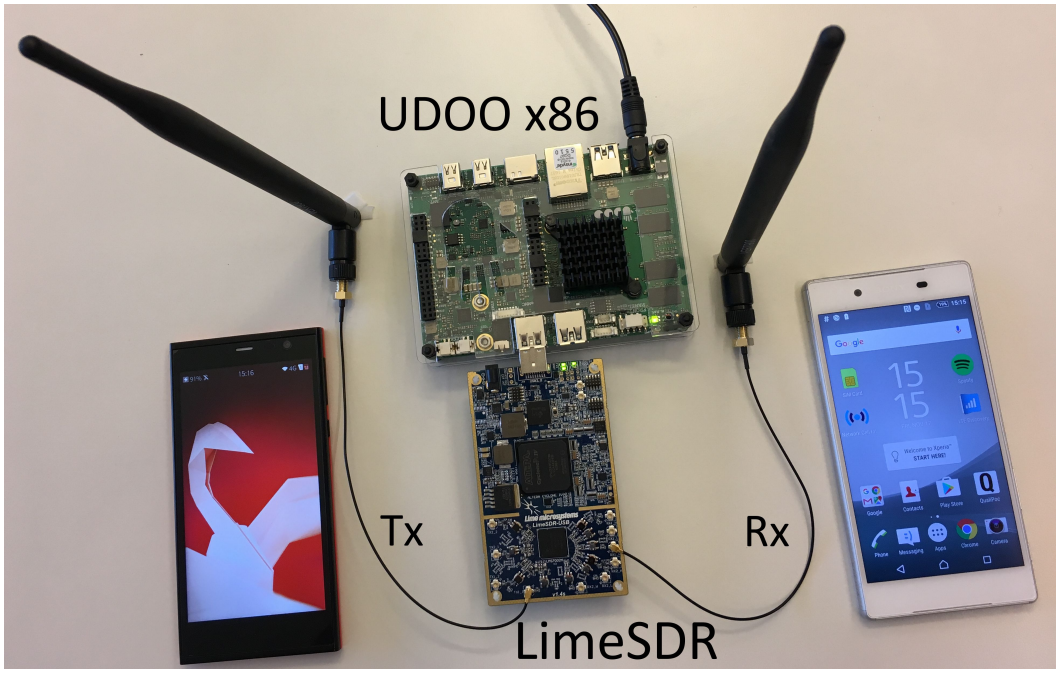

Fig. 3.3.: `Rogue eNodeB` (minimal)

For a minimal active mode, we use *pdsch_enodeb* application from `SRSLTE` which uses `LimeSDR` to operate a `rogue eNodeB` as shown in Figure 3.3. Nevertheless, the operation is similar to eNodeB built is using USRP. For verification and evaluation purposes, we also monitor the signaling messages directly on the UE. To achieve this, we also built a custom tool called *cell_logger*, which runs on the host PC to acquire the information directly from the UE's baseband processor and are referred to as diagnostic messages. Further, we decode the RRC and NAS messages exchanged with eNodeB and MME, respectively, using Wireshark.

C. Rogue UE

To create a `rogue UE` we used the software *srsue* from the `SRSLTE` suite together with USRP B210 as shown in Figure 3.4. `Rogue UE` searches for a specified frequency and initiates communication with the legitimate network, i.e., eNodeB (also with the MME). We leverage `rogue UE` for two purposes. First, to impersonate a target USIM by using its IMSI. Second, inject false RRC *measurement report* information into a legitimate network to poison the operator's database. We do not require any security procedures to inject such messages to the network and hence remain anonymous

Fig. 3.4.: Rogue UE, with programmable USIM cards and smart car reader

during this procedure. Nevertheless it is also possible to impersonate a legitimate subscriber by using the assigned GUTI by the network.

D. Relay

A relay acts a MITM and consists of a rogue UE and a rogue NodeB. The configuration of the rogue eNodeB is similar to the eNodeB discussed above. Further, it is directly connected to the rogue UE (on a different host) that relays the traffic between the victim UE and the legitimate network. We followed a similar approach, like in [138], to maintain a stable connection between legitimate UE and the network. However, we used a frequency number for the operation of rouge eNodeB different from the legitimate operator and hence avoiding our rogue UE connecting to our rogue eNodeB. For the setup in Figure 3.5, we use the modified srsenb (like above) and a modified srsue to receive and relay the control plane messages (RRC and NAS) between the legitimate network and victim UE. Our major modifications involve the integration of srsue and srsenb segments. Moreover, we used directional antennas and power amplifiers to improve the signal conditions between rogue UE and legitimate network. Similar to this relay setup, we have a UE segment and eNodeB segment in our NB-IoT testbed and also refer to them as a relay in our experiments.

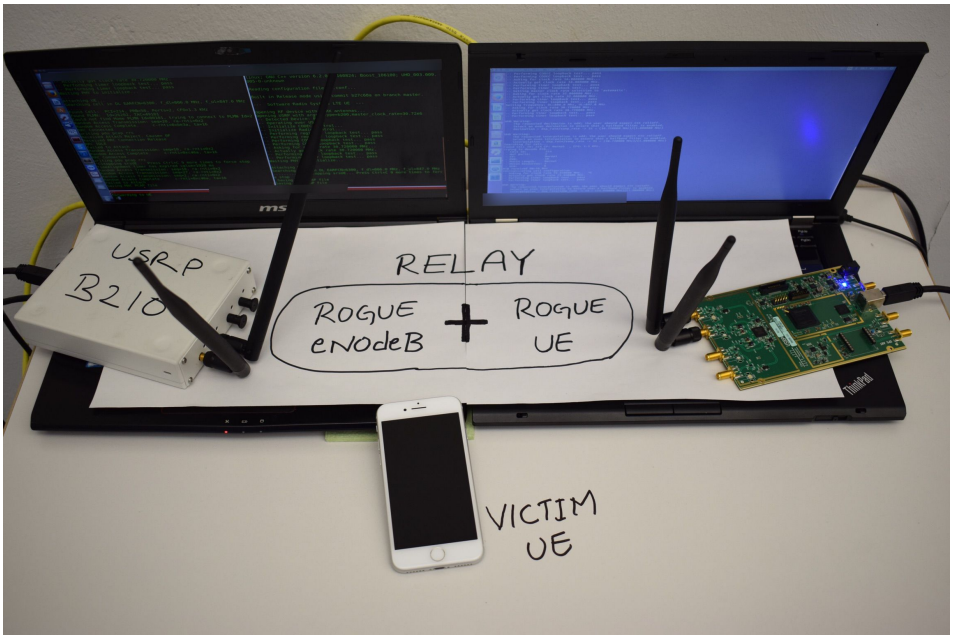

Fig. 3.5.: Relay

Note: We performed all the experiments using our test phones, and extreme care is taken not to interfere with nearby communications. Further, we have legitimate permissions from an operator to transmit in one of their commercial LTE frequencies. We carried out most of the active attacks in a Faraday cage [57] to avoid affecting other UEs. For attacks in real LTE networks, we took care not to interrupt standard service to other UEs in the testing zone. Initially, we determined the GUTIs of our test UEs via passive attacks and fed them into our `rogue eNodeB`. We programmed our `rogue eNodeB` to accept *"TAU / Attach / Service Requests*" only from these specified GUTIs and to reject all requests from unknown UEs with the EMM reject cause number 12 *"Tracking area not allowed*" [16]. Upon receipt of this message, all UEs other than our test UEs disconnect automatically from our `rogue eNodeB`.

Attack amplification: Related to our passive attacks, we determined the average cell radius of a major operator in a city is 800 meters for the 2.6 GHz and 1 km for the 800 MHz frequency band. The USRP B210 used for our attacks has a maximum output power of 20dbm (100mW) [106] with a coverage range of 50 to 100 meters. However, the signal coverage area can be increased with a suitable power amplifier as shown in our hardware setup. Specifically, based on the COST 231 radio propagation model [146], we calculated that by mounting a USRP at the height of 10m (e.g., on a street lamp) and amplifying the power by 10 dB, it is possible to operate a `rogue eNodeB` for every subscriber in a cell. For a reference, `OpenBTS` projects [89, 159] use USRPs to provide GSM coverage in rural areas with >2 km coverage with an

external power amplifier and antenna. Similarly, the signal coverage area of our `rogue eNodeB` could be increased to demonstrate the feasibility of the attack.

Attack Stealthiness: The end-subscribers cannot detect the presence of the operation of the rogue components. Unfortunately, the mobile OS (e.g., Android or iOS) cannot detect them. The reason is that the protocols are either executed in USIM and the baseband chip that communicates limited information to the mobile OS. Although there are special applications that can offer such detection, [42] proves that they can be bypassed. On the other hand, carrier networks can be able to detect the rogue components unless they have specialized equipment such as [68]. Nevertheless, the attacker can implement smart techniques to bypass them, as shown in our attacks.

Attacks that directly interact with the network components using `rogue UE` also remain stealthy during our experiments, since all the operator networks we tested do not possess any detection features. Although LTE networks have support for intelligence, they are merely used for traffic distribution, load management, and etc. to achieve optimized network performance.

3.3.3 Ethical Considerations

We strictly understand the legal terms while performing our experiments due to their active transmission over licensed wireless channels. We took precautions not to disrupt communications of unknown and unrelated UEs or networks. We leverage a Faraday cage wherever possible and performed experiments under the careful guidance of a operator. Next, our work reveals vulnerabilities in LTE specifications, which are already in use in every LTE-enabled UE worldwide. Further, we also encountered several implementation issues in popular smartphones and LTE network configuration issues. Therefore we made an effort to responsibly disclose our work to the relevant standard bodies and affected parties. Our reports are acknowledged by all vendors and network operators we contacted. For those vendors who have a standard responsible disclosure process in place, we followed the process. For instance, we approached GSMA through their coordinated vulnerability disclosure program and reported our attacks. Further, GSMA has taken the responsibility to distribute our research among operators worldwide and also supported in pushing mitigations on the operational networks. Our work is listed in various vulnerability disclosure programs [85, 84, 125, 65] from GSMA, Qualcomm, Huawei, and 3GPP.

4 Trading Security for Availability

Ensuring the availability of communication services anytime, anywhere, is vital in a connected society. Today an essential objective of mobile network operators is to provide seamless coverage and connectivity to their customers. Operators manage and optimize the radio network through various techniques such as link adaption, automatic repeat request, admission control, handover, load balancing, etc. Further, LTE introduces enhancements to existing radio management protocols to identify and troubleshoot radio problems. In particular, release 10 networks are fitted with self-organizing capabilities to automatically configure and optimize the network performance during its operational state.

A radio network is made available to the UE by the broadcast messages transmitted by eNodeBs, precisely, *System Information Block* messages. Conventionally, this information is available in clear-text to all the mobile devices. However, there have been significant efforts from the standards to introduce Public Key Infrastructure (PKI) architecture into the system to prevent a mobile device from connecting to rogue networks. But due to the infeasibility of the solution [28], 4G networks did not adopt this architecture and, hence, SIB messages are broadcast in clear-text.

To maintain the mobility of UEs on a radio level, the network (eNodeB) requires continuous measurements from the UE. As illustrated in Figure 2.5, UE periodically reports its measurements to eNodeB over the RRC control plane, e.g., to facilitate handover. In case of handover or connection failures, possibly due to interference or coverage holes, the network would want to know the cause for such a failure. Thus, UE instantly logs the failure issue in its memory and later reports to the eNodeB (OAM) when requested. To precisely geolocate the issue, UE may include its GPS position in the report.

Traditionally, operators followed the drive testing approach for network optimization. However, they have been expensive and time-consuming. LTE introduced a cost-efficient, and a standardized mechanism called the Minimization of Drive Test (MDT) [33]. In MDT, the network programs the commercial UEs (with subscriber consent) to perform measurement logging of preferred eNodeBs and networks. Later, when requested, UE transfers this information to the network and is used to identify weak

coverage areas. Such measurement information is also used by SONs to identify problems and fix them instantly.

Although the traditional way of gathering information from UEs remained similar across all network generations, the amount of information received from UE has significantly increased in LTE. From a security point of view, UEs logged measurements for several hours can be translated to the location trace of the UE/subscriber, and exposure of such information is a privacy threat. Given, the use of encryption is optional over the air [23], location information can be leaked to passive adversaries if the operator has disabled RRC encryption (by using the Null encryption algorithm).

Networks place implicit trust in the information received from mobile devices. Note that these devices operate in vulnerable locations and can easily be compromised. It is necessary to investigate if operators have sufficient measures to filter out false information received from compromised devices. Similarly, UEs trust the information broadcast by the eNodeB and lack verification mechanisms, both standardized and vendor-specific. It lures them to connect to fake base stations, and the subscriber is unaware of this switch. Given the LTE system design, with greater emphasis on the availability of the network to UEs, we investigate the security and resilience aspects by applying our threat models to radio management protocols.

4.1 Security Weaknesses in LTE Radio Network Management

The LTE radio network management includes various RRC functions needed to set up over-the-air connectivity, optimize connections and troubleshoot problems between eNodeB and UE as described in [12, 6]. We describe these functions in detail and highlight their security weaknesses that enable **location leak attacks in section 4.2** and **DoS attacks in section 4.3**. Further, we discuss the rationale behind these weaknesses and propose mitigations that can be (and are) applied to the 4G and 5G RRC specifications. The RRC functions can be classified as namely broadcast information, measurement reporting, and SON features.

4.1.1 Broadcast Information

In this RRC protocol function, the eNodeB periodically broadcasts SIB messages which carry information for UEs to access the network, perform cell selection, and

other information as described in [12]. Next, temporary identities associated with UEs (i.e., GUTIs) are transmitted over the air as paging messages in a broadcast channel. Such broadcast messages are neither authenticated nor encrypted. Hence an active or passive adversary can decode them with appropriate equipment. Since these broadcast messages are only sent in specific geographical areas, we can use the method described in [96] to reveal the presence of subscribers in a targeted area by exploiting these broadcast messages. As detailed in section 3.3.2, the attacker can utilize this broadcast information to configure the `rogue eNodeB` and perform malicious activities.

4.1.2 Measurement Reporting

In LTE, UE performs network measurements and sends them to the eNodeB in RRC protocol messages when requested. Such UE measurement reports are necessary for network operators to perform handovers and also troubleshoot signal coverage issues. In particular, there are two types of UE measurement reports - one sent in *"Measurement Report*" used as part of handover procedure and another one in Radio Link Failure (RLF) report - which is used to troubleshoot signaling coverage. However, since these messages are not protected during the RRC protocol communication, an attacker can obtain these network measurements by only decoding from radio signals.

We now explain the importance of two RRC protocol messages and measurement information they carry. First, *"Measurement Report*" message is a necessary element during a handover procedure in LTE networks. Generally, eNodeB sends an RRC message indicating what kind of information is to be measured, and in response, the UE sends *"Measurement Report*" messages. We discovered that the LTE specification allows sending this message to the UE without the AS security context [12]. Second, the RLF report is a feature to detect connection failures caused by intra-LTE mobility and inter-system handovers between LTE, GSM, and 3G networks. Upon detection of such events, RLF reports are created by the UE and forwarded to eNodeB when requested. These reports are collected by the OAM system for troubleshooting.

As per the LTE standard specification [12] Appendix A.6, the *"UEInformationResponse*" message carrying RLF report should not be sent by the UE before the activation of the AS security context. However, we discovered that major LTE baseband vendors failed to implement security protection for messages carrying RLF reports. This suggests that the specification is ambiguous, leading to incorrect interpretation by multiple baseband vendors. In particular, *"Measurement Report*" and *"UEInformation-Response*" messages contain serving and neighboring LTE cell identifiers with their

corresponding power measurements and also similar information of GSM and 3G cells. Additionally, if supported, the message can include the GPS location of the UE (and hence of the subscriber). We exploit the above vulnerabilities to obtain power measurements, which we then use to calculate a subscriber's precise location.

4.1.3 SON Features

SON comprises several sub-functions to enable the automatic operation, maintenance, and optimization of the network. As the 3GPP standardized only the high-level SON functions, vendors are free to implement the low-level features using their proprietary algorithms. Hence, we gathered SON configuration details directly from SON solution providers and engineers of two major European mobile network operators. We study and reveal the information of different algorithms in this section. Also, some of these algorithm's details are published in various Internet resources and 3GPP documents [6]. By analyzing this information, we discovered security weaknesses in the three most essential SON functions that lead to DoS attacks presented in section 4.3.

Automatic Neighbour Relation (ANR) Management. It allows the automatic discovery of eNodeBs and the establishment of relations between them. For this, ANR depends on the *measurement reports* provided by the UE. If any of the cell(s) or PCI(s) reported via *measurement reports* is not in its neighbor list, with the assistance of UE, the eNodeB acquires the ECGI of the unknown cell. Further, to create a relation or connection to the unknown cell, the eNodeB acquires the IP address of the unknown eNodeB and establishes an X2 interface with it. During this setup, both eNodeBs share their serving cell list, operating frequency bands, neighbor lists, etc [6]. Hereafter, they can perform X2 based handovers.

ANR function residing in each eNodeB maintains a table called neighbor relation table to manage connections with neighboring cells [6]. A neighbor relation table contains three attributes for each cell present in it: NoRemove, NoHO, and NoX2 [6]. If the attributes are set to true for NoRemove - eNodeB cannot remove the neighbor cell from the NR table, NoHO - handover cannot be initiated to this neighbor cell, NoX2 - cannot set up an X2 interface to this neighbor cell. During optimizations, the SON engine can toggle these attributes to control connections between neighbor cells. However, both 3GPP standard [6] and implementations [78] allow X2 relation setup without any authorization. It implies that the X2 relationship can be set up between any two eNodeBs provided that the PCI and ECGI being used are valid and known to the OAM. Creating X2 relations without any control generates excess signaling load over the X2 interface and exhaust related network resources.

Automatic PCI Configuration. When two neighboring eNodeBs are operating on the same EARFCN and PCI, it causes conflict and can be of two types: PCI collision and PCI confusion. PCI collision happens when two intra-frequency cells are using the same PCI. As a result, UEs in the overlapping region of the two cells will lose synchronization with the eNodeB and fail to decode signals correctly. PCI confusion occurs when the adjacent cells of a cell happen to use the same PCI, leads to incorrect handovers.

A self-configuring functionality can avoid conflicts by enabling the eNodeB to perform an automatic selection of a PCI during power-up based on the information available from the OAM and its neighbors. However, during the operational phase, if conflicts are detected, they are resolved through a PCI optimization procedure. There are several ways an eNodeB to detect conflicts: via X2 messages from neighboring eNodeBs, ANR reports [6], and dedicated network scanners (such as *network listen* [127]). One of the implementations indicated that an alarm is raised in the OAM when an eNodeB detects a PCI collision [126].

Following the detection, the OAM decides to schedule (possibly during low traffic times) a PCI optimization procedure where a different/new PCI is assigned to the affected eNodeB, and simultaneously the neighboring eNodeBs are updated about this change. Some implementations allow eNodeBs to instantly and automatically perform optimization and change their PCI [126]. When the PCI optimization feature is not activated, the operator will likely notice disturbances in the network due to handover failures, and this requires manual intervention to identify the root cause and perform the optimization. In either way, the operator is alarmed about the conflict, and the eNodeB needs restarts to acquire a new PCI. Thus, a reboot can be triggered by merely broadcasting PCI over the air interface. Such an unreliable control strategy adopted by SON implementations allows an adversary to control the operation of an eNodeB.

Mobility Robustness Optimization (MRO) It is a self-optimizing technique to manage failures caused by incorrect handover settings. In particular, it monitors the handover statistics, identifies abnormal handover scenarios, and optimizes the handover-related parameters. The goal is to identify and avoid unsuitable eNodeBs so that failures such as RLF can be eliminated. A discussed UE can experience RLF due to a handover that happens too early or too late or to a wrong cell [6]. Consequently, the UE creates an RLF report consisting of the current cell (where the RLF occurred). If available, similar information of 2G and 3G cells are also included in the report.

To instantly recover from a handover failure, the 3GPP specification [6] allows the UE to re-establish the lost connection, either to the source eNodeB or target eNodeB. Accordingly, UE sends a *RRC connection re-establishment request* message to a suitable eNodeB and receives the required radio resources to resume the lost connection. Following this, UE transfers the RLF report to the eNodeB. In case when the eNodeB does not have enough information to resume the connection, it responds with a *RRC connection re-establishment reject* message, which causes the UE to switch from *RRC CONNECTED* state to *RRC IDLE* state and suspend any call/data procedures. Next, UE performs a new cell search procedure [12] and links to the best available eNodeB.

In parallel, the SON engine continuously monitors the handover performance of the eNodeB through various KPIs. For instance, the handover execution success rate is measured as the ratio of handover execution success number to the handover execution attempt number [81]. The resulting success rates and the RLF reports acquired from a UE are used as a metric by the MRO function to analyze handover performance and generate a KPI. When the KPI is declining, the SON engine initiates optimizations to solve handover-related problems. For instance, in certain vendor implementations [78, 79], eNodeBs exhibiting frequent handover failures are treated as abnormal and removed from the neighbor list so that handovers can be prohibited to them.

Another vendor implementation suggests tuning the Cell Individual Offset (CIO) [37, 79]. A CIO decides the point at which a *measurement report* can be sent to the eNodeB, thereby delaying or advancing the handover decision. One of the experts implied that in some cases, the eNodeB with poor performance is disconnected from the live network and will be compensated either by using surrounding eNodeBs or 3G/2G networks.

Although *measurement reports* and RLF reports are securely transmitted to the eNodeB [8], the information contained in them is not verified by the network. Hence, a compromised UE can deliberately inject false information into these reports that are later used by the SON engine to perform optimizations which result in poor network performance. In *RRC CONNECTED* state eNodeB receives *measurement reports* from the UE to track its mobility. Precisely, these reports contain network information that is used by the eNodeB to make handover decisions. Note that, *measurement reports* are received over an encrypted channel that is set up after a successful authentication procedure [8]. However, this important network information is not verified by the network, in particular by the eNodeB, before making any handover decisions. This

indicates that by operating a `rogue eNodeB`, an adversary can exploit the handover procedure and inject false network information into the *measurement reports*.

In a radio, environment eNodeBs are detected and identified purely based on their unencrypted broadcast information, i.e., SIBs, and further, there is implicit trust in such information. Similarly, eNodeBs have implicit faith in the radio information received from the UEs and do not verify the authenticity of its content, such as measurement reports, etc. In summary, it is evident from these weaknesses that SONs operate based on numerous parameters collected from LTE network operation such as *measurement reports*, RLF reports, and PCI. These parameters are leveraged to perform network optimizations. Significantly, SON lacks a mechanism to verify the authenticity of these parameters and entirely trusts the LTE security mechanisms for the correctness of these parameters.

4.2 Compromising LTE Subscriber Privacy: Location Leaks

In this section, we show how the approximate location of an LTE subscriber inside an urban area can be inferred by applying a set of new passive, semi-passive, and active attacks. In particular, we track down the location of a subscriber to a cell level (e.g., 2 km^2 area) using passive attacks (L1[1]) and further determine the precise position using active attacks (L3). We first describe the background for the attacks by summarizing the features and aspects of LTE that are used by the attacker. We then characterize preliminary measurements used for realizing the attacks and new techniques for triggering subscriber paging. Finally, we explain the attacks in detail.

4.2.1 Location leak enablers

We identified a set of features/implementations, especially in LTE networks and today's widely used data applications that enable location tracking of subscribers to an exact level compared to what is existing in GSM networks. We refer to them as network configuration issues, subscriber identity mapping techniques, and observations about specific LTE network access protocols. We will later make use of all of these aspects in developing our location leak attacks.

[1]For the sake of simplicity, we refer location leak attacks as L1, L2, and L3

A. Network configuration issues

In LTE, network operators deploy various methods to minimize signaling overhead introduced due to the evolution of networks, devices, and smartphone applications [115]. Two such deployment techniques relevant to our attacks are:

Smart Paging. In GSM, paging messages are sent to an entire location area. Thus it only allows the attacker to locate a subscriber within a large (e.g., 100 km^2) area [96]. However, LTE paging is directed onto a small cell rather than to a large TA. Such Smart Paging allows an attacker to locate an LTE subscriber within a much smaller (e.g., 2 km^2) area, which is a typical LTE cell size as observed in our experiments in a major city.

GUTI persistence. Generally, a fresh GUTI is allocated in the following situations: (a) when MME is changed due to handover or load balancing, b) during TAU or $Attach$ procedure, and c) when network issues NAS *"GUTI reallocation command*". However, network operators tend to not always change GUTI during the above procedures [150][2]. Thus, it allows a passive attacker to track UEs based on their GUTIs.

B. Social identity to subscriber mapping

In previous work, phone calls (originating from a landline phone) [96] and silent Short Message Service (SMS) [113] techniques were used for paging GSM subscribers thereby mapping TMSIs to their phone numbers. However, these methods are not as effective anymore due to the availability of tools to detect such attacks [158, 148]. We now discuss some features in social network messaging applications that can be used to trigger LTE paging requests to devices in which the subscriber has installed the corresponding social network applications.

Facebook 'Other' message folder: Many Facebook [54] users do not know about the *'Other'* message folder (as shown in Figure 4.1) in Facebook. Usually, when a message is received from a Facebook friend, it will be stored in the standard inbox folder of that user. But messages from people who are not in the friend list may be directed to the *'Other'* folder. Further, the user is *not notified* about messages in the *'Other'* folder. The user himself has to manually check *'Other'* folder to even notice that there are waiting for messages. According to Facebook [47], this is intended to protect users against spam. When an LTE subscriber has the Facebook application

[2]The reason for not changing GUTIs often is to avoid signaling storms in LTE network as described in [150].

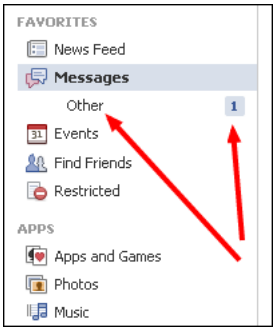

Fig. 4.1.: 'Other' folder in Facebook

installed on his LTE device, *all* incoming Facebook messages, including those that end up in the 'Other' folder, trigger a paging request by the network. Other Facebook features, such as repeated friend requests or poking (depending on the user's profile settings), also trigger paging requests. However, in those cases, unlike in the case of messages that end up in the 'Other' folder, the Facebook application notifies the user.

WhatsApp 'typing notification': WhatsApp supports a 'typing notification' feature - when someone ('sender') starts composing a message to a person ('recipient') using WhatsApp, the WhatsApp client UI at the recipient shows a notification to the recipient that an incoming message is being typed. If the recipient is using a WhatsApp client on an LTE device, this ends up triggering a paging request.

C. Initial measurements

We performed a measurement study on LTE networks of three major operators to understand GUTI allocations, Smart Paging, and mapping of tracking area and cell dimensions to examine the feasibility aspects of location leak attacks. Before measuring GUTI allocations and Smart Paging, we consider the following timing constraints for the paging procedure in LTE. Paging messages are sent only if a UE is in IDLE state. During an active connection, there are no paging messages. According to to [7], if the UE remains silent for 10 seconds during a connection, the eNodeB releases the associated radio resources and the UE moves into the IDLE state.

GUTI variation: GUTI reallocation depends entirely on operator configuration. We investigated GUTI allocation and reallocation methods used by several operators. Specifically, these experiments verify whether GUTIs are temporary in practice. We used a Samsung B3740 LTE USB data stick as the UE since it allows us to view the RRC and NAS messages in Wireshark [132]. The changes in GUTI can be seen in the *"Attach Accept*" or *"TAU Accept*" NAS messages in the Wireshark traces. We

identified these NAS messages and recorded GUTIs for every operator for further analysis. Besides, GUTI variation can be verified with engineering mode on a few selected handsets, for example, LG G3 [167]. Our results in Table 4.1 show that GUTI allocation and reallocation mechanisms are similar among all operators. The results are summarized below:

- Periodically (once an hour and once in 12 hours) detaching and attaching the UE while it was stationary resulted in the same GUTI being re-allocated in all three operator networks. A UE did not have its GUTI changed for up to three days when it is stationary or when moving between TAs in the city.

- When UE was moving inside the city for 3 days while remaining attached to the network, no change in GUTI was observed in any operator's network.

- If a UE was completely turned off for one day, a new GUTI was allocated when it was subsequently turned on. In the case of one of the operators, the newly assigned GUTI differed from the old one by only one hexadecimal digit. This implies that GUTIs were not chosen randomly.

Based on the above observations, we conclude that the GUTI tends to remain the same even if a UE is moving within a city for up to three days. Hence, temporary identities are not really temporary in any of the three networks. This allows an attacker to perform passive attacks.

Activity	***Smart Paging***		***GUTI changed?***
	on Cell	***on TA***	***(All operators)***
Facebook Message	*Yes*	*No*	*No*
SMS	*Yes*	*No*	*No*
VoLTE call	*No*	*Yes*	*No*
Attach and Detach every 1 hour	-	-	*No*
Attach and Detach every 12 hour	-	-	*No*
Normal TAU procedure	-	-	*No*
Periodic TAU procedure	-	-	*No*

Tab. 4.1.: GUTI variations and Smart Paging behavior

Smart Paging. We identified multiple cells in a busy TA for each operator and placed our passive LTE air-interface sniffer within each cell. The test UE was placed in one of the cells and remained stationary for the experiment duration. Table 4.1

presents the set of activities performed to trigger paging messages. The results are summarized as follows:

- Paging for Voice over LTE (VoLTE)[3]) call occurs on the entire TA, and paging for other IP applications occurs on the last seen cell. This is referred to as application-aware paging [116]. Since VoLTE has higher priority and strict timing constraints compared to other data applications, the network pages the complete TA to find the UE quickly.

- When the UE paging is triggered via Facebook or SMS messages, sniffers detected a particular paging message only in the cell where the UE is located (or last seen). This implies that all operators are using Smart Paging.

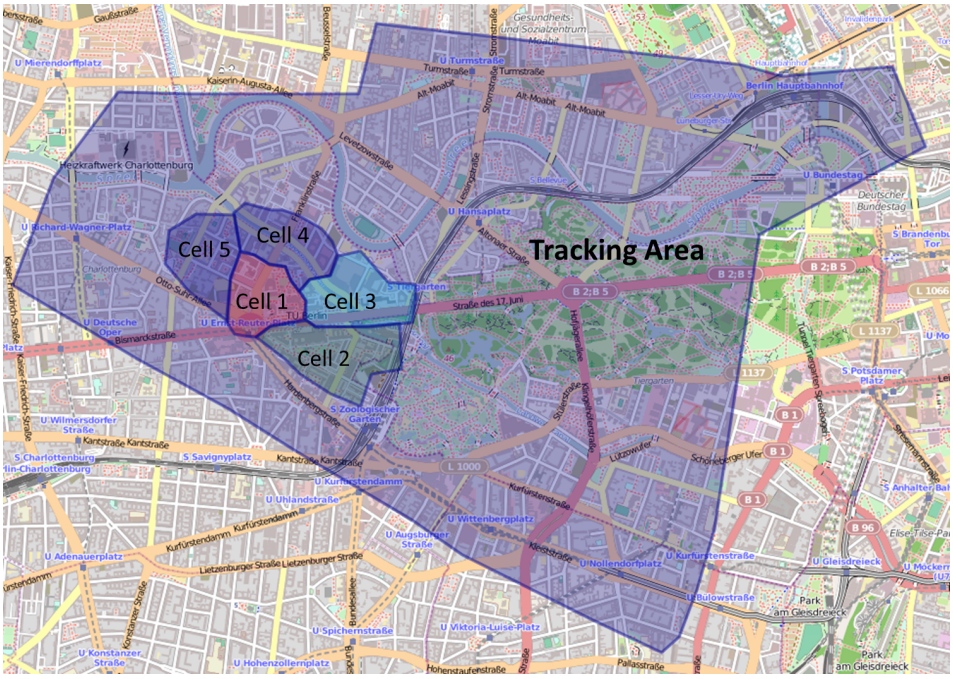

Fig. 4.2.: LTE tracking area and cells of a major operator in a city

Mapping tracking area and cell dimensions. It is necessary to know the size of LTE tracking areas and cells deployed in a metropolitan city for determining a victim's location. In particular, this knowledge enables an attacker to identify targeted TAs for specific regions and network operators in the city. We created a database that maps TACs to GPS coordinates by slowly bicycling through the city. The TACs have periodically broadcast in SIB Type number 1 messages [12]. We logged them using our passive attack setup. Further, to determine the surface area

[3]VoLTE stands for voice over LTE, and it is for voice calls over an LTE network, rather than the 2G or 3G connections which are usually used.

covered by a tracking area, we calculated the region covered by the points with the same TAC, and the results are plotted in Figure 4.2[4]. The size of the TA inside the city varies from 10 to 30 km^2. According to OpenCellID [49] tracking areas outside the city center cover 80 - 100 km^2. The TAs are smaller in size compared to the GSM location areas plotted by [59] in the same city.

Since the granularity we obtain through our attacks is on a cell level, it is important to know cell sizes in the LTE network as compared to GSM. Further, this knowledge helps in positioning the `rogue eNodeB` to maximize the effect of active attacks. To plot cell boundaries, we used the `cellmapper` [44] Android application, which reports the cell ID, eNodeB ID, and Radio Signal Strength Indicator (RSSI) of the cell in real-time. Initially, we identified a point with high signal strength (possibly close to the eNodeB) and marked it for the reference. Then we walked in all directions from the reference point till reaching the cell edge. Cell edges are identified when RSSI becomes very poor, and the UE triggers a cell change. In this way, we traced the boundaries of the 5 cells and marked them inside the TA, as shown in Figure 4.2. Based on the cell sizes measured, we find out that a major operator implemented microcells in their LTE infrastructure. Typical size of a microcell ranges from 200 - 2000 m in radius [91].

4.2.2 Passive attack - link subscriber locations/movements (L1)

In passive attack mode, the attacker's objective is to collect a set of IMSIs and GUTIs which can be used for two purposes. First, to verify the subscriber's presence in a particular area, and second, to reveal past and future movements in that area. To achieve this, we sniff over the LTE air interface using our `passive listener` (section 3.3.2) and decode broadcast paging channels to extract IMSIs and GUTIs. These identities can be collected in locations such as airports or subscriber's home or office. The attacker needs to map IMSI or GUTI associated with a particular subscriber to reveal his/her presence in that area. Since GUTI is persistent for several days in our experiments (see Section 4.2.1), its disclosure makes the subscriber's movements linkable. The mapping between GUTI and IMSI is possible using semi-passive attacks.

[4]Underlying Map source - OpenStreetMap

4.2.3 Semi-Passive attack - leak coarse location (L2)

The objective of the semi-passive attack is to determine the presence of a subscriber in a TA. Further, find the cell in which the subscriber is physically located. In particular, we demonstrate the use of novel tracking techniques to determine the TA initially and then exploit Smart Paging to identify a cell within that TA.

Determining tracking area and cell ID

We use the following two methods to generate signaling messages for performing the attack.

Using VoLTE calls. We placed 10 VoLTE calls to the victim. The VoLTE call connection times are short at around 3 seconds, according to the previous work [144]. Hence, the attacker has to choose the call duration so that it is long enough for a paging request to broadcast by the eNodeB but short enough to not trigger any notification on the UE's application user interface. As explained earlier, VoLTE has high priority, and therefore its paging requests are broadcast to all eNodeBs in a TA. Hence it is sufficient to monitor any single cell within the TA for paging messages. The observed GUTIs undergo a set intersection analysis where we apply the method proposed by Kune et.al [96] to reveal the mapping between the GUTI and phone number of the subscriber. Once successful, the presence of the subscriber is confirmed in that TA.

Using social network and applications. Social identities are a compelling attack vector because mobile subscribers nowadays use mobile phones for accessing popular social networks and instant messaging applications. The primary intention of the attacker is to trigger paging requests via social identities without LTE subscribers being aware of it. For triggering paging messages, various mobile applications can be used. Due to the popularity and size of the user base, we chose Facebook and WhatsApp applications for our experiments. However, tracking subscribers using social applications is not as effective as using VoLTE calls

We used Facebook messages as described in Section 4.2.1 to trigger Smart Paging to localize the target subscriber to a specific cell. Similar to VoLTE calls, we send 10-20 messages to the subscriber via Facebook and do the set intersection analysis to link GUTIs to Facebook profiles. If the mapping is successful in a particular cell where the attacker is, the presence of the subscriber is confirmed. Otherwise, the attacker needs to move to other cells and repeat the same procedure. The attacker can

also place `passive listeners` in every cell to speed up the localization procedure. However, this is expensive. The subscriber's presence is successfully determined in a cell that is typical of size 2 km^2, i.e., much smaller than a GSM cell.

We also used WhatsApp similarly to exploit its "typing notification" feature. In this case, the attacker requires the phone number to identify the subscriber on WhatsApp. Also, the victim's privacy settings must allow the attacker to view the victim's WhatsApp profile. First, the attacker sends a message to the target recipient. Once it is received, the recipient's WhatsApp application will list it in the inbox. For the attack to succeed, the recipient mustn't block or delete the attacker's contact. Later, the attacker opens his active chat window corresponding to the recipient and composes a message but does not send it. Due to the "typing notification" feature, the recipient can see that the attacker is typing in the chat window. During this procedure, the network triggers paging requests destined for the recipient's LTE devices.

4.2.4 Active attack - leak fine-grained location (L3)

Once the attacker determines a TA and cell where the subscriber is present, the next goal is to find his/her location more precisely. We now demonstrate two methods in which the attacker exploits a specification and an implementation vulnerability to this end.

1. Via measurement reports. We consider a subscriber who is initially attached to a legitimate eNodeB. The attacker forces him/her to attach to a `rogue eNodeB` by applying the techniques mentioned in section 3.3.2. The subscriber's UE completes RRC connection procedures and initiates a *TAU* procedure with the attacker's `rogue eNodeB`. Next, UE enters into a CONNECTED state. The attacker creates a *"RRC Connection Reconfiguration*" message with different cell IDs (possibly 3 or more neighbor cells) and necessary frequencies and sends it to the UE without any protection. After receiving this unprotected message, UE computes the signal power from neighboring cells and frequencies and sends an unprotected *"Measurement Report*" message to the `rogue eNodeB`.

If the UE supports *'locationInfo-r10'* feature [12], it includes its GPS coordinates in the measurement report. This feature is not yet widely supported by current smartphones - however, one of our test phones exhibited this behavior.

2. Via RLF reports. In this attack, two `rogue eNodeBs` are operated in the same cell where the subscriber is present. Initially, eNodeB 2 is OFF and eNodeB 1 ON to

create an RLF scenario to the UE. The UE initiates a connection to eNodeB 1 and enters into the CONNECTED state, as shown in Figure 4.3. We turn OFF eNodeB 1 upon receiving a TAU request from the UE. At the same time, eNodeB 2 is turned ON. Meanwhile, UE detects that it has lost sync with the eNodeB 1 and starts RLF timer (T310).

When the RLF timer expires, UE creates an RLF report [12] and goes into IDLE mode. In this mode, UE starts the cell selection procedure as specified in [10] to attach to eNodeB 2. As before, UE enters the CONNECTED state with eNodeB 2 and indicates the availability of the RLF report in a TAU message. Upon receiving this message, the attacker sends an unprotected *"UEInformationRequest*" message to UE from eNodeB 2, thereby requesting UE to send the RLF report to eNodeB 2 in response. As a result, UE sends the resulting response in an unprotected *"UEInformationResponse*" message containing the RLF report. This report includes failure events and explicitly signal strengths of neighboring eNodeBs.

Besides, according to the LTE specification [33], the RLF report can include GPS coordinates [12] of UE at the time it experienced the radio failure.

Determining subscriber's precise location

The aforementioned measurement and RLF reports provide signal strengths allowing the active attacker to calculate the distance between the UE and the `rogue eNodeB`. This calculation is performed using a trilateration technique as described in [43]. Figure 4.4[5] shows how this technique is used to determine the subscriber's location. The distance estimates are calculated as d1, d2, and d3 for three neighboring base stations. The zone of the intersection point of three circles is the subscriber's approximate location in a cell. However, if *'locationInfo-r10'* feature is supported in measurement and RLF reports, accurate location can be determined using GPS coordinates as exhibited by one of our test UEs (as seen in Wireshark) in Figure 4.5.

Several of the vulnerabilities we exploited are in the LTE specifications rather than in the UE's baseband software. Therefore, all LTE-capable UEs conforming to these specifications are affected. For evaluation, we selected popular smartphones incorporating baseband implementations from top vendors who dominate the market share worldwide [87]. We successfully verified that all these phones are vulnerable to our attacks. Besides, all UEs have the implementation vulnerability leading to attack L3.

[5]Underlying Map source - OpenStreetMap

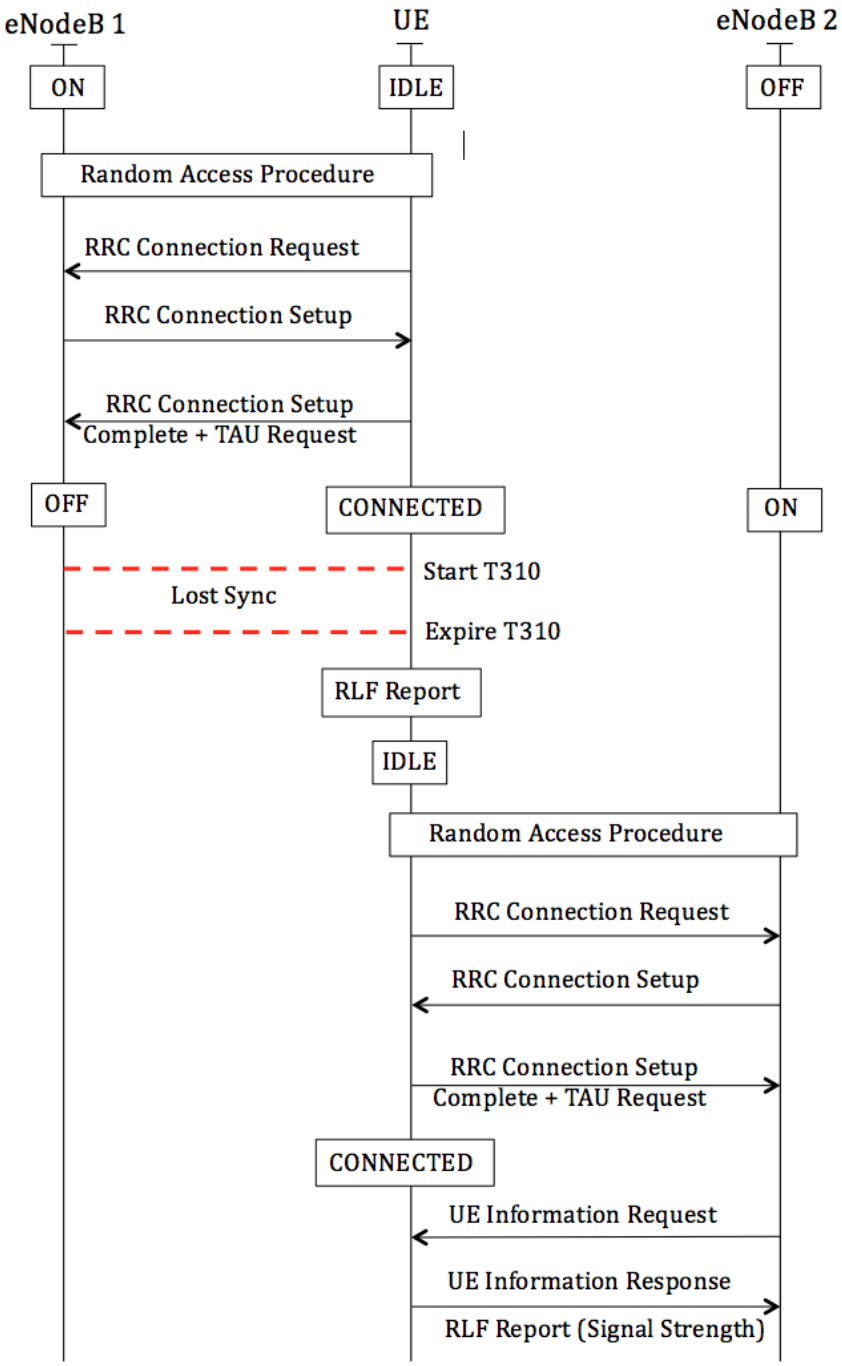

Fig. 4.3.: Retrieving RLF report from UE (L3)

Fig. 4.4.: Determining subscriber's precise location using trilateration (L3)

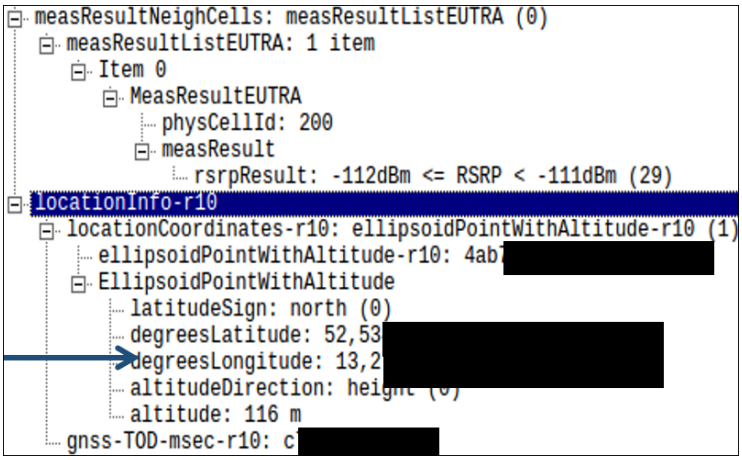

Fig. 4.5.: Subscriber's precise location via GPS coordinates

4.3 Rogue Devices in SON

In this section, we evaluate the resilience of a self-organized LTE network against our active threat models. The key idea is to operate `rogue eNodeB(s)` that leverage legitimate mobile devices as covert channels to inject false measurements and network configurations into the SON ecosystem. Further study the resulting impact on the network and its subscribers. In this process, we uncover three new types of DoS attacks (labeled as S1, S2, S3) in LTE networks and demonstrate them using our low-cost testbed costing only 300 dollars. Also we confirmed their effectiveness against commercial LTE networks. In particular, the active attacker can shut down network services for a certain period in a 2 km^2 area. Furthermore, the active attacker can completely block network services to UEs in a targeted area and also downgrade them to use less secure 2G and 3G network services.

4.3.1 SON Poisoning Attacks (S1 - S3)

An active adversary can poison the SON data to create persistent DoS attacks against both the mobile operator's network and end-users mobile devices. Before this, a set of preliminary experiments are conducted to sketch various cell and eNodeB attributes.

A. Preliminary measurements

The active adversary requires the knowledge of LTE network deployments of the target region. Such deployment information can be acquired through mobile network databases available online such as OpenCellID [160]. However, this information may not be accurate, since networks are continuously evolving and changing their configurations to adapt coverage requirements. Hence, for our attacks, we gather network deployment information by being a passive adversary initially. As an example, we plotted the cell coverage Figure 4.6[6] of a specific area in a metropolitan city using `passive listener`. Due to the availability of several frequency bands, operators choose to deploy multi-frequency networks.

We plotted two eNodeBs, each operating 2 cells and each with a different EARFCN. LTE deployments allow neighboring cells/sectors to use the same EARFCN, but they should operate with a different PCI for avoiding conflicts with each other. The cell sizes indicate that they are microcells, typically having a cell radius of 500 m to 2 km.

[6]Underlying Map source - OpenStreetMap

In certain regions, we observed that operators deployed up to 6 EARFCNs depending on the network coverage demand and availability. The attacker strategically chooses his position by targeting a cell or a group of cells based on the capability of the attack setup. To perform the DoS attacks in this paper, first, the attacker passively collects all the broadcast information such as EARFCN, PCI, ECGI of the surrounding cells. Next, the attacker turns active and operates a rogue network.

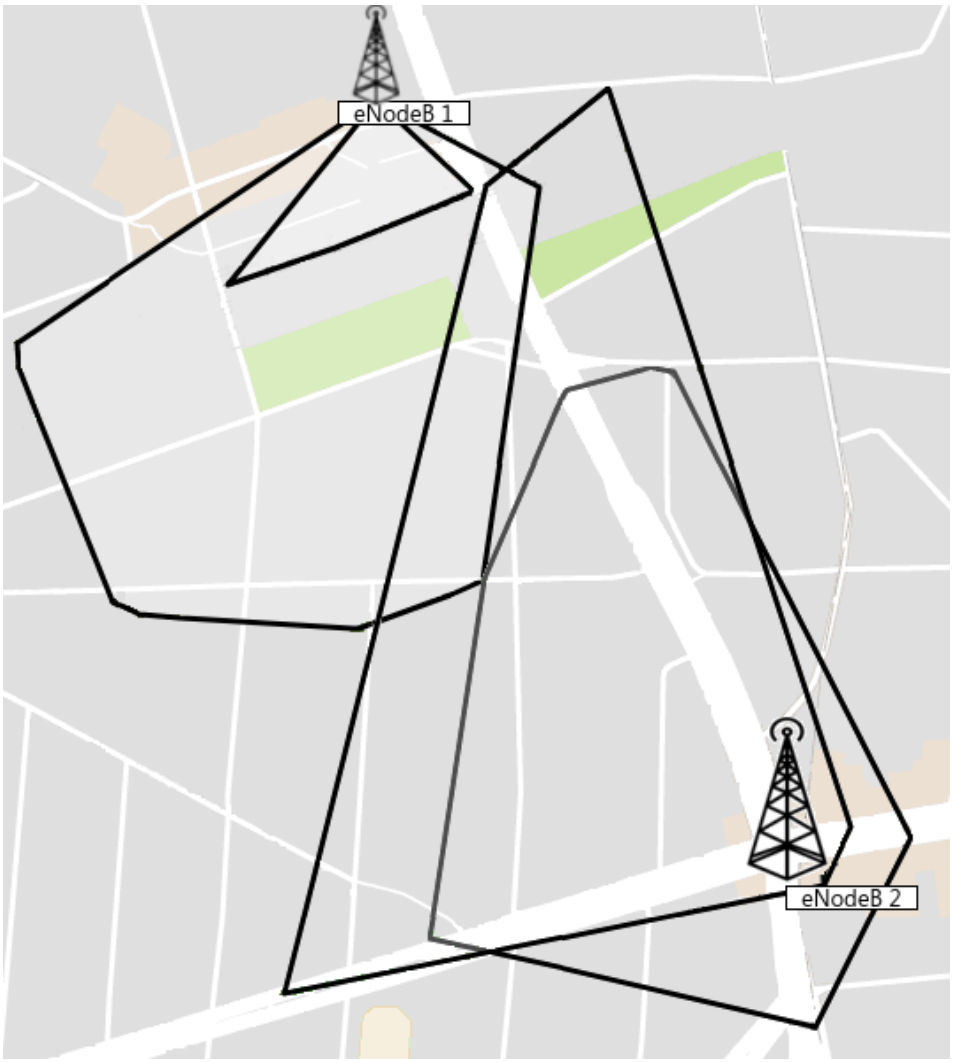

Fig. 4.6.: LTE multi-frequency network deployment in a target area

B. DoS Attacks in Practice

We describe three types of DoS attacks in this section. The attacks S1 and S2 are directly targeted at the network, and S3 is targeted at mobile devices. However, the consequences of all attacks affect both subscribers and operators. Further, we perform an experimental evaluation of these attacks with our test phones.

1) X2 signalling flood (S1)

We operate a minimal `rogue eNodeB` to broadcast (spoof) the PCI and ECGI of a legitimate operator owned cell that is distantly located (at least 2 km) from a targeted eNodeB (operator owned). It allows us to introduce a new legitimate cell into the operational network geographically. UEs in the *RRC CONNECTED* state detect the presence of our `rogue eNodeB` and communicate the PCI to the targeted eNodeB via *measurement reports*. By spoofing legitimate identities, the targeted eNodeB is deceived that a legitimate eNodeB is closely located and it can perform handovers. But, since the PCI is not available in its neighbor list, the ANR procedure is invoked by the target eNodeB to retrieve the ECGI spoofed by our rogue cell. Upon acquiring ECGI from the UE, the target eNodeB acquires X2 IP address from the OAM and initiates an X2 connection with the legitimate eNodeB that is located 2 kms away. Due to a lack of authorization to control the X2 setup, both eNodeBs connect over the X2 interface, create entries in their NR tables, and share information over the X2 interface. This generates ample X2 signaling and wastes resources for the network. Repeating the similar spoofing attack for multiple eNodeBs cause heavy singlaing over X2 interface. Later in S3 we describe how the injected rogue measurement data can cause handover failures in the network.

2) Cell Outage (S2)

A `rogue eNodeB` is set up to impersonate a legitimate cell by spoofing its EARFCN, PCI, and ECGI parameters. Eventually, either via passive network scanners or measurement reports or X2 reports, the real eNodeB detects a PCI collision and raises the alarm to the OAM. To resolve the collision, eNodeB initiates a PCI optimization procedure that involves automatic reboot and selection of a new PCI either from the OAM or by auto-configuration where eNodeB scans for all neighboring cells. Hence, the region under the eNodeB's coverage would have service outage for its subscribers for 8 to 10 minutes [80] (depending on the vendor configurations).

3) Handover Hijacking (S3)

We operate a `rogue eNodeB` to spoof EARFCN, PCI, and ECGI values of a target eNodeB (which is in the NR table of a source eNodeB) and cause a PCI confusion to the UE as illustrated in Figure 4.7. By doing so, both UE and source eNodeB are fooled to believe that the `rogue eNodeB` is the target eNodeB. Since UE observes stronger power levels from the `rogue eNodeB`, it conveys `rogue eNodeB`'s PCI and measurements to the source eNodeB via *measurement reports*. UE sends the measurements of `rogue eNodeB` but, source eNodeB considers them to be the

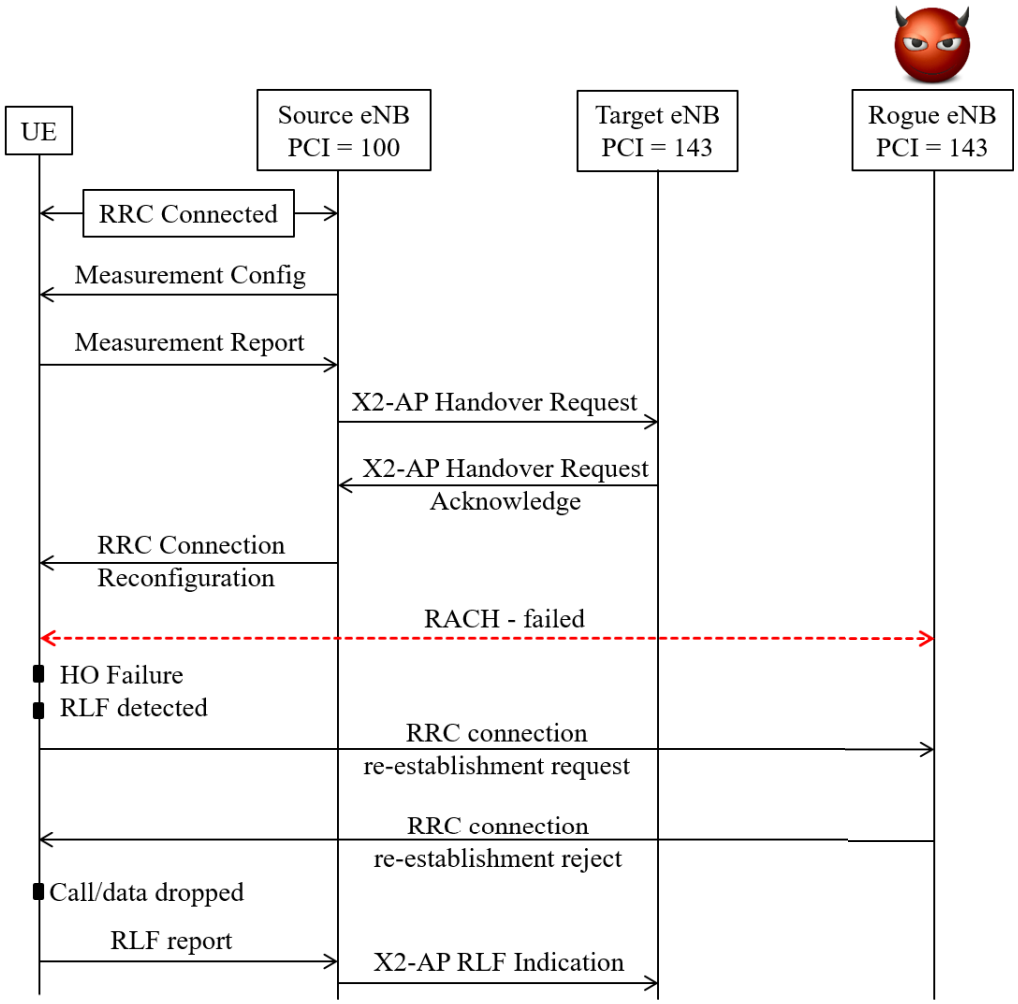

Fig. 4.7.: LTE Handover Hijacking

measurements of target eNodeB for two reasons. First, the reported PCI is registered in the NR table as target eNodeB, and second, the lack of verification on the received measurements. However, as the measurements satisfy the handover requirements, source eNodeB performs necessary handover procedures with the target eNodeB and issues a handover command to the UE.

Now, UE is confused by the dual PCI operation (PCI confusion), but since the signals from the `rogue eNodeB` are dominating the one from the target eNodeB, UE prefers to connect with `rogue eNodeB`. Thus, UE initiates the RACH procedure and experiences handover failure since `rogue eNodeB` does not possess the RACH information required to grant access to the UE. As a result, UE creates and caches an RLF report so that it can indicate this failure event to the legitimate network. The RLF report is logged with the details of `rogue eNodeB`, such as its PCI, ECGI, EARFCN, and power measurements, which are the same as target eNodeB.

Next, the UE re-establishes a connection to the rogue eNodeB to recover from this failure by sending a *RRC connection re-establishment request* message. Since rogue eNodeB cannot offer a legitimate service, it rejects the request with a *RRC connection re-establishment reject* message making the UE terminate call/data procedures and induce DoS to it. UE remains connected to the rogue eNodeB until the attacker releases it. When UE retains a legitimate connection, it transfers the RLF report to the network. Meanwhile, the target eNodeB, which was expecting a connection request from the UE as part of the handover, notices that the handover event was unsuccessful and registers this failure event locally or with the OAM, which can later be used to determine the handover KPI. By creating such handover failures continuously, the KPIs of the legitimate eNodeB eventually decline and this causes activation of MRO procedures.

4.3.2 Evaluation and Impact

We leveraged our experimental setup to practically evaluate the feasibility, persistence, and impact of our attacks. The attack S1 targeting the network is not assessed as it directly interferes with the commercial eNodeB signals and disrupts regular mobile network communications. Therefore, we evaluated the attack S1 and S3 since they are leveraged against our test phones. LTE networks implementing the SON features discussed in this paper are vulnerable to our attacks. Although vendors may have different implementations but based on our study, we identified that the basic methodology of SON operation is identical across them. Further, we discussed both S1 and S3 attacks with mobile network operators and confirmed their feasibility in practice.

While performing S1, we could successfully verify with our *cell_logger* that the network triggered an ANR procedure. The impact of S1 is visualized by executing S3, where we impersonate a legitimate cell (spoofed in S1). To demonstrate S3, we utilized a SON from one of the largest European operators. We set up our rogue eNodeB to spoof the PCI of the cell located adjacent to the test network. The range of our rogue eNodeB is restricted to our test phones only, since we performed S3 experiment in a Faraday cage. The test UEs are connected to the test network and are engaged in VoLTE call. When the rogue eNodeB is turned ON with high power, in less than a second, all 4 UEs received handover command from the source eNodeB. Following this, all four UEs attempted for a connection request on our rogue eNodeB but failed to acquire one. As a result, the UE suspends VoLTE call and creates an RLF report containing the spoofed legitimate PCI and ECGI of a neighboring eNodeB and forwarded it to the legitimate network when released from the rogue eNodeB.

4.4 Trade-off Analysis

We explain the background behind the vulnerabilities in LTE radio management protocols by considering various trade-offs between security and criteria like availability, performance, and functionality. We show that the equilibrium points in the trade-offs had shifted today compared to where they were when the LTE security architecture was being designed. Table 1.1 summarizes our analysis.

4.4.1 Possible trade-offs and Discussion

Security vs. Availability. We demonstrated a vulnerability in the LTE RRC protocol specification that allows the adversary to obtain unprotected measurement reports from UEs (L3). We consider the following two angles to explain the trade-off. On the one hand, in some cases, network operators require unprotected reports for troubleshooting purposes. In particular, if the UE is not able to establish a connection with the eNodeB, then it may be necessary to send measurement reports without protection to allow the network to identify the technical reason behind the fault. This seems to be the reason behind the note in LTE RRC specification, which points out that the 3GPP Radio Access Network (RAN2) working group decided to permit UEs to send reports even without security activation [12].

On the other hand, during the design work for the LTE security architecture, the 3GPP security working group (SA3) suggested that all RRC protocol messages should be sent in encrypted form [17]. Hence, the vulnerability in RRC protocol specification is a conscious exception to this security design guidance [12]. Clearly, 3GPP has concluded that in this particular case, the requirement of having network availability all the time to all UEs outweighs security concerns related to subscribers' privacy.

Next, we demonstrated in S3 that the LTE handover procedure could be hijacked and is vulnerable to DoS attacks and can be explained as a trade-off between availability and security. Firstly, 3GPP standardized the handover procedure to be very fast and seamless to ensure that the UE receives a persistent network connection when traversing across cells. Hence the source eNodeB transfers UE to the best available target eNodeB that is selected based on the power measurements reported by UE. For this, the eNodeB completely trusts the measurements received from the UE. Secondly, the source eNodeB is interested in knowing all surrounding eNodeBs that are visible to the UE so that it can discover new eNodeBs and establish a relationship with them through the ANR function. Therefore, as a standard procedure [8], UE reports all visible cells on one or more configured frequencies and facilitates the

eNodeB to establish neighbor relations. However, UE might report information spoofed by rogue cells, and the source eNodeB does not verify the authenticity of this information. Source eNodeB cannot distinguish between real and rogue cells. Clearly, in both cases, the authenticity of the measurement information is ignored, and seamless connectivity and ANR are considered highly significant. In other words, the availability of the network is preferred over security.

Security vs. Functionality. Our attacks that leak coarse-grained location information by using social network messaging services (L2) is an example of the tension between security and functionality. The introduction of TCP/IP based data communication on top of mobile communication infrastructures has dramatically expanded the functionality that third-party developers can build for these networks. But such a flexible software architecture makes it harder to avoid or detect the type of vulnerability that led to this attack. Furthermore, even if individual app developers would fix their applications (e.g., Facebook could change the application architecture of their Messenger application to ensure that messages that end up in the "Other" box do not trigger paging requests), other application developers may make similar mistakes. To avoid such vulnerabilities in a modern mobile communication system like LTE, it would require significant developer outreach and education to help them design and build mobile-optimized applications [64].

Security vs. Performance. A third example we observed is that in some operator networks, GUTIs are not changed even after three days of usage (L1). LTE specifications do not mandate any GUTI reallocation frequency, leaving it as a policy decision to operators. One possible reason for the low GUTI-change frequency is the operators' wish to reduce signaling overhead by trading off privacy.

SON Poisoning. SON is all about automating network management functions to boost its performance and ensure that the subscribers are provided with the optimal quality of service. However, a significant problem of LTE SON is its dependency on unreliable input sources. As shown in our poisoning attacks (S1-S3), various commercial LTE SON implementations blindly trust the measurements from the UE's leading to unwanted consequences like potential signaling flood in the network and cell outages. In particular, we highlight three problems:

First, the ANR function builds X2 connections between eNodeBs for seamless handovers and fast service to subscribers. As explained in the LTE tradeoff above, network availability is preferred over security by the ANR function. Furthermore, the lack of sufficient control to authorize the setup of X2 relations is a clear example of inadequate design.

Second, PCI optimization ensures that adjacent LTE cells do not operate with the same PCI and avoid a collision. Since collisions lead to handover failures and affect subscriber experience, the eNodeB quickly restarts without any further interrogation about the eNodeB that created this collision. On the one hand, this guarantees reliable and fast service recovery to the subscribers, but on the other hand, eNodeB lacks a mechanism to interrogate the intention behind this collision. Further, decision-based on the vulnerable source such as UE reports endangers the security of the system.

Third, the MRO function allows the network to enhance and adjust its coverage in real-time based on various handover statistics. In particular, RLF and handover reports are critical factors in controlling handover settings of a cell and that enable superior subscriber experience when traversing across cells. Though such reports are collected in an encrypted form, nevertheless they are vulnerable and contain rogue information that can severely affect subscriber experience. Clearly, MRO function excluded the security implications that could arise from vulnerable input sources that could drive the entire network into an unstable state.

4.5 Potential Mitigations

We now discuss potential countermeasures against attacks demonstrated in earlier sections. In particular, we identify protocol-level and operational fixes that can be implemented by baseband vendors and mobile network operators. Some of these countermeasures are much more straightforward than others. Similarly, some of our proposals may cause hidden dependencies, and more changes may be needed in the networks than what is apparent from our descriptions.

Protection against location leaks. LTE broadcast information includes subscriber identities that enable tracking of UEs (L1 and L2). The broadcast information must be sent in unprotected messages from the LTE system design perspective. There are two solutions to avoid UEs being tracked. One solution is to protect broadcast messages using a public key mechanism, but this requires relatively significant changes in LTE protocols. According to [55], 3GPP decided against the usage of public-key mechanisms because its implementation cost was deemed too high. However, our findings may have changed the equilibrium in this trade-off. Consequently, a scheme where public/private keys are used only for network elements could be justified now. Messages from the network could be signed by using a public key digital signature mechanism; UEs would then be able to verify the authenticity of such messages.

Thus it prevents rogue network elements from sending false information, e.g., fake messages indicating radio link failures (L3). Messages towards the network could be encrypted using the public key of the serving operator; UEs would not need to send their identities in the clear to initiate network *Attach* procedure. It is not easy to protect paging messages with public-key mechanisms, even if we would have public keys for UEs because UEs would have to try to decrypt all paging messages. All these proposed fixes require ensuring global availability and verifiability of public keys of network components (such as eNodeB). The first phase of 5G security standardization has been completed, and notably, the PKI mechanism is utilized to conceal the transmission of IMSI over the air.

The second solution is more realistic as it does not require a change in protocols. Network operators would re-allocate GUTIs often enough to avoid tracking. One of the national operators to whom we reported our findings, acknowledged the feasibility of our attacks, and already configured their networks to prevent tracking based on GUTIs. This solution would protect against passive attacks (L1). A certain degree of protection against semi-passive adversaries could be achieved by making the adversary's actions more visible to the subscriber. There are already such tools [158, 148] available, but the challenge is in making them usable and useful to all types of subscribers. LTE specification vulnerability regarding UEs sending measurement reports without integrity protection needs to be addressed by the 3GPP security group for all baseband vendors to implement the fix in their products eventually. The simplest solution is to transmit measurement reports only after setting up the security context; this solution is standardized in the LTE from release 13 specifications [8].

Besides, our experiments in 2019 across several LTE networks worldwide reveal that operators have update their configurations and GUTI freshness and randomness is guaranteed for every LTE attach procedure. Therefore, LTE networks and UE's implementing latest 3GPP specifications and our proposed mitigations are resistant to our location leak attacks (L1 - L3).

Reliable SON. Operators prepare concise data for their network planning and deployment. Later during the operational state, they rely on the UE reports to troubleshoot coverage holes and make adjustments. Instead of entirely relying on the UE measurements, operators can use dedicated network listeners [127] that continuously monitor changes in the network architecture and track the addition and deletion of new elements in their ecosystem. Moreover, such listeners are not power constrained like UEs and hence can apply intelligent techniques to identify fake measurement

data. This data can be cross verified against the data acquired form *measurement reports* and the existing network planning data to detect anomalies.

Alternatively, network operators can learn by co-relating existing network data (e.g., less than 12 hours before) and verify the correctness of newly gathered data. It is unlikely that their difference will have a large delta. In the case of larger deltas, OAM can initiate an investigation into the affected eNodeBs and identify the root cause of this change. E.g., eNodeBs can acquire signal strengths along with the angle of arrival of the user to learn specific locations where handover occurred (successfully) and decide whether to allow or disallow particular handover procedures. Although this method is proposed in [145], authors apply this to enhance the network performance by controlling unwanted handovers, which implies that it can be used to control handovers to rogue eNodeBs. Further, this method does not require any changes to existing LTE standards, especially on the UE side.

3GPP has initiated a new study item [22] to brainstorm various solutions for solving the fake base station problem in mobile networks. It also addresses the SON poisoning attacks described in this chapter. However, it is still under investigation, if standardized solutions could be introduced to improve security and privacy issues in SON, or guidelines could be given to vendors to make the implementation better.

5 Trading Security for Performance

LTE is an all-IP network that is heavily driving the use of content-rich media through social networks and smartphone applications. New applications such as online gaming, mobile TV, cellular Vehicle to Vehicle (V2V) demand high data rates, and low latencies. In contrast, NB-IoT and LTE-M type devices desire lower data rates for intermittently receiving or sending a few bytes of data. However, this may incur heavy signaling in the network, given the massive number of IoT devices. To offer such variable performance, operators should maintain an optimized control plane. In any networked system control plane is the key to access the data plane. Thus a damaged control plane will also destroy access to the data plane.

One of the biggest problems faced by operators is signaling storms [150, 115] over the control plane. For instance, an attacker can flood the network with an enormous amount of messages to cause a DoS to entire area/cell. Differently, storms can also occur in legitimate situations such as the rapid opening and closing of network connections by a massive number of IoT devices. Further, at public gatherings (e.g., a protest) can cause similar effects where hundreds of devices perform mobility management procedures and overload the control plane. It can lower the network performance if the operators do not effectively manage such traffic patterns. 3GPP does not consider flooding or jamming type attacks as a significant threat as they are not persistent; the network stabilizes once the attacker shuts down the rogue device(s). However, persistent DoS attacks are a serious concern and should be addressed both by operators and standards.

Network performance on mobile networks is inherently linked to the battery performance of the end-user device. Conserving device power has been a critical concern for all the involved parties, including carriers, device manufacturers, and end-users. Both operators and standards have adopted several approaches over the years to optimize control plane procedures. For instance, solutions in [103] cache user sessions, device security, and capabilities to avoid their retransmissions during EMM procedures. The MME caches such information and re-uses when the device connects to the network, thus avoiding signaling overhead. The 3GPP has designed the EMM protocols such that the network can control the connection attempts made by mobile devices. Although the number of attempts varies between different baseband imple-

mentations, these protocols exist right from GSM that were designed to primarily offer performance.

Security is an essential tool in the network, but LTE security improvements should not degrade the performance. Operators have full control of the security in the network; however, as a common subscriber/consumer, there are no methods available to detect the security comprises made by the operators to improve network performance. For instance, encryption of the data plane is an optional feature in LTE. Nevertheless, misconfigurations or performance goals may lead to the use of null encryption algorithms that threaten the user's privacy. Though encryption has remained an optional feature in the context of mobile networks [23, 1, 18], authentication has been made mandatory from 3G networks. Nevertheless, operator configurations may refrain from executing AKA protocols during EMM procedures such as during a TAU [76] to reduce signaling. Since one round of AKA protocol involves deriving a new NAS and AS security context that costs additional signaling over the control plane, as discussed in section 2.2.2. It also triggers signaling between home and serving network to request new authentication vectors.

In this chapter, we bring to light a new set of vulnerabilities in the subscriber and device management protocols that trade security for performance. Such choices comprises subscriber privacy and lead to persistent network unavailability. We analyze these trade-offs and demonstrate various attacks using our testbed. Further we realize the impact of using commercial devices and networks.

5.1 Security Weaknesses in LTE Device and Subscriber Management

The MME manages the LTE devices and subscribers through various NAS protocols defined in [16]. Especially, the registration process is organized into various protocol interactions between UE, eNodeB, and MME, as shown in Figure 2.2. This chapter discovers various security and privacy issues in the LTE registration process. We demonstrate **fingerprinting attacks in section 5.2** and **DoS attacks in section 5.3**. Further, we discuss the rationale behind these weaknesses and propose mitigations that can be (and are) applied to the 4G and 5G NAS specifications. We first study the weaknesses in three major functions of the LTE registration procedure, namely, Network access control, Device capability transfer, and AKA protocol.

5.1.1 Network Access Control

To control UE's mobility and to register with the EPC, there are two NAS protocol messages and are described below.

TAU and Attach procedure. One of the primary function these protocol messages is to inform the network about UE's present location in the serving area of the operator. This allows the MME to offer network services to the UE, e.g., when there is an incoming call. For this purpose, UE notifies the MME of its current TA by sending a *"TAU Request*" message and also includes its network modes. Generally, UE operates in various network modes for voice and data connections, as stated in [16]. Still, for this work, we focus only on two modes: i) EPS services (i.e., LTE services), ii) both EPS and non-EPS (i.e., GSM or 3G) services.

During a *TAU* procedure, the UE and MME agree on one of these modes depending on the type of subscription (for example, USIM is subscribed for LTE services), and network capabilities supported by the UE and by the operator in a particular area. During *TAU* procedure, the network may deny some services to UEs, for example, if the subscriber's USIM is not authorized for LTE services or if the operator does not support certain services in the serving area. The LTE specification [16] defines certain EMM procedures to convey such denial messages to UEs. Specifically, these are sent in *"TAU Reject*" messages which are exempted to operate even in the absence of control plane security i.e., they do not require integrity protection.

5.1.2 UE Capability Transfer

A UE supports several capabilities for various LTE services and operations. They are classified into core network capabilities [32, 5], and radio access capabilities [8, 27] and are communicated to the MME and UE during the registration process, as shown in Figure 2.2.

Cellular UE. The core network capabilities are sent to the network in the *attach request* message and contain non-radio related capabilities, e.g., security algorithms, telephony features, etc. Further, UE reports its support for 3G, 2G, and CDMA networks if available. Similarly, radio access capabilities provide radio aspects of the UE, such as supported frequency bands, receive and transmit capabilities, etc. They are sent in *UE Capability Information* message. eNodeB stores these capabilities at the MME until UE de-registers from the network. In other words, these capabilities are used by the eNodeB until UE de-registers from the network.

Cellular-IoT UE. NB-IoT and LTE-M device capabilities differ from the traditional UEs. As the main purpose of these devices is only to send/receive small amounts of data intermittently, the 3GPP introduced Power Saving Mode (PSM) into LTE specifications in 2015 [32] to lower power consumption in such devices. PSM is a state where UE is powered-OFF but remains registered with the network. Precisely, the 3GPP indicates to turn off the baseband and thus radio operations but, applications (or sensors) can still operate depending on the device settings. A UE can request the use of PSM by including a timer *T3324* in the *Attach* or *TAU Request* messages. *T3324* defines the period that the UE stays active before entering into PSM. During this active state, UE monitors the eNodeB channels for incoming messages from the network.

We found three weaknesses in the capability exchange protocols between UE and network.

- Since the *Attach Request* message containing the capabilities is always sent unencrypted [32], and hence it is accessible over the air to both passive and active adversaries.

- To protect against MiTM attacks, the LTE security architecture mandates reconfirmation of previously negotiated security capabilities after the AKA procedure [24]. In particular, the network sends an integrity-protected message, including the list of supported security algorithms that it previously received from the UE. However, there is no similar confirmation for other capabilities.

- Mobile network operators request radio access capabilities prior to the RRC security establishment, as shown in Figure 2.2. As a result, UE's radio access capabilities are transferred in plaintext and are accessible to both passive and active adversaries.

5.1.3 Authentication and Key Agreement Protocol

We identified a logical vulnerability in the AKA protocol used in the 4G network, as described in section 2.2.2: the protection mechanism of the SQN can be defeated under specific replay attacks due to its use of Exclusive-OR (XOR) and a lack of randomness. We show how to leverage this vulnerability to break the confidentiality of SQN, thus defeating the purpose of a dedicated protection mechanism and breaking an explicit privacy requirement [1]. This vulnerability is also applicable to the AKA variants used in 3G and 5G networks.

A. Logical Vulnerability

Our attack vector exploits a lack of randomness and the use of XOR in *AUTS*, more precisely in the concealed sequence number $CONC^* = SQN_{UE} \oplus AK^*$ where $AK^* = \mathsf{f5}^*(R,\ K_{IMSI})$. The value R is extracted from the challenge $R, AUTN$ received by the *UE*. Therefore, if the *UE* receives two times the same challenge $R, AUTN$ and yield two synchronization failures, then the two concealed *SQNs* will be of the form: $CONC_1^* = SQN_{UE}^1 \oplus AK_1^*$ and $CONC_2^* = SQN_{UE}^2 \oplus AK_2^*$ such that

$$AK_1^* = \mathsf{f5}^*(R,\ K_{IMSI}) = AK_2^*.$$

Therefore, an attacker having a genuine challenge $R, AUTN$ for some *UE* can transmit it to the *UE* at two different times t_1 and t_2, retrieve values $CONC_1^*$ and $CONC_2^*$, and compute:

$$\begin{aligned} CONC_1^* \oplus CONC_2^* &= (SQN_{UE}^1 \oplus AK_1^*) \oplus \\ &\quad (SQN_{UE}^2 \oplus AK_2^*) \\ &= SQN_{UE}^1 \oplus SQN_{UE}^2 \end{aligned}$$

where SQN_{UE}^i is the value SQN_{UE} at time t_i. We show in the next section that by cleverly choosing several timestamps t_i's, the attacker can exploit values such as $SQN_{UE}^i \oplus SQN_{UE}^j$ to break the confidentiality of SQN.

B. Breaking the Confidentiality of SQN

We show how an active attacker who knows any *UE*'s identity (temporary, permanent, or encrypted)is then able to learn the n least significant bits of SQN_{HN}, stored in the *HN*. The attacker (using a `rogue UE`) first fetches $2^n + 2$ successive, fresh, authentication challenges intended for the targeted *UE* and replays (using a `rogue eNodeB`) a total of $2(n+2)$ of them to the *UE*.

The interaction is depicted in Figure 5.1. The attack ends with an offline computation using `algo`$(\cdot)$ which takes fetched *AUTS* messages as inputs and returns the n least significant bits of the sequence number SQN_{HN}.

In a nutshell, the attack consists in choosing appropriate injections and timestamps t_i such that the attacker can retrieve values $\delta_i = SQN_{HN} \oplus (SQN_{HN} + 2^i)$ for $1 \leq i \leq n$. We then explain how one can infer from the δ_i's the n least significant bits of SQN_{HN}. Finally, we also show that under certain circumstances (*i.e.,* when the *UE* is performing a lot of authentication sessions when in the attack area), a far less costly variant of the attack (only $n + 2$ injections) achieves the same goal.

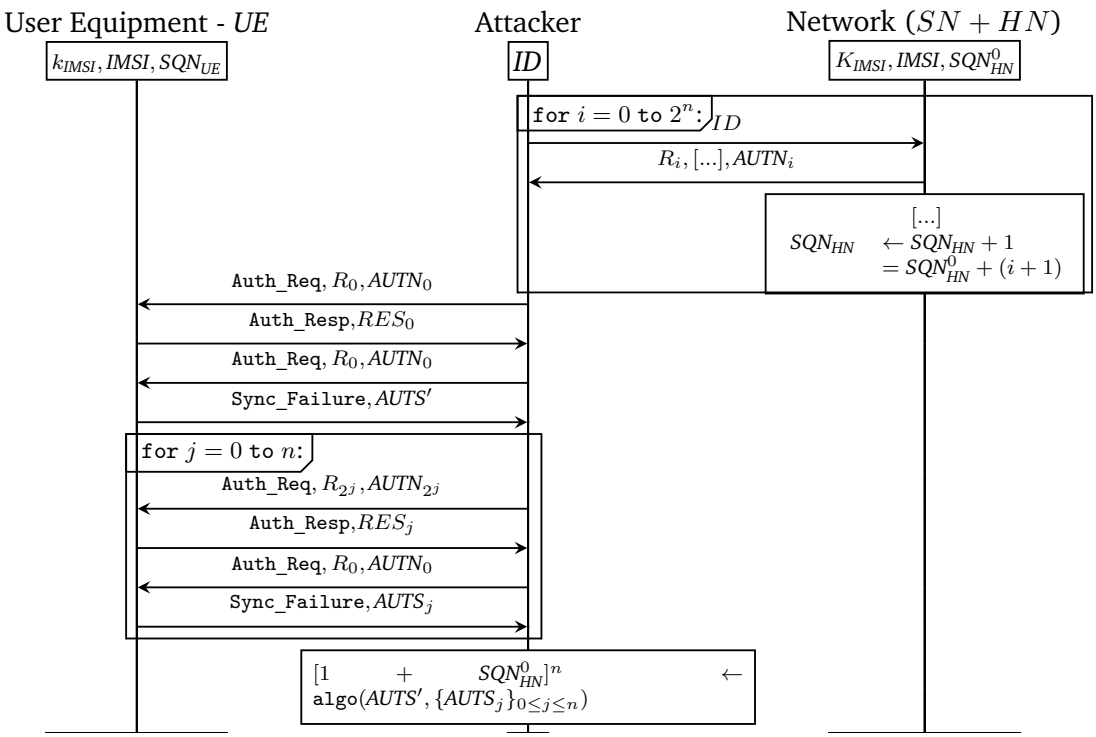

Fig. 5.1.: Sequence Number Inference Attack (where SQN^0_{HN} is the initial *SQN* for *IMSI* stored in the *HN* and $[X]^n$ denotes the n least significant bits of X).

We describe the attack and our inference algorithm when the *HN* increments SQN_{HN} by 1 after each successful authentications as described in section 2.2.2. Our attack works for any such increment; the interaction is always the same, and we designed a generic algo$(\cdot)$ parametrized by the increment used by the operator. However, for the sake of clarity, we only describe here our attack and our generic algorithm for an increment equal to 1. Note that the whole algorithm might infer more than n bits for some inputs; we report on practical results of this algorithm in section 5.2.2.

Fetching Data. In a first phase (loop for $i = 0$ to 2^n from Figure 5.1), the attacker needs to fetch consecutive challenges $R_i, AUTN_i$ intended for the targeted *UE*. This is made possible by the fact that, in the AKA protocol, *UE* receives such challenges prior to authentication but after identification. Therefore, an attacker only needs to know one valid identity of the targeted *UE* (*e.g.,* IMSI, temporary identifiers such as TMSI, or encrypted permanent identities such as SUCI) in order to (partly) impersonate the *UE* to the *SN* (and the corresponding *HN*) and get those challenges. We will explain in section 5.2.2 how this can be easily done in practice. Note that because SQN_{HN} is incremented by 1 after the computation of every challenge, $R_i, AUTN_i$ is computed based on some SQN value (that we denote by $SQN_{HN}(AUTN_i)$) equals to $SQN^0_{HN} + i$.

Immediately after the first phase, the attacker injects the first challenge he obtained: $R_0, AUTN_0$. From the *UE*'s perspective, this is a genuine challenge (the MAC verification `(i)` succeeds) that has never be received before and that is based on a recent enough $SQN_{HN}(AUTN_0) = SQN^0_{HN}$ (the freshness verification `(ii)` succeeds). At this time (before the second loop), SQN_{UE} equals $SQN^0_{HN} + 1$. Then, the attacker injects again the challenge $R_0, AUTN_0$ yielding a synchronization failure containing some $AUTS' = \langle c', MAC^* \rangle$ message where the conceal SQN equals:

$$c' = (SQN^0_{HN} + 1) \oplus \mathsf{f5}^*(R_0, K_{IMSI}).$$

In the last phase (loop for $j = 0$ to n from Figure 5.1), the attacker injects $R_{2^j}, AUTN_{2^j}$ that is accepted by the *UE*, in order to make the *UE* updates its SQN_{UE} to the value

$$SQN_{UE} := SQN_{HN}(AUTN_{2^j}) + 1 = SQN^0_{HN} + 2^j + 1.$$

After each such injection, the attacker then injects again the challenge $R_0, AUTN_0$ provoking a synchronization failure containing some $AUTS_j = \langle c^*_j, MAC^*_j \rangle$ where:

$$c_j = (SQN^0_{HN} + 2^j + 1) \oplus \mathsf{f5}^*(R_0, K_{IMSI}).$$

Inference Algorithm. We now describe `algo`$(\cdot)$ that takes the $n + 2$ fetched *AUTS*'s messages (*i.e.,* c', c_j for $0 \leq j \leq n$) as inputs and outputs the n least significant bits of $1 + SQN^0_{HN}$. Recall that $c' = (1 + SQN^0_{HN}) \oplus \mathsf{f5}^*(R_0, K_{IMSI})$ and $c_j = (1 + SQN^0_{HN} + 2^j) \oplus \mathsf{f5}^*(R_0, K_{IMSI})$. Therefore, for any $0 \leq j \leq n$, it holds that:

$$c' \oplus c_j = \quad (1 + SQN^0_{HN}) \oplus (2^j + 1 + SQN^0_{HN}).$$

We note δ_i the quantity $c' \oplus c_i$. One has that $\delta_i = (2^i + X) \oplus X$ for all $0 \leq i \leq n$ where $X = 1 + SQN^0_{HN}$ is the quantity we seek to infer the n least significant bits of. In a nutshell, the idea of the algorithm consists in analyzing how remainders propagate in $(2^i + X)$ at bit position i and $i + 1$ (in little-endian notation) by looking at δ_i. Considering X and δ_i as arrays of 48 bits in little-endian, we describe the algorithm in Algorithm 1 that, given the δ_i's, infers n bits of X. Note that this algorithm can be executed completely offline on the collected data.

Improving the Attack Under Stronger Threat Model. When the targeted *UE* stays a long time in the attack area or intensely consumes mobile services (triggering a lot of AKA authentication sessions), the attacker has a simpler way to break the confidentiality of SQN. This kind of scenarios are realistic when the attack areas

Data: $\delta_i = (2^i + X) \oplus X$ for $0 \leq i \leq n$ (in little-endian), $n < 48$
Result: Res: n least significant bits of X (in little-endian)
$\mathrm{Res} \leftarrow [0,\ 0,\ \ldots,\ 0]$ `//size` n
for i **from** 0 **to** $n - 1$ **do**
  `//Let's analyze` δ_i `at bit positions` $i,\ i+1$
  $(b_1,\ b_2) \leftarrow (\delta_i[i],\ \delta_i[i+1])$
  if $(b_1,\ b_2) == (1,\ 0)$ **then**
    `//no remainder propagate when` $+2^i$ `to` X
    $\mathrm{Res}[i] \leftarrow 0$
  elif $(b_1,\ b_2) == (1,\ 1)$ **then**
    `//a remainder propagates when` $+2^i$ `to` X
    $\mathrm{Res}[i] \leftarrow 1$
  else `//cannot happen`
    Error
end
return (Res)

Algorithm 1: *SQN* Inference Algorithm

are *e.g.,* offices where targeted *UEs* stay most of the day but expect to be safe when being outside attack areas (*e.g.,* at home).

Essentially, instead of fetching the challenges R_i,*AUTN*$_i$ and injecting the challenges that are accepted by the *UE* (*i.e.,* R_0,*AUTN*$_0$ and then R_{2^j},*AUTN*$_{2^j}$ for $0 \leq j \leq n$), the attacker can let the *UE* attaches to any genuine *SN* and let it receives challenges and completes the AKA sessions. The attacker just passively eavesdrops on the exchanged messages, notably the challenges, and counts the number of successful authentications. However, the attacker still needs to (actively) replay the challenge R_0,*AUTN*$_0$ at appropriate times; more precisely, after the *UE* received the genuine challenge (R_0,*AUTN*$_0$) and then challenges (R_{2^j},*AUTN*$_{2^j}$) for $0 \leq j \leq n$. This variant is far less costly: it only requires passive attacking capabilities and $n + 2$ additional (active) injections.

Variants for Other *SQN* Policies. According to non-normative parts of the specification [1]), *SQN* and its update policy can take different forms. We briefly explain how our attack can be easily adapted for those variants.

SQN can be composed of two components *SQN* = *SEQ*||*IND* where *SEQ* is a 43 bits long integer that counts all past AKA sessions and *IND* is a 5 bits long index that describes the *SN* for which the given *SEQ* is valid. When such a policy is in use, one can use a slightly different variant of our attack: (i) injections of authentication challenges should be done while using the same *SN* identifier towards the *UE* and the same *SN* while fetching authentication tokens, and (ii) the algorithm used to infer

bits should drop the 5 bits of *SQN* corresponding to *IND*. This allows the attacker to break the counter part of *SQN*, namely *SEQ*; leading to the same privacy attacks.

5.2 Fingerprinting Cellular Devices and Subscribers

Based on the weaknesses we identified in LTE device and subscriber management protocols, we uncover two new privacy attacks that fingerprint devices and subscribers in LTE networks. We classify the attacks as: a) Mobile Network Mapping (section 5.2.1) (referred as F1) and b) Subscriber activity monitoring (section 5.2.2) (referred as F2) and are discussed as follows.

5.2.1 Mobile Network Mapping (MNmap) (F1)

MNmap reveals various techniques to fingerprint active devices in a mobile network and intellectually estimate the underlying applications. We start by understanding different UE capabilities in detail and their usage in commercial devices and applications. Next, a reference model is designed using a set of known devices and techniques that can be applied to fingerprint and distinguish unknown devices and applications. Lastly, we use our reference model to perform MNmap attack and discuss the impact and challenges of doing such an attack.

A. Reference Model

The term device-type in our research represents device specifics such as the combination of the maker, model, software, and the application(s) on the device. The manufacturing of cellular-enabled devices involves multiple entities: a baseband vendor producing the modem, a device manufacturer integrating the modem with other components such as sensors or displays, and an application developer providing lightweight firmware or full-stack operating system. Baseband vendors define UE capabilities according to the 3GPP standards [8] And make them adjustable for device manufacturers and application providers according to their specifications and requirements. Due to a large number of optional capabilities (several hundred), each baseband manufacturer may implement a subset of the full capabilities distinctly. Similarly, device and application providers can also adjust the UE capabilities. Based on these distinct implementations, we discovered that it is possible to identify a device-type and its corresponding application.

Each target application requires different UE capabilities. For example, a mobile phone involves telephony capability. A tracking device requires persistent GPS access, while telephony is not always needed. Cars require multiple capabilities: GPS for navigation, V2V for self-driving car [20]. All these capabilities are defined in the modem and are enabled/disabled according to the target application. Thus, there is a direct correlation between a UE capability and a target application.

Device identification or fingerprinting is based on the differential analysis of the capabilities that are obtained from a UE. Initially, we perform dedicated experiments to learn the ground truth information about device-types and create a reference model from it. This reference model is a vast database of capabilities from which we models techniques and fingerprints to identify device-types. We used 40 devices for our experiment, including mobile phones, cars, tablets, routers, USB data sticks, e-bikes, cellular IoT devices like trackers, and coffee machines (detailed list in Table B.1 in Appendix B). Device-types are then systematically identified based on a tree-based model shown in Figure 5.2 consisting of four levels (marked in different colors). The first level identifies the baseband vendor and the model of the device, and the second level differentiates cellular and cellular IoT devices. The third level determines the device's application, and the fourth level identifies the device manufacturer and application provider.

By using our `rogue eNodeB` setup, we acquire both the core network and radio access capabilities from the test devices and analyze them. In particular, UE initiates a registration process with our `rogue eNodeB`, and we extract the capabilities from the *Attach Request* and *UE Capability Information* messages. We then compare the implementation differences of specific capabilities listed in Table B.2 to identify the right baseband vendor and model. Further, we investigate the presence/absence of one or more capabilities listed in tables B.3, B.4 and B.5 in Appendix B to determine the right device level and further deduce the device-type details. We define each of the levels and corresponding identification techniques as follows:

Baseband Vendor Name and Model. We primarily identify the baseband vendor and model of the UE. As the number of active baseband vendors is limited, we can distinguish them using a few implementation differences in the capabilities. We consider the following popular baseband vendors with a significant market share: Qualcomm, Samsung, MediaTek, Intel, and Huawei. We discovered a set of capabilities, as shown in Table B.2, that is (de)activated in each of these basebands and allows us to identify the vendor. For instance, Qualcomm based UEs, by default, do not support the NULL integrity algorithm EIA0 [23]. EIA0 is mainly used for emergency calls, and Qualcomm baseband dynamically activates it, unlike other ven-

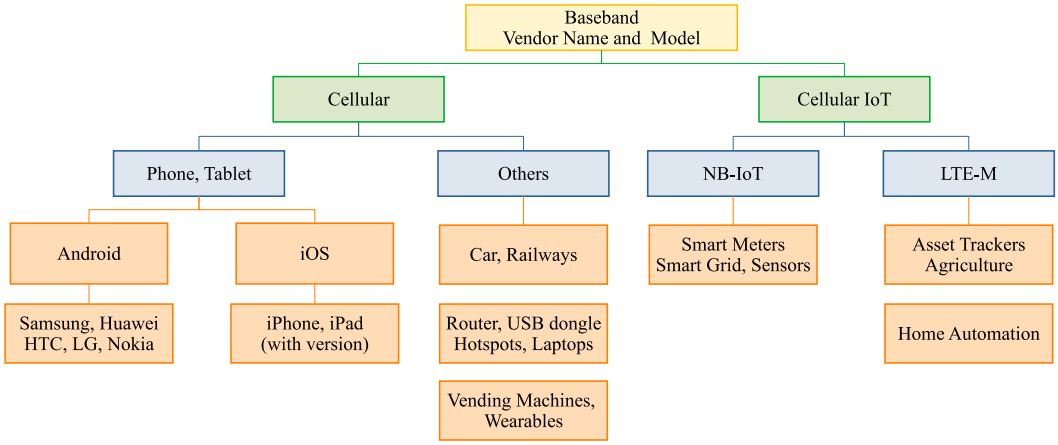

Fig. 5.2.: Device type identification levels

dors. Hence any UE lacking the support for EIA0 can be considered as a Qualcomm baseband. Similarly, Huawei basebands support all the listed capabilities. Further, Samsung, Intel, and MediaTek can be differentiated based on the combination of other capabilities.

Next, every baseband model is designed to support a particular LTE specification release and a corresponding set of capabilities. By referring and comparing a baseband model to our reference model, the model name (or number) of the baseband can be determined. E.g., release 9 specifications support only LTE technology, whereas 10 specifications support LTE-advanced features. Hence in the case of Qualcomm, the former is found in the MDM9615 baseband model and the latter in MDM9625 (or higher) models. Upon revealing the model, the corresponding list of devices using this baseband model can be obtained from various sources on the internet, such as GSMArena [67] and WikiDevi [163]. This information is later used in other levels as assistance to identify the device manufacturer and also the application.

Cellular vs. Cellular IoT. 3GPP defines various UE Categories (Cat) depending on their LTE specifications and the supporting technical capabilities [27], between 0 and 19. Further, NB-IoT and LTE-M are different categories and features defined, especially for IoT applications. These categories do not support voice calling features and instead support power-saving features. As shown in table Table B.3, timers *T3324* and *T3412 ext* are included in *Attach Request* message to indicate power-saving features [32]. Hence when these timers are active, we can accurately decide level two that they are a certain type of cellular IoT device.

Phone vs. Others. The primary use of a mobile phone is to make voice calls. Therefore voice capability is activated by default. In contrast, there are cellular

modems dedicated to data-only purposes without voice calls; hence, we categorize them as “others”. These include data sticks, cars, hotspots, wearables like watches, etc. The device capabilities in table Table B.4 clearly distinguish UEs that are phones from all other UEs that are not phones. Unlike “others”, a phone indicates its *UE Usage Setting*, *Voice Domain Preferences* and *voice codec* support to the network and activates voice calling capabilities. iPhone models can be distinguished based on the specification release and also the UE category, whereas we have a different approach to distinguish various android manufacturers.

A UE fixed in a car requires GPS features to be turned continuously ON. Further, in LTE and 5G networks, UE capabilities indicate V2X or V2V support. When such a capability is detected, it can be referred to as a vehicle. A railway specific modem has unique features that support frequencies dedicated to railways such as GSM-R [69]. Differently, USB dongles and routers (also hotspots) are purely data-oriented and lack any voice codec facilities. These distinct capabilities can distinguish different devices at level 3.

NB-IoT vs. LTE-M. While both NB-IoT and LTE-M are targeting low-powered IoT applications with 10 years of battery life [26, 63], they have different operational aspects. NB-IoT uses different radio channels compared to LTE-M and hence, easily distinguishable from each other. The separation of these two categories assists in identifying the underlying IoT application.

Android vs. iOS. iPhones have been continuously using basebands from either Qualcomm or Intel. Thus, devices using other basebands are not considered as an iOS device. Although Android devices can use Qualcomm or Intel baseband, we noticed many differences between Android and iOS devices with the same baseband as shown in tableTable B.5. *MS assisted GPS* is a capability that we found disabled in all tested iPhone models, but whereas it always enables across android models using Qualcomm and Intel baseband. Note that we did not consider phones with other operating systems such as Windows and Firefox due to their low market share.

Android Device Manufacturers. Based on our analysis, Android device manufacturers have individual preferences in choosing their basebands. Huawei and Samsung basebands are exclusively used in-house. Other manufacturers such as LG, Nokia, HTC use basebands from multiple vendors such as MediaTek, Qualcomm, and Intel. Hence, by referring to the device list [67, 163], it is possible to narrow down the possible options and determine the right phone manufacturer.

Application. Cellular types devices are multi-purpose devices with moderate to high computing capabilities and can be identified based on the above techniques. For example, upon detecting a router, its operating system can be inferred from various internet sources. In contrast, cellular IoT type devices have less computing power and are dedicated to single application usage. LTE-M provides better latency than NB-IoT, making it suitable for mission-critical applications such as those involving emergency data and precision tracking data. A wide range of applications and the appropriate category is defined in [63] as a recommendation to the device manufacturers.

Similarly, the application can be inferred based on the requested timer values. A UE can request lower *T3412* costs, such as 15 seconds or less to save more power. This could be translated into a device or a sensor like smart-meter that only pushes data to a server and do not expect any responses. Differently, a vending machine or asset trackers require up to 1-minute active state depending on the requirements. However, this heavily depends on the settings of the application. Some devices may use the default value supplied by the baseband manufacturer, which may not be optimal for their specific use case.

B. Fingerprinting Attack

The primary goal of this attack is to identify devices on a mobile network by analyzing their capabilities. Since a UE transfers its capabilities to the network without performing authentication [8], an active adversary can acquire these capabilities (both core and radio) by operating a `rogue eNodeB` as described in our setup. Besides, a passive adversary can also learn UE's core network capabilities but not the radio capabilities (provided they are exchanged after RRC security setup). In this section, we perform the attack being an active adversary as we require both core and radio capabilities to perform a granular identification.

We present an experiment with an unknown UE and apply our reference model to determine its device-type. Upon receiving a *TAU Request* message from the UE, we extract the core network capabilities and send a *UE Capability Enquiry* message. The UE responds with a *UE Capability Information* message, and we obtain the radio capabilities from it and release the UE to a legitimate network using a *RRC Release* message. In our experiment, an unknown device was identified to use Intel XMM7480 baseband based on our model, due to its Cat 6 support. It is determined as a phone/tablet since the device has voice support (Table B.4) and reports itself as a voice-centric device. By searching the smartphones and tablets with Intel XMM7480 baseband, we could identify that this is an iPhone 8.

The secondary goal of this attack is to determine potential vulnerabilities applicable to the identified device. Precisely, MNmap can be supplemented with vulnerability information from the external sources such as vulnerability databases from baseband vendors (Huawei [83], Qualcomm [128]), OS developers (Google [61], Apple [39]) and device manufacturers (Samsung [139]) and perform targeted attacks. Further, these device fingerprints can be combined with the permanent identifier IMSI to track subscribers while 5G prohibited the plaintext transmission of IMSI in any situation [25, 18], fingerprinting of a device and user is still possible when the device-type information is unique among the nearby devices.

C. Evaluation and Challenges

While we only consider 5 major baseband manufacturers, our reference model is also expandable to other baseband manufacturers. Identifying the baseband vendor and chipset model is the biggest achievement and can be easily accomplished with the set of fingerprints defined in Appendix B. We evaluate our experiments with 10 other unknown test UEs and determining up to the fourth level. These 10 devices are closely related to the devices registered in our reference model. The MNmap depends on the reference model and publicly available databases to infer the device-type information. Hence a bigger and diverse reference model is required for accurate device-type identification. In some instances, it is also possible to reveal the baseband version and even the operating system version of the device.

Phones, tablets, routers, and automotive devices are easily identified using our reference model, whereas determining the application of cellular IoT devices is challenging due to its limited set of capabilities and similarities among several applications. Another challenge is to determine the application OS version since the baseband model, and mobile OS versions are not linked and not synchronously updated. Besides, in certain UEs (especially phones), a USIM card can activate/deactivate specific capabilities. E.g., frequency bands are enabled and disabled according to particular settings by the network operator. Hence, identification is affected by the USIM card setting and should be considered during MNmap attack.

5.2.2 Subscriber Activity Monitoring (F2)

We reveal a new privacy attack against all variants of the AKA protocol (including 5G AKA and EAP variants) that breaches subscribers' privacy more severely than known location privacy attacks [40, 73] do. Our attack exploits a new logical vulnerability, as discussed earlier. We show that partly learning SQN leads to a new class of

privacy attacks (*i.e., activity monitoring attacks*): an active attacker can leverage fake base stations and our attack to learn information about targeted subscribers' mobile service consumption, even when subscribers move away from the attack area (*e.g.,* range of a fake base station).

This is in stark contrast to location attacks that do not reveal service consumption and requires the targeted subscribers to stay in attack areas. Less importantly, we show that our logical vulnerability also yields a new location attack. We demonstrate the feasibility of our attack using widely available and low-cost setup on commercial 4G networks in several European countries. Our attack affects all 3G and 4G devices currently deployed all over the world and future 5G devices (according to the specification [18]).

In a nutshell, the attacker needs to conduct the previously described attack when targeted subscribers are in the attack area, thereby learning n significant bits of *SQN* at different times $t_1, t_2, \ldots$. The attacker can then relate this information to the number of AKA sessions subscribers have made between times $t_1, t_2, \ldots$. Next, the attacker can relate the number of AKA sessions some *UE* has performed in a given period of time to its typical service consumption during that period. Therefore, the attacker learns the typical service consumption of targeted subscribers between times $t_1, t_2, \ldots$ even if such subscribers escape the attack area most of the time (*i.e.,* in between times t_i).

A. Relating SQN Increases to Activity Patterns

We first need to learn the value that is added to *SQN* after each successful authentication. The conclusion of our practical investigations is that this value is 1 for all tested operators. This value is needed because equal differences of *SQN* could be resulted by different operations: if the victim *SQN* had been increased by 20, it could be the result of either 4 increases of 5 (4 authentications) or 2 increases of 10 (2 authentications).

We found how much *SQN* is increasing upon authentication for several operators by running the algorithm algo() for several values of the increment and keeping the value yielding no Error (see Algorithm 1). We stress that this has to be done just once for or a given operator.

Next, in order to relate information about the number of AKA sessions of a victim with the victim's activity, we have to exploit the fixed authentication policies discussed in Section 5.2.2 (*i.e.,* which user's activities trigger an authentication and thus an AKA

session). Because of the different operator configurations, authentication may or may not happen on each *SN* network attach, call or reception/sending of SMSs. As a result, we also analyzed how frequently authentication is performed by analyzing signaling messages during repeated attach procedure (by calling or sending SMSs). We found that there are little variations in authentication frequency among operators but for most of them, an authentication was required for each outgoing call and sent SMS). Despite those variations, one can easily infer the fixed policy for some operator, once for all, by inspecting signaling messages *e.g.,* on her own phone.

B. Examples of Practical Scenarios

We now illustrate the potential real-life impacts of our activity monitoring attack with two practical scenarios.

Spying on embassy officials, journalists, or any high-value target. Assuming an adversary having a fake base station near an embassy, he can learn the officials' activity not only when they are at the office during working hours, but also when they are not, including during evenings and nights (*e.g.,* at home) or during business trips. Therefore, such an attacker may learn if targets use different SIMs cards for private use (no activity at home), if some specific time periods (*e.g.,* one evening and night) were specifically busy (a lot of calls or SMSs were made yielding a big rise of SQN), if one is using his work phone at home, if a phone was switched off for certain time periods or trips (possibly indicating multiple SIMs usage), *etc.*

Better ads targeting. Consider for instance a shop that is willing to know more about its customers (*e.g.,* for improving ads targeting) using fake base stations. This kind of scenario has already been reported [111] (using Wi-Fi capabilities of smartphones) and exploited [45] in real shops. Our attack causes a new threat in that context since it leaks to the shop typical customers' mobile consumption during time periods *between* customers' visit (while they escape the attack area).

C. Deriving Location Attacks

Using variants of our attack, one could mount location attacks (*i.e.,* inferring if some targeted *UE* is in some physical area) even if the *leak of identity* (currently enabling IMSI-catchers attacks) and the *traceability based on failure messages* were fixed.

More precisely, we first assume that the identity request phase would be well-protected using *e.g.,* encryption (as done in 5G, phase 1 [18]). Second, we assume

that the two failure cases (MAC or freshness failure) would be merged (AUTS message is also sent out in case of MAC failure, the network being able to infer the reason of the failure) to address the aforementioned known flaw. Under those assumptions, to the best of our knowledge, there is no known attack that could break subscribers' privacy. However, either of the two following variants of our attack still allows an active adversary to perform location attacks[1].

First, if an attacker knows a value $CONC_0$ of some targeted UE_0 and obtains a value $CONC$ from some unknown UE (this can be easily obtained by replaying a genuine challenge), then he can infer if the unknown UE is UE_0 with very high probability by inspecting how large is $CONC_0 \oplus CONC$, interpreted as an integer. Indeed, when both *UEs* do not match then $CONC_0 \oplus CONC = (SQN_{UE_0} \oplus AK^*_0) \oplus (SQN'_{UE_?} \oplus AK^*_?)$ (where $AK^*_? \neq AK^*_0$; see Section 5.1.3) which is a 48-bits random-looking value. By contrast, when they do match, then $CONC_0 \oplus CONC = SQN_{UE_0} \oplus SQN'_{UE_0}$ which is very likely a small value (we never observed more than 10 bits-values). We stress that this can be done even when the *SQN* values of different *UEs* are close to each other or even equal. E.g., if Alice and Bob both have the same *SQN*, the attacker will still be able to locate Alice later on and distinguish her from Bob.

Second, by learning sufficiently many bits of some targeted *UE*, an active attacker will be able to track this *UE* with reasonable probability by keeping track of the SQN values he may repeatedly learn (recall that SQNs are 48-bits long so they almost injectively identify *UEs* even taking into account the fact that they evolve). Obviously, the practicality of this second attack heavily depends on the number of bits one can infer, closeness of *SQNs* between different *UEs*, the frequency at which the target visits the attack area, and the speed at which the target's *SQN* rises.

We consider those location attacks as potential threats for the upcoming 5G, phase 2 that may address previous flaws but not necessarily this new attack.

D. Proof of Concept Attack

In this section, we show how to conduct our attacks in practice on 4G networks using the low-cost experimental testbed. We then explain practical aspects which make our attack easily feasible (*e.g.,* issues in different operator's network and security configurations). First, we collect the victim's authentication challenges using a `rogue UE` and next, operate a `rogue eNodeB` to inject legitimate AKA related signaling messages to the victim UE.

[1]Note that our activity monitoring attack can also be exploited under those circumstances.

Obtaining authentication challenges. We used our rogue UE and configured it with the target's IMSI for obtaining authentication challenges. Essentially, the rogue UE tries to impersonate the target's USIM. When doing so, each session fails because our UE does not know the target's secret key K (so it cannot compute the appropriate RES) but, before the failure, we obtain a new, genuine authentication challenge that is intended for the target's USIM. We were able to fetch authentication tokens using the USRP at a surprisingly high speed (see discussion later) but, if for some reason, a network recognizes the USRP as a fake smartphone, we can still use genuine phones with programmable [153] USIM cards (ca. 80€).

AKA configurations in commercial networks. Before explaining our attack, we report on our investigation on AKA related security configurations of 4G networks which make our attack easier to perform. We selected several major European 4G operators including three German, three Austrian, two French, and one Swiss operators.

We were successfully able to collect authentication challenges intended for the targeted USIM at any moment for any subscriber in the world. Note that to achieve this step, the attacker only needs to know the IMSI (or any temporary or encrypted identity) of that particular victim's USIM. If the attacker knows the subscriber's mobile phone number, he can perform HLR Lookup attacks [51] to learn victim's IMSI. The previous work [142] also demonstrates how to find IMSI and GUTI of the targeted victim by knowing the mobile phone number or social identities such as email, Facebook and Twitter. Based on the data collected from our experiments, we studied the following parameters of the operator's 4G networks.

We stress that there is no need to learn more information (*e.g.,* private key K_{IMSI}) about the targeted USIM). We found that most operators allowed to fetch authentication challenges without a rate limit. Using our first setup using *srsUE*, we were able to fetch fresh, unused authentication challenges consecutively at the speed of 1 per second. Using our second setup involving a smartphone, we were able to fetch more than 30 challenges in less than 10 minutes. We expect a setup based on multiple rogue base stations to achieve much better performance.

Executing Attack. We operate a `rogue eNodeB` to inject messages and eavesdrop on replied messages. We use this method to fetch AUTS messages that a USIM sends as part of the AKA protocol. Prior to this, the attack requires obtaining a larger number of authentication challenges of victim's USIM. In our attack, we did not observe any countermeasure preventing us to fetch a large amount of them. The more consecutive challenges one fetches, the more is the number of bits he can infer

from the *SQN* of the victim. Then, using a rogue base station, the attacker is then able to inject parts of those challenges and store replied *AUTS*. For instance, we were able to request 1025 authentication challenges and collected 12 *AUTS* from 24 injections of AKA messages. Using our generic *SQN* inference algorithm, those 12 *AUTS* messages were enough to infer at least 10 bits of *SQN* (the least significant ones), sometimes more. Obviously, an attacker with greater capabilities and more elaborate setups (notably multiple rogue base stations for fetching challenges; which turns out to be the bottleneck) could infer more bits.

Attacks Feasibility and Impact. We now describe the feasibility of our subscriber activity monitoring attacks against commercially deployed 4G devices. Further, we discuss possibilities of extending the coverage range of the USRP device. The AKA protocol vulnerability we found is part of the 3GPP specifications and does not rely on implementation issues in 4G/3G devices. In fact, the affected AKA protocol is implemented in the USIM and not in the baseband OS of devices. Thus, any 3G/4G device deployed worldwide having active USIM card is affected by our attacks. For our investigations, we selected prepaid USIM cards of few leading cellular operators. We collected and stored unused authentication challenges of related USIM cards as described before. Then we successfully verified that these USIM cards were vulnerable to our attack. As mentioned earlier about feasibility in 5G networks, if no dedicated mechanism for mitigating our attack is implemented, 5G devices will also suffer from our attack.

5.3 DoS Attacks on LTE Subscribers

In this section, we demonstrate how an attacker can exploit LTE subscriber and device management protocols to cause persistent DoS to LTE subscribers. In particular, the device capabilities and access control protocols are manipulated by the attacker to perform either a complete or partial denial of network availability. We categorize them as a) Denial of network availability and b) Downgrading of network service. The former targets to completely forbid the UE from connecting to a specific or all network technologies whereas the second targets to degrade the service(s) received over the LTE network. For referencing the attacks, we label them as D1, D2, D3, D4, and D5. Finally, we discuss their impact on LTE subscribers and operator services.

5.3.1 Denial of Network Availability (D1 - D2)

We present two types of DoS attacks that refrain a subscriber from accessing the mobile network. This is achieved by an active attacker that injects NAS protocol messages to the subscriber UE and exploit the weaknesses discussed in section 5.1. We discuss the attacks below:

A. Downgrade to non-LTE network services (D1)

We identify a vulnerability in the LTE specification, which enables the following DoS attacks D1. We exploit the fact that certain *"TAU Reject*" messages sent from the network are accepted by UEs without any integrity protection. In particular, there is no need for mutual authentication and security contexts between the UE and network for accepting such reject messages. Note that, the attacker does not need any security keys to send *"TAU Reject*" messages. Hence, the attacks can be targeted towards any LTE subscribers within the range of the `rogue eNodeB`. Similar types of attacks are also possible with *"Service Reject/ Attach Reject*" messages.

As shown in Figure 5.3, the UE sends *"TAU Request*" message to attacker's `rogue eNodeB`. Note that as the UE is attached to the real network, this message can be integrity protected using the existing NAS security context. However, according to LTE specification [16](section 4.4.5), this message is not encrypted. As a result, `rogue eNodeB` decodes it and responds with a *"TAU Reject*" message. The attacker includes EMM cause number 7 *"LTE services not allowed*" into this message. As no integrity protection is required, the victim's UE accepts the message. The UE proceeds to act on the indicated rejection cause by deleting all existing EPS contexts associated with the earlier (real) network.

As a result, UE updates its status to *"EU3 ROAMING NOT ALLOWED*"[2] and considers the USIM and hence the UE as invalid for LTE services until it is rebooted, or USIM is re-inserted. Further, UE does not search for or attach to legitimate LTE networks even if they are available in that area, causing a denial of service. However, if supported, the UE searches for GSM or 3G networks in the same area to gain network services. By downgrading subscribers, an attacker could attempt to launch known 2G or 3G attacks, besides the loss of LTE services.

[2]It means that last *TAU* procedure was correctly performed, but reply from the MME was negative due to roaming or subscription restrictions.

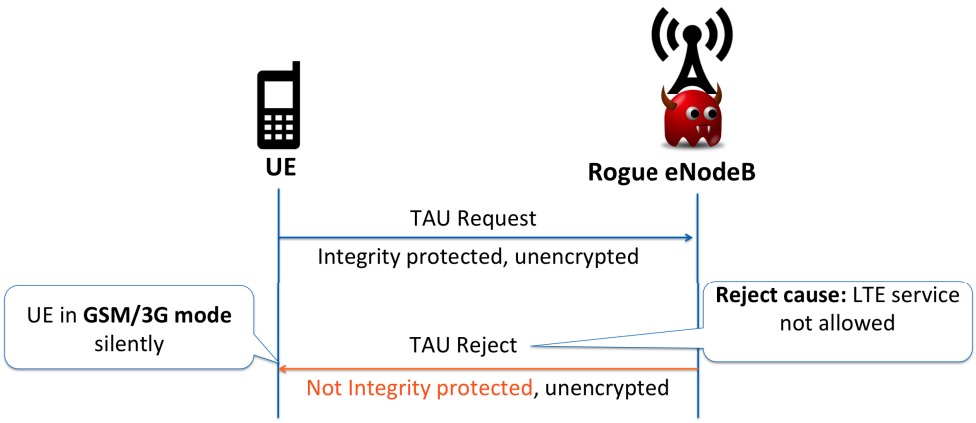

Fig. 5.3.: DoS attack - denying LTE network services (D1)

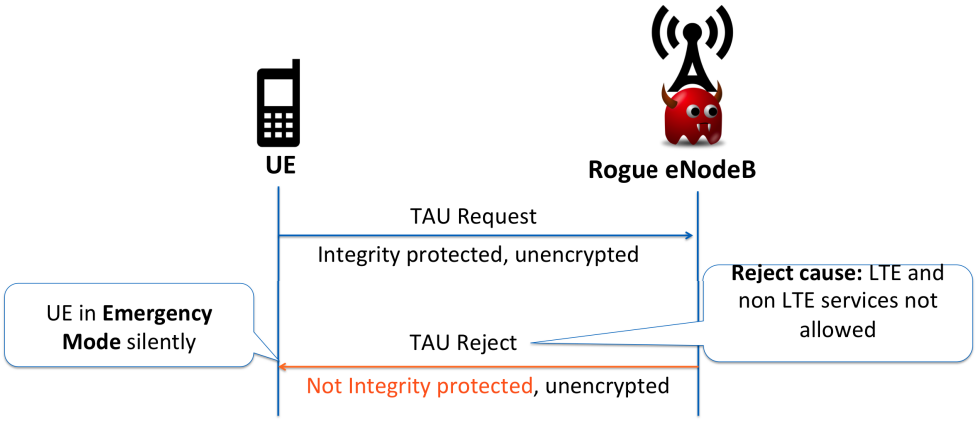

Fig. 5.4.: DoS attack - denying all mobile network services (D2)

B. Denying all network services (D2)

The operation D2 is similar to D1, but the result is different. The UE initiates TAU request procedure and `rogue eNodeB` responds with a TAU Reject the message with the cause number 8, which is *"LTE and non-LTE services not allowed*". After receiving this message, the UE sets LTE status to *"EU3 ROAMING NOT ALLOWED*" and considers USIM invalid for the network until it is rebooted or USIM is re-inserted. Further, it enters the state EMM-DEREGISTERED: UE's location is unknown to the MME and is not reachable for any mobile services. As a result, UE does not attempt to attach to LTE, GSM, or 3G networks for standard services even if networks are available. The UE remains in the EMM-DEREGISTERED state; also it moves to a new TA or even to a new city, thereby causing a persistent denial of service. Signaling messages exchanged between the UE and the `rogue eNodeB` are shown in Figure 5.4.

5.3.2 Downgrading Network Services (D3 - D5)

We present a set of DoS attacks that manipulates and hijacks a LTE registration process. This is achieved by a MITM that interferes in the NAS and RRC protocol transactions and exploit the weaknesses discussed in section 5.1. We present three types of DoS attacks below:

A. Blocking Telephony services (D3)

In this attack, the active attacker modifies messages exchanged between the eNodeB and UE by using operating a MITM `relay`. The UE initiates a *"Attach Request*" message to the eNodeB, and the attacker intercepts this message. The message contains *"Voice domain preference and UE's usage setting*", which informs the network about UE's voice calling capabilities. The attacker removes these capabilities from this unprotected message and adds *"Additional update type - SMS only*" before forwarding it to the network. The network accepts this message and executes the AKA protocol with the UE to complete the $Attach$ procedure. However, at this step, the MME configures UE's profile with the received (modified) capabilities, thereby allowing only SMS and data services.

When there is an incoming call for UE, the MME rejects it and informs the cause to the subscriber who is calling. On the other hand, if UE tries to make an outgoing voice call, the network rejects this request and informs the cause. This is an example of a bidding-down attack. The denial is persistent since the attack is valid even after the attacker has moved away. However, the user can recover from the attack by restarting the UE or moving to another TA. 3GPP specifications do indeed mention a timer (T3245) that a UE can use to recover from EMM DISCONNECTED state [14]. However, the use of this timer is optional (none of the devices we tested implement this timer). The default timer value (24-48 hours) is too large in the case of DoS attacks.

B. Radio Service Hijacking (D4)

A bidding-down attack performed on a UE by hijacking its radio capabilities shown in Table 5.1. A UE communicates its radio access capabilities [8] with the eNodeB and indicates its support for specific radio operations. An eNodeB needs to respect the received UE radio access capabilities when configuring and scheduling data/signaling for the UE [27, 82]. They are essential to guarantee the subscriber with the right level of service according to the device features. For instance, *UE Category* is used to

set the number of bits allocated by the eNodeB over the radio channels for a UE in both downlink and uplink transmissions [27]. The higher the category, the higher the number of bits allocated. This directly translates to the data rate of the UE over the air-interface. For instance, theoretically, a Cat 6 UE is entitled to receive a maximum of 300 Mbps speed on the downlink with CA and MIMO features support, whereas a Cat 1 UE has a peak of 10 Mbps. Similarly, a UE can make voice calls directly over the LTE network without falling back to the GSM network with VoLTE capabilities.

Tab. 5.1.: LTE Radio Access Capabilities

Capability	Usage
UE Category	Defines data rate over radio
CA and MIMO	boosts capacity of network and downlink data rate
Band	radio frequencies supported by UE
VoLTE	IP based voice calling on LTE

We perform a MitM attack using our experimental setup to hijack the radio access capabilities of a UE during its registration procedure. The fact that mobile network operators configured their eNodeB to request UE capabilities before the RRC security setup allows a MitM adversary to alter the *UE Capability Information* sent by the UE. To exploit this vulnerability on a commercial network, we use an iPhone 8 as a victim UE in our experiment. It is a Cat 12 device and houses an Intel XMM7480 baseband and can boost speeds up to 600 Mbps and further also support CA, MIMO, and several LTE bands. The flow of the attack is pictured in Figure 5.5. To trigger the attack, our `relay` is configured with a TAC that is different from the iPhone 8's current registration area. This will lure it to initiate a TAU procedure, which is rejected by the relay with a *TAU Reject* message. As a result, this will delete the current security context and other temporary identities in the iPhone 8 and initiate a new registration procedure by sending a *Attach Request* message to our `relay`.

We forward this message to the legitimate network using our rogue UE segment and allow the iPhone 8 to finish the NAS security setup successfully. Since this is a new registration and not a TAU procedure, MME requests the eNodeB to acquire UE capabilities. Our `relay` forwards the *UE Capability Enquiry* message received from legitimate eNodeB to the iPhone 8 and retrieves the capabilities in the *UE capability Information* message in a plain-text format.

Upon receiving them, we alter the capabilities in the following way: UE Category is changed from Cat 12 to Cat 1, CA, and MIMO are disabled, VoLTE required capabilities are disabled. Also, all the supported bands are disabled except for the current operational band. Next, we forward the modified *UE Capability Information* message

to the legitimate network and allow the iPhone 8 to successfully establish RRC security and complete the registration procedure with *Attach Accept* being delivered to iPhone 8. Subsequently, we release the UE to the legitimate network using a *RRC release* message. Note that eNodeB forwards these (modified) capabilities to MME, which are then stored for future transactions, i.e., when UE reconnects to the eNodeB to send/receive data, the capabilities are transferred from MME to eNodeB without repeating the UE capability transaction procedure.

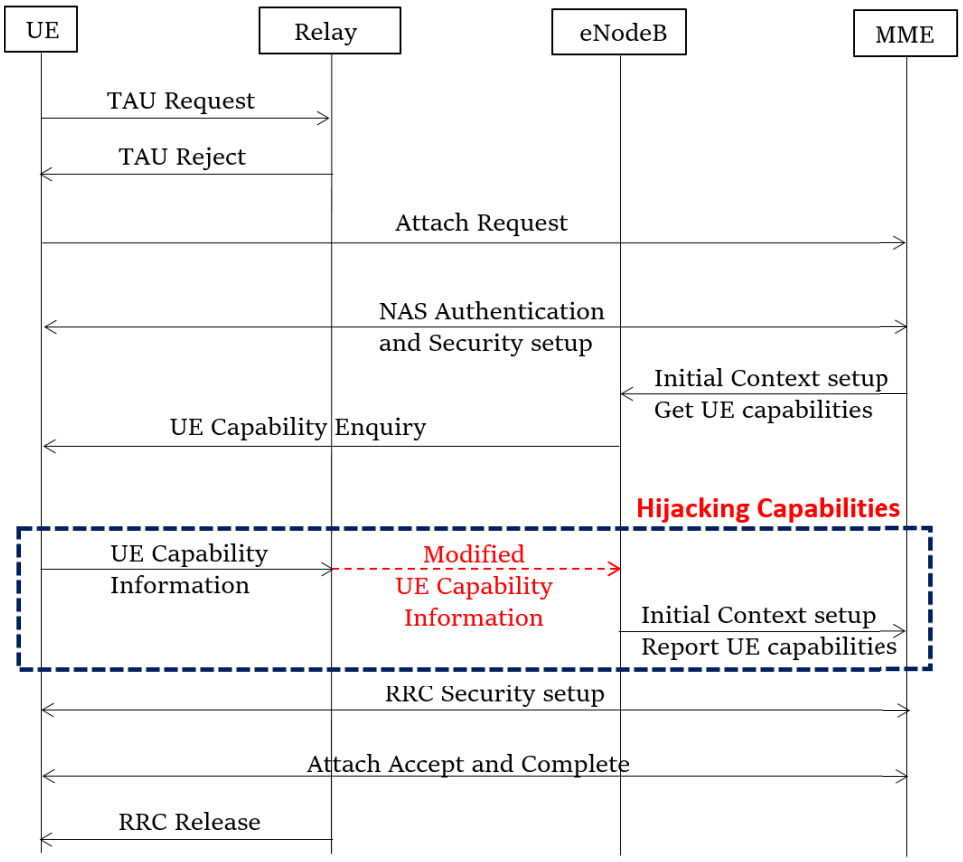

Fig. 5.5.: MITM Capability Hijacking Attack

Hereafter when the iPhone 8 connects to any legitimate eNodeB, it is treated as a Cat 1 device and receives a downlink data rate according to what a Cat 1 device is entitled to receive [27]. Thus the iPhone 8's speed and quality of service are downgraded after this attack. Further, during a voice call operation, due to lack of 4G band support, iPhone 8 is handed over to a 3G base station for call continuity. As a result, the UE will lose access to certain services and also cannot receive the elite QoS and data rate as initially allocated to the subscriber (based on the USIM data plan). We discuss more on our experiments and evaluation with different UEs in the next subsection.

Experiments & Evaluation. In normal conditions, the iPhone 8 offers a data rate (with an elite USIM plan) of 27 Mbps on the downlink. Under the attack, the data rate of the iPhone 8 as measured using Speedtest [147] reduced to 3.7 Mbps. We tested this on two commercial networks and discovered that the maximum speed we received is 5 Mbps. We repeated the experiments with other Gigabyte LTE Cat 16 devices that can boost up to 1 Gbps speeds: a Nighthawk M1 Mobile router [112] and Samsung Galaxy S8 phone. During our tests, although a Cat 16 device supports a theoretical downlink speed of 1 Gbps, we observe 35 to 38 Mbps in practice during low-traffic hours (after 21:00). However, after the attack, the downlink speed is reduced to 2.9 Mbps.

Differently, in peak hours (10:00), the speed is further reduced to 1 Mbps. Although our test SIM is entitled to receive a high quality of service and data rate, the bottleneck persists at the radio layer. Hence, when a UE's radio cannot support higher speed, having an elite subscriber profile is useless. As per the standard [8] UE capabilities can be requested without establishing security and are reflected in the operator's network configurations.

Furthermore, we recorded registration procedure traces of 30 network operators from 20 countries worldwide. We discovered that 20 out of 30 operators are affected by the vulnerability V2, i.e., UE capabilities are requested before RRC security. Hence, an adversary can perform a MitM attack on these networks and downgrade subscriber's services. However, the remaining 10 networks perform RRC security before the UE capability transaction procedure, i.e., the capabilities are transferred in an encrypted and integrity protected message. As a result, any MitM operation will be detected on the eNodeB and aborted.

We also observed that the majority of the networks do not request UE capabilities during periodic TAU or standard TAU procedures to preserve radio resources because the size of UE capabilities accounts for 8188 octets [13] and is the most extended radio message. Further, our experiments with a UE that is registered and roaming inside a city, the network did not request UE capabilities for a week, which means that the MME retained UE capabilities for several days. Besides, we also observed that the network request UE capabilities whenever UE switches to the connected mode and has some data to transmit. However, the network request only the 3G related capabilities of the UE but not the LTE capabilities [6]. Hence, the UE's LTE capabilities are retained at the network for a more extended period, and also the UE remains affected even if the attacker deactivates the `relay`.

C. Battery Draining on NB-IoT Devices (D5)

We drain the battery of low-powered NB-IoT devices by being a MitM on the LTE air-interface. To demonstrate this attack, we mount our NB-IoT testbed as a MitM (`relay`) and Quectel BC68 Evaluation Kit [129] (referred to as BC68 hereafter) as a victim UE. As BC68 is a development board, we have access to its diagnostic ports and can monitor its LTE signaling messages and internal activity logs. In the attack, our `relay` modifies the contents of the *Attach Request* message as shown in Figure 5.6. In specific, the `relay` is configured in such a way that it lures the BC68 to trigger a TAU procedure. Upon receiving a *TAU Request* message, our `relay` acknowledges it with *TAU reject* message, which causes the BC68 to delete its previously-stored context and temporary identifiers and start a new registration by sending *Attach Request* message to our `relay`.

Subsequently, our `relay` removes the *T3324* from the message and forwards it to the legitimate network without modifying any other contents. Further, as overseen by the `relay`, both legitimate MME and BC68 perform authentication and establish NAS security. Finally, a *Attach Accept* message is delivered to BC68 and is released to the legitimate network. Note that the *Attach Accept* message does not contain *T3324* since, the MME did not receive it in the *Attach Request* message. Thus, BC68 cannot activate PSM and does not power OFF. Instead, it decodes broadcast messages from the eNodeB and performs cell measurement activities leading to power consumption. Besides, the network assigns *T3412 ext* to BC68 with a value of 310 hours, which indicates that it should perform the next TAU procedure after approximately 13 days.

During our experiments, in a scenario without the attacker, T3324 is configured to 30 seconds and T3412 ext to 13 days. Thus BC68 enters into PSM 30 seconds after it completes registration and performs a periodic TAU after 13 days. But, under the influence of the attack, UE is always ON for 13 days and performs periodic TAU after 13 days. We measured the current and power consumption of BC68 for several days with and without the attacker and plotted in the Figure 5.7. The initial registration with the network causes the initial peak of current drawn in both cases. Without PSM, BC68 performs power measurements of neighboring cells that consume power. This is reflected in constant fluctuations in the current consumption. In contrast, when PSM is active, the baseband is OFF and consumes almost negligible current.

The 3GPP [26] promises 10+ years of battery life for NB-IoT devices when powered with a 5Wh battery. When we extrapolate our results for 5 Wh battery (assuming no losses), with PSM, BC68 consumed 0.65 mA of average current, making 1538 hours

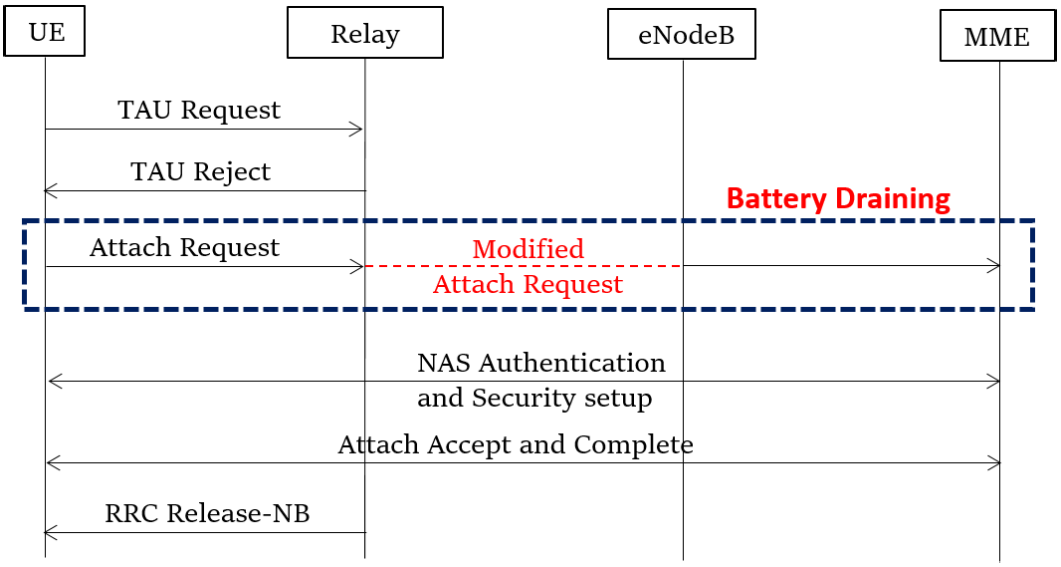

Fig. 5.6.: MitM Power drain attack on NB-IoT devices

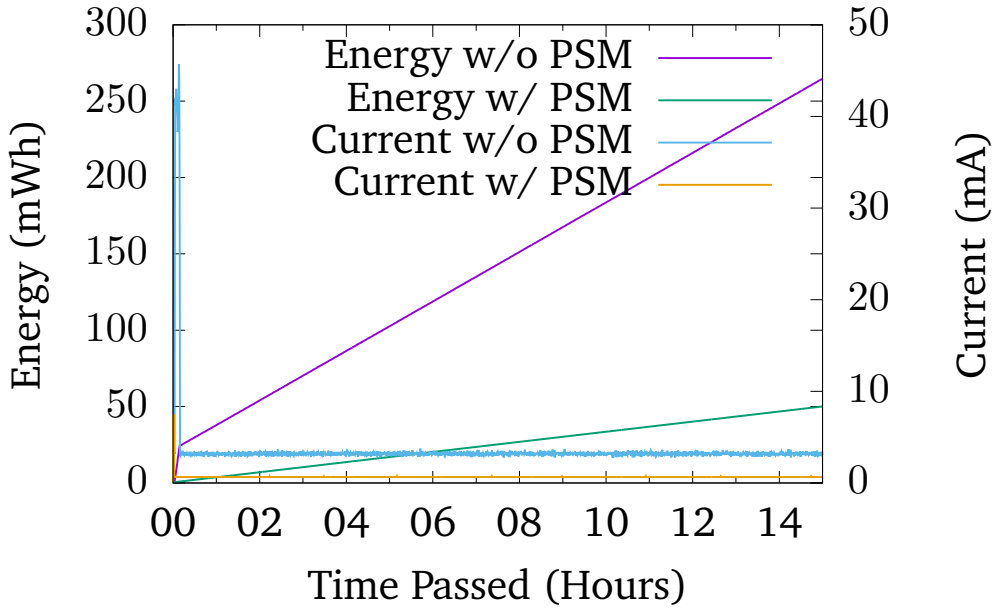

Fig. 5.7.: Current and power consumption of BC68 with and without PSM

(64 days) to draw the full power. In contrast, under the attack, BC68 consumed 3 mA of average current with 5 V input, making 333 hours (13 days) to draw the full power. Hence, a power drain attack reduces battery life by a factor of 5. Note that the total battery life decrease depends on other factors, such as sensors attached to it and how often the communication is performed. In our experiments with BC68, no sensors are attached, and no messages were exchanged, and the baseband explicitly uses all the current.

The attack persists even when the attacker turns off the `relay` and holds until the *T3412* or (*T3412 ext*) expires in the UE. In our experiments, we observed that specific networks implement 10 to 15 days as a periodic TAU timer. It can massively vary depending on the subscription of the SIM, IoT application, and configuration of the operator. To recover from the attack, the UE should reconnect to the network and perform a registration procedure (or TAU) in the adversary's absence.

5.3.3 Feasibility and Impact

The vulnerabilities we discovered are present in the 3GPP NAS and RRC protocols and are mainly exploited during the LTE registration procedure, which is vital for any UE to receive appropriate access to the data plane. Unlike the LTE jamming, DoS attacks described in [90], our attacks are against UEs in a particular area instead of against LTE networks. A successful attack would deny the target UE from utilizing network services. Typically, the UE remains in a non-service state for a while, even if the attacker shuts down his `rogue eNodeB` or moves away from the attacking area. Consequently, this attack is more severe than other types of DoS attacks (for example, jamming and RACH flood [120] that are difficult to prevent). The attack is silent since neither the UE nor the eNodeB can detect the modification of the capabilities. It is also persistent because UE capabilities received during the registration procedure are stored at the MME for a configured period (until UE is turned off as observed). During this period, the altered capabilities are used to set the data rate and services for the UE. Impact of these attacks are as follows:

We further investigated how UEs recover from DoS attacks. We found out that all UEs recover after rebooting or re-inserting the USIM. Additionally, UEs having baseband from most vendors can recover by toggling the flight mode.

- Subscriber's UE may not alert the user about the unavailability of legitimate services. However, depending on the alert notification capabilities provided by the application layer of various mobile operating systems installed on the UE,

the subscriber could be notified of limited services or no network connectivity status. We noticed that there is no standard approach across different mobile operating systems to indicate the type of active network mode (e.g., 2G/GSM, 3G, LTE) to the user.

- Subscribers will not be able to receive or make any calls and data connections. Subscribers will not be able to receive or make regular calls and data connections. Hence, a significant loss is incurred to both network operators and their subscribers. Network operators are not able to offer services since subscribers are unavailable technically, and no billing would occur.

- UE can still make emergency calls. However, emergency calls are not possible when UE is attached to a `rogue eNodeB`.

- LTE-capable M2M devices that are not attended by technicians daily could be blocked out from network services for a long time. This is because M2M devices need to be rebooted, or USIM needs to re-inserted to recover from the attacks.

- Attack D5 mainly targets the low-powered NB-IoT devices, and hence all manufacturers implementing the LTE release 12 standards are affected by this vulnerability.

- D5 can cause a severe decline in the data rate. Further, voice calls will be denied to UE, and in some instances, UE has to switch to 2G/3G networks to perform calls and also handovers. A downgrade to lower generations of the network will make UE vulnerable to more attacks. V2V services can be blocked to UE entirely, or they are offered with low QoS and high latencies.

- UE should be restarted and re-registered to recover from the attack. A subscriber affected by the attack would potentially launch a complaint with the customer service or switch to another operator.

5.4 Trade-off Analysis

In this section, we explain the background behind the vulnerabilities in LTE device and subscriber management protocols by considering various trade-offs between security and performance, availability, and cost. We show that the equilibrium points

in the trade-offs had shifted today compared to where they were when the LTE security architecture was being designed. Table 1.1 summarizes our analysis.

5.4.1 Possible trade-offs and Discussion

Security vs. Performance. We observed that UEs are required to reboot or re-insert USIM after DoS attacks to regain network services. This behavior, exhibited by all LTE devices we tested, is according to the LTE specification. Since the network denies services for valid reject causes described in [16], the UE restricts itself from re-initiating LTE (or any mobile network) *Attach* procedure to conserve battery power. Besides, frequent unsuccessful *Attach* requests from UEs would increase the signaling load on the network. These are the reasons why the LTE specification requires the UE to reboot or re-insert USIM to recover from reject messages. This preference for performance over security leaves LTE subscribers vulnerable to the DoS attacks (D1 & D2).

As another example, during *Attach*, UE's security capabilities are sent back to it for confirmation after security activation to protect against bidding down attacks. This is an application of the well-known 'matching history' principle used in security protocol design [48]. However, UE's network capabilities are not protected similarly, enabling a different type of bidding down attack (D3). The reason for not applying the matching history principle to all negotiated parameters, as discussed in section 5.1, indicates another trade-off where added security has not outweighed performance loss due to the full application of the matching history principle. To apply the matching history principle to all parameters would have required the inclusion of a cryptographic hash of all the parameters, instead of the parameters themselves. However, confirming only the security information capabilities, which take up much less space (only a few bits) compared to a full cryptographic hash, minimizes the overhead in signaling.

Similarly, in current 3GPP standards, it has been a design choice [22] to allow UEs RRC capabilities to be sent unprotected, i.e., before AS security activation. The reason for allowing that is to enable the network to do early optimization for better service/connectivity. Hence this is a conscious exception made in the design to trade security to achieve better performance on the network.

Trade-offs in AKA protocol design.
We demonstrated how an attacker with low-cost tools fetch unused authentication challenges (R, $AUTN$) of any 4G subscriber in the world from any network. We now explain the AKA protocol design choice of the authentication method and trade-offs

responsible for enabling access to *R* and *AUTN*). The AKA is a challenge-response type of protocol and utilizes a symmetric encryption-based authentication mechanism. We believe that the reason for choosing symmetric encryption stems from three trade-offs.

Security vs. Cost. High cost of introducing a PKI into the 3G/4G systems and asymmetric encryption mechanism in USIM, paves the way for choosing symmetric encryption-based authentication technique. Due to this high cost, the 3GPP designers were limited in the previous 3G and 4G networks; however, PKI is introduced in 5G, only for protecting identities [18]. Note that authentication in 5G, excluding identification, is still based on symmetric cryptography.

Security vs. Availability. *i.e.,* Use of symmetric key avoids the risk of shutting down legitimate subscribers during a case of the network fail or crash [55]. For example, if the *SN* (in particular MME) software crashes, the temporary identity of a subscriber can not be recognized. In such a case, the network needs to request a permanent identity from the subscriber.

Privacy vs. Network Efficiency. The AKA is a *one round-trip* authentication protocol; *i.e.,*, only two exchanged messages are needed to establish mutual authentication, after identification. The chosen mechanism to achieve mutual authentication with only two exchanged messages is the synchronized *SQN*.

Allowing the *UEs* to generate a random number could have enabled different authentication methods. Meanwhile, in the year 2000 (when 3G AKA was designed), *UE*'s computational resources were limited. With three exchanged messages, the protocol would not need this synchronized state, and this additional message exchange could have enhanced privacy. However, this additional message exchange would also negatively impact the network efficiency, notably, because it would always require a message exchange between the *HN* and the *SN* as well [55].

The above trade-offs force the network to send *R* and *AUTN* to perform a round of challenge-response for the authentication of a subscriber's temporary or permanent identities. This allows an attacker to impersonate a subscriber's identity to fetch unlimited *R* and *AUTN* challenges from any network. One reason why an attacker can fetch those challenges of any subscriber from any network is the trust between the *SNs* and the *HNs*. Indeed, in 3G and 4G architectures, the *HN* and *SN* trust each other due to roaming agreements.

Further, such illegitimate requests are difficult to filter out from the legitimate ones due to the risk of shutting down real subscribers from accessing the network. One potential solution is to rate limit (based on time or numbers) authentication requests per subscriber; however, an attacker could learn such kind of rate limit by merely testing the network. We found that one of the operators is implementing the rate limit of 3 consecutive failures. Moreover, an attacker could bypass this countermeasure by requesting authentications challenges from different *SNs*.

5.5 Potential Mitigations

Protection against DoS. The specification vulnerabilities responsible for DoS attacks based on *TAU* procedure (D1 and D2) can be fixed without changes in the protocol itself. The 3GPP SA3 group may propose a new mechanism based on a counter or timer value to recover from DoS attacks. If the UE is detached from the network for a specific duration as a result of a *TAU* reject messages, it should reset the configuration settings in the USIM or baseband to re-attach itself with the network without bothering the user, i.e., without having to reboot or require re-insertion of USIM. If there is an infrastructure to support the distribution of operator public keys, *TAU* reject messages could be signed by the network and verified by UEs.

Next, we discuss protection against DoS stemming from bidding down attacks (D3 and D5). During a *Attach* procedure, the UE's network and security capabilities are sent to the network. The attacker can modify this list to downgrade capabilities reported by the UE and forward it to the network. To protect against such modification, both 3G and LTE contain the partial 'matching history' mechanism discussed above. This allows UE to check that its original list of security capabilities are identical with the ones received by the network. We argue that similar protection for network capabilities is required because the DoS attack has a persistent nature. This would, of course, need a change in the LTE protocols. Again, with the use of operator public keys, it would be possible to use digital signatures to protect lists of capabilities broadcast by the network. Alternatively, the negotiation of network capabilities could be done after AKA is completed.

Fortunately, both timer-based recovery and a full 'matching history' mechanism are now standardized from LTE release 14 specifications. Detailed solution is available from [32, 5, 23, 18, 25]. Several baseband vendors also implement the timer mechanism in their devices.

Device Capability Protection. 3GPP should consider mandating security protection for UE capabilities. In particular, *UE Capability Enquiry* message should be accessible/requested by the eNodeB only after establishing RRC security. This will prevent rogue eNodeBs from accessing UE capabilities and also capability hijacking by a MitM as shown in D4. 3GPP has decided to add this mitigation to the LTE standard from release 15 specifications [35] and will also be applied to 5G networks as well. Even though if our fix is implemented into LTE standards, baseband vendors need more extended periods to update their basebands, and hence attackers can still exploit this vulnerability.

On the network operator side, eNodeB configuration should be changed such that an eNodeB should request *UE Capability Information* only after establishing RRC security. This is a straightforward fix and can be implemented by the operators as a software update on their eNodeBs. Nevertheless, in practice, only a minor number of operators are acquiring capabilities after security setup. The difference among various operators we tested indicates that this is an implementation problem and is affecting specific vendors.

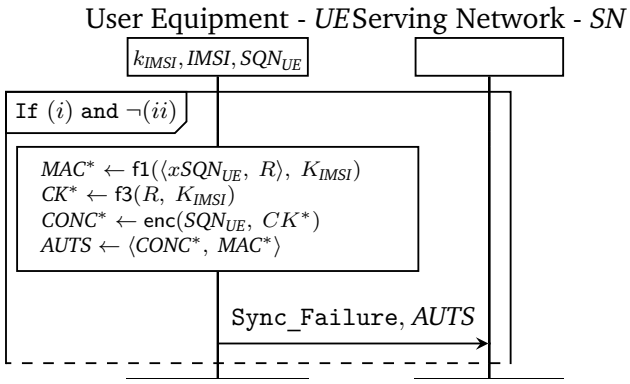

Fig. 5.8.: Fix F1: Fix by symmetrically encrypting SQN_{UE}

Fixing AKA protocol - Symmetrically Encrypt SQN_{UE}. We present an easy fix that is easy to deploy in the current cellular system and only requires changes in baseband and authentication server software in the HSS. We propose to modify the concealing mechanism: instead of using XOR (having algebraic relations enabling to cancel out AK^*), USIM may use symmetric encryption. Note that current USIMs and the HSS (in particular AuC) are already capable of symmetric encryption. The symmetric key to encrypt SQN_{UE} could be derived from the key K_{IMSI} and R in the received authentication challenge. The resulting fix is depicted in Figure 5.8. This can be very easily adapted to fix the linkability of failure messages as well. It suffices to hide the failure reason inside the ciphertext $CONC^*$ as follows:

$$CONC^* \leftarrow \text{enc}(\langle \texttt{Reason\_Failure},\ SQN_{UE}\rangle,\ CK^*).$$

The *HN* is required to decipher the $CONC^*$ to learn the reason for the *UE*'s authentication failure. However, this mechanism could add extra processing load on the *HN* due to the decryption requirement. Alternatively, such a processing load of the *HN* could be offloaded to the *SN* by transmitting decryption key in a set of authentication vectors. Finally, note that this solution suffers from a minor flaw: when the attacker triggers two times a synchronization failure by injecting the same authentication challenge while SQN_{UE} has not changed, then the two replied $CONC^*$ would be equal, leaking to the attacker the information that SQN_{UE} is still the same (we consider such an attack impractical though.

6 Future Work and Conclusion

The 3GPP SA3 group has standardized the 5G security architecture in [18]. The security gaps over the air-interface are narrowing down due to strict 5G security requirements in terms of privacy and availability. Although several issues related to privacy and availability of the network have been rectified based on our research, we identify potential problems that may still affect the various aspects of 5G networks. We consider these as challenging research questions for future work. We discuss them as follows and conclude this thesis.

A. 5G Security

The new architecture for 5G networks supports backward compatibility with previous generations. Hence, the 5G system will have inherent vulnerabilities from 4G, which should be addressed, mainly in the non-stand-alone type of network, where a 4G core is leveraged to support a 5G radio network. However, a new 5G core will address all the issues raised in this thesis and can be further investigated to identify new vulnerabilities.

Trade-offs. Security always comes at a cost. To address specific application needs in 5G, operators might have to design variable security requirements. In some instances, security compromises are possible to achieve required levels of speeds, performance, and low latencies (1ms). Operators need to understand the right balance with the evolving threat landscape and appropriately adjust security levels of the network. Using encryption and integrity protection is under the operator's control, and the subscriber has no mechanism to verify the security levels of the network. Note that the majority of these new security features introduced in 5G, are optional and operators have the freedom to activate/deactivate them in the live networks. Hence, 5G trade-offs can lead to some dangerous configuration flaws that may compromise privacy of users.

Security exceptions. The problem of pre-authentication messages, *w.r.t* subscriber management protocols can appear in 5G networks again. Since, other protocol messages explicitly not quoted in this research still prevail in the specifications and

should require similar recovery mechanisms. We find an open challenge here in the 5G NAS protocols [3] Although the use of timer-based recovery is proposed, some of the baseband vendors we tested in 2019 still did not implement these mitigations. Understandably, updating requires a considerable amount of time, but security cannot be an add-on feature anymore and should be fixed before early 5G network deployments.

Solving Fake Base Station. Fake base station has been a continuous problem in mobile networks, as shown in our research, and there is no standardized mechanism available to mitigate this. Although various other solutions are available in the market, they are not efficient, as shown by [42, 124]. 3GPP has finally decided to study potential solutions to tackle this issue in 5G networks [22]. For instance, in [18], UE-assisted network-based detection is proposed, in which *measurement reports* are leveraged to detect fake base stations. Nevertheless, it is not a robust solution, and the attacker can still circumvent detection techniques. Hence, 3GPP offers the liberty to baseband vendors to implement proprietary technologies. Trade-offs against battery consumption should be considered for implementing a solution inside the baseband.

Prevention is also important as much as detecting a fake base station, since, connecting to a fake base station itself is a DoS. Further, if various other protocol messages can be used for detection is still an open question. Also, it is prudent to investigate if machine learning techniques could be useful in this area of research to adapt to evolving attacker's capabilities.

B. Conclusion

We conclude with our contributions to improving 4G and 5G access network protocols and recommendations for security design in future specifications.

LTE Research Testbed. We developed a low-cost experimental testbed to mount various passive, semi-passive, and active attacks in 4G and 5G networks. The testbed comprises of SDR as hardware and customized implementations of open-source LTE end-user and network side protocol stacks.

New Vulnerabilities in 4G and 5G Networks. In this thesis, we discovered vulnerabilities in LTE access network protocols that lead to new privacy and availability attacks on subscribers and networks. Primarily, the weaknesses are a result of the trade-offs between LTE security and performance and availability.

We demonstrated location leak attacks that can precisely locate LTE users in a target region. Next, we show rogue devices can poison the operator's measurement data due to vulnerable implementations of SON based infrastructure. We present a new type of fingerprinting devices and subscribers in mobile networks using our MNmap tool. We also disclose a subtle vulnerability in the AKA protocol affecting 4G and 5G networks and allow attackers to learn a new type of privacy-sensitive information about subscribers; *i.e.,* consumption patterns.

We discuss DoS attacks that completely deny mobile service to subscribers and degrade the LTE service and also downgrade them to use less secure networks such as 2G/3G. This shows that the LTE registration process is not tamper-resistant and can be hijacked with MITM relay. We reveal various misconfigurations inside the operator's networks that catalyze the attacks and further amplify the consequences of them. We identify basebands that violate LTE security specifications.

We demonstrated all our attacks could be mounted using our testbed and using real devices in commercial networks. *The majority of our attacks due to the design shortcomings of LTE protocols* and hence has a strong impact and affect billions of devices worldwide that implement 3GPP standards. Furthermore, the *attacks remain stealthy and persistent on the end-user devices*.

Security Trade-off Analysis. We studied the underlying reasons for the vulnerabilities by analyzing their trade-offs and their effectiveness today. The need for engineering the correct trade-offs between security and other requirements (availability, performance, and functionality) led to the vulnerabilities in the first place. Such trade-offs are essential for the success of any large-scale system. But the trade-off equilibrium points are not static. We recommend that future standardization efforts take this into account.

3GPP follows the excellent practice of documenting exceptions when specification needs to deviate from the general security design principles recommended by the security working group (as was the case with L3 or D1/D2/D3/D4/D5). We recommend further that each such exception should also trigger an analysis of its implications. For example, if an exception is made to forego integrity protection for a denial message from the network, then the standards group should consider what happens and how to recover if the denial message contains incorrect information.

The design philosophy of LTE security required leaving some safety margin in security mechanisms to protect against changes in trade-offs. However, we learned in this research that the safety margins turn out to be too narrow. As general learning

on an abstract concept level, it would be better to include agility in the security mechanisms instead of a rigid safety margin. The future 5G technology will offer better possibilities to engineer agility and flexibility for security because software-defined networking and cloud computing are among the key concepts of emerging 5G architectures.

Applied Mitigations. We proposed mitigations to restore privacy and availability aspects in 4G and 5G networks. They are enforced into the specifications, and several LTE network operators have updated their configurations and settings. Further, baseband vendors and handset manufacturers have addressed our security findings and fixed their implementations. It is essential that the lessons learned from 4G networks should be applied to 5G networks that can be resilient to future threats.

A summary of our research, including discovered vulnerabilities, attacks, trade-off considerations, and mitigations that are enforced into standards and operational networks, are presented in the Table 1.1.

Acknowledgements

First, I would like to thank my supervisor, Prof. Dr. Jean-Pierre Seifert, for his guidance, motivation, support, and, most importantly, believing in me and giving the freedom to work on the subject that I find interesting and impactful for the community. This research wouldn't have been possible without his support and also the team of security in telecommunications group at TU Berlin. I want to extend my sincere gratitude to Ravishankar Borgaonkar, who has always been there to brainstorm, discuss ideas and guide me through setting up the right research goals. Further, I would like to thank Prof. N. Asokan and Prof. Valtteri Niemi for their contributions to an important chapter of this research. It was an honor working with them! I want to extend my thanks to Shinjo Park, with whom I was co-working, and our collaboration has resulted in excellent research in 4G and 5G networks.

I would like to thanks the mobile industry, especially Qualcomm, Huawei, GSMA, 3GPP, and Deutsche Telekom, for their constant support towards this research. In particular, I would like to thank the members of GSMA, especially James Moran, Eric Gauthier, Howard Peter, Samantha Kight, for actively supporting our research through their CVD program. Moreover, I'd like to thank Qualcomm and Alex Gantman for their support for this thesis. Also, I would like to extend my gratitude to Nico Golde and Kevin Redon for smoothly handling the bug reports and for their valuable reviews and comments on my research. I thank Stefan Schröder, Peter Howard, Steve Babbage, Günther Horn, Alf Zugenmeier, Silke Holtmanns, and the anonymous reviewers for their thoughtful feedback on our papers. I want to thank the open-source community team, especially the Osmocom, SRSLTE, and OpenLTE groups, for their excellent work. Without these projects, security investigations in the field of telecommunications, in general, wouldn't have been possible. Last but not least, I'd like to thank my family and friends for all the encouragement, love, and support over the years.

Appendix

Acronyms

A

- **3GPP** Third Generation Partnership Project
- **AKA** Authentication and Key Agreement protocol
- **ANR** Automatic Neighbour Relation
- **AS** Access Stratum
- **AUTN** Authentication Token
- **DoS** Denial-of-Service
- **E-UTRAN** Evolved Universal Terrestrial Radio Access Network
- **ECGI** E-UTRA Cell Global Identifier
- **EMM** EPS Mobility Management
- **eNodeB** evolved NodeB
- **EPC** Evolved Packet Core
- **EPS** Evolved Packet System
- **GSMA** GSM Association
- **GUTI** Globally Unique Temporary Identifier
- **HN** Home Network
- **IMEI** International Mobile Equipment Identity
- **IMSI** International Mobile Subscriber Identity
- **KPI** Key Performance Indicator
- **LTE** Long Term Evolution
- **M2M** Machine to Machine
- **MCC** Mobile Country Code
- **MDT** Minimization of Drive Test
- **MITM** Man In The Middle
- **MME** Mobility Management Entity
- **MNC** Mobile Network Code
- **MRO** Mobility Robustness Optimization
- **NAS** Non Access Stratum
- **NB-IoT** Narrow Band - Internet of Things
- **OAM** Operation, Administration, and Maintenance
- **PCI** Physical Cell ID
- **PKI** Public Key Infrastructure
- **PSS** Primary Synchronization Signal
- **RACH** Random Access Channel

- **RLF** Radio Link Failure
- **RRC** Radio Resource Control
- **RSSI** Radio Signal Strength Indicator
- **SIB** System Information Block
- **SN** Serving Network
- **SON** Self Organized Network
- **SQN** Sequence Number
- **SSS** Secondary Synchronization Signal
- **TA** Tracking Area
- **TAC** Tracking Area Code
- **UE** User Equipment
- **USIM** Universal Subscriber Identity Module
- **USRP** Universal Software Radio Peripheral
- **V2V** Vehicle to Vehicle
- **VoLTE** Voice over LTE

B

Test Devices & Fingerprints

1. Devices for Reference Model

Manufacturer	Model	Baseband Type
Samsung	Galaxy Alpha	Intel XMM7260
Samsung	Galaxy S6	Samsung Exynos Modem 333
Samsung	Galaxy S7	Samsung Exynos 8890
Samsung	Galaxy S8	Samsung Exynos 8895
Huawei	Honor 7	Kirin 935
Huawei	P20	Kirin 970
HTC	One E9	MediaTek X10
LG	G Flex 2	Qualcomm MSM8994
Sony	Xperia Z5	Qualcomm MSM8994
Sony	Xperia X	Qualcomm MSM8956
Planet Computer	Gemini	MediaTek X27
Apple	iPhone 6	Qualcomm MDM9625
Apple	iPhone 8	Intel XMM7480
Apple	iPhone 8 (US)	Qualcomm MDM9655
Apple	iPhone X (US)	Qualcomm MDM9655
Google	Nexus 5X	Qualcomm MSM8992
Nokia	8110 4G	Qualcomm MSM8905
Asus	ZenFone 2E	Intel XMM7160
Huawei	E3372	Huawei
Samsung	GT-B3740	Samsung CMC220
Sierra Wireless	EM7455	Qualcomm MDM9635
Fibocom	L850-GL	Intel XMM7360
Telit	LN930	Intel XMM7160
AVM	FritzBox LTE	Intel XMM7160
Huawei	B310s	Huawei
Netgear	Nighthawk	Qualcomm MDM9250
GlocalMe	G2	Qualcomm MSM8926
Quectel	BC68	Huawei NB-IoT
Quectel	BC66	MediaTek NB-IoT
Quectel	BG69	Qualcomm MDM9206
Audi	A6	Qualcomm MDM9635
Samsung	SM-V110K	Qualcomm MDM9206
Mobile Eco	ME-K60KL	Qualcomm MDM9206
Apple	Watch Series 3	Qualcomm MDM9635M
Huawei	MediaPad M5	Kirin 960
Apple	iPad 5th gen	Qualcomm MDM9625M

Tab. B.1.: UE's from Phones, Laptop, Cars, IoT chipsets, USB Data sticks, and etc.

2. Device Fingerprints

Capability	Huawei	Samsung	Intel	Mediatek	Qualcomm
CM Service Prompt	1	0	0	0	1
EIA0	1	1	1	1	0
Access class control for CSFB	0	1	0	1	1
Extended Measurement Capability	0	0	0	1	0

Tab. B.2.: Differences among Baseband Vendors

Capability	Cellular	Cellular IoT
PSM timer: T3324	0	1
Extended timer for periodic TAU: T3412 ext	0	1

Tab. B.3.: Cellular vs. Cellular IoT

Capability	Phone	Other
UE's usage setting	Voice Centric or Data Centric	Not present
Voice domain preference for E-UTRAN	CS Voice or IMS PS Voice	Not present
UMTS AMR codec	Present	Not present

Tab. B.4.: Phone vs. Others

Capability	Android	iOS
MS assisted GPS	1	0
voiceOverPS-HS-UTRA-FDD-r9	1	0

Tab. B.5.: Android vs. iOS

List of Figures

C

List of Tables

D

Bibliography

[1] 3GPP. *3G security; Security architecture*. TS 33.102 (cit. on pp. 1, 13, 15, 70, 72, 76).

[2] 3GPP. *3G Security; Specification of the MILENAGE algorithm set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 2: Algorithm specification*. Technical Specification (TS) 35.206. (3GPP) (cit. on p. 15).

[3] 3GPP. *5G; Non-Access-Stratum (NAS) protocol for 5G System (5GS); TS 24.501 version 15.0.0 Release 15*. Technical Specification (TS) 24.501 (cit. on p. 104).

[4] 3GPP. *Characteristics of the Universal Subscriber Identity Module (USIM application) 3GPP TS 31.102 version 12.5.0 Release 12*. URL: `http://www.3gpp.org/dynareport/31102.htm` (cit. on p. 10).

[5] 3GPP. *Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Mobile radio interface Layer 3 specification; Core network protocols; Stage 3 (3GPP TS 24.008 version 14.4.0 Release 14)*. TS 24.008. 3rd Generation Partnership Project (3GPP) (cit. on pp. 71, 100).

[6] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2*. TS 36.300 (cit. on pp. 10, 20, 42, 44–46, 93).

[7] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); overall description; stage 2, Specification 3GPP TS 36.300 version 12.4.0 Release 12*. URL: `http://www.3gpp.org/dynareport/36300.htm` (cit. on pp. 17, 49).

[8] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification, version 15.3.0 Release 15*. TS 36.331. 3rd Generation Partnership Project (3GPP) (cit. on pp. 7, 11, 36, 46, 63, 66, 71, 77, 81, 90, 93).

[9] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode*. TS 36.304 (cit. on p. 36).

[10] 3GPP. *evolved universal terrestrial radio access (E-UTRA); user equipment (UE) procedures in idle mode; Specification 3GPP TS 36.304 version 12.4.0 Release 12*. URL: http://www.3gpp.org/dynareport/36304.htm (cit. on pp. 17, 55).

[11] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Self-configuring and self-optimizing network (SON) use cases and solutions*. TR 36.902. 3rd Generation Partnership Project (3GPP) (cit. on p. 20).

[12] 3GPP. *LTE Evolved Universal Terrestrial Radio Access (E-UTRA) radio resource control (RRC) protocol specification," Specification 3GPP TS 36.331 version 12.3.0 Release 12*. URL: http://www.3gpp.org/dynareport/36331.htm (cit. on pp. 42, 43, 46, 51, 54, 55, 63).

[13] 3GPP. *LTE;Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification (3GPP TS 36.323 version 14.3.0 Release 14)*. TS 36.323. 3rd Generation Partnership Project (3GPP) (cit. on p. 93).

[14] 3GPP. *Mobile radio interface Layer 3 specification; Core network protocols; Stage 3*. URL: http://www.3gpp.org/dynareport/24008.htm (cit. on p. 90).

[15] 3GPP. *Network Architecture ; Specification 3GPP TS 23.002 version 12.7.0 Release 12*. URL: http://www.3gpp.org/DynaReport/23002.htm (cit. on p. 10).

[16] 3GPP. *Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3 Specification 3GPP TS 24.301 version 12.8.0 Release 12*. URL: http://www.3gpp.org/dynareport/24301.htm (cit. on pp. 11, 17, 39, 70, 71, 88, 98).

[17] 3GPP. *Rationale and track of security decisions in Long Term Evolved(LTE) RAN / 3GPP System Architecture Evolution (SAE (Release 8), TR 33.821 V1.1.0*. URL: http://www.3gpp.org/DynaReport/33821.htm (cit. on pp. 2, 3, 6, 22, 25, 27, 33, 63).

[18] 3GPP. *Security architecture and procedures for 5G System*. Technical Specification (TS) 33.501. 3rd Generation Partnership Project (3GPP), 2018 (cit. on pp. 13, 15, 29, 70, 82–84, 99, 100, 103, 104).

[19] 3GPP. *Service requirements for the Evolved Packet System (EPS)*. TS 122.278. (3GPP) (cit. on p. 33).

[20] 3GPP. *Service requirements for V2X services*. Technical Specification (TS) 22.185. 3rd Generation Partnership Project (3GPP), 2018 (cit. on p. 78).

[21] 3GPP. *Specification of the TUAK algorithm set: A second example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 1: Algorithm specification*. Technical Specification (TS) 35.231. (3GPP) (cit. on p. 15).

[22] 3GPP. *Study on 5G security enhancements against false base stations*. Technical Report (TR) 33.809. 2018 (cit. on pp. 6, 29, 67, 98, 104).

[23] 3GPP. *System Architecture Evolution (SAE); Security architecture*. TS 33.401 (cit. on pp. 1, 13, 15, 42, 70, 78, 100).

[24] 3GPP. *System Architecture Evolution (SAE); Security architecture; (3GPP 33.401 version 12.14.0 Release 12)*. URL: `http://www.3gpp.org/dynareport/33.401.htm` (cit. on pp. 10, 13, 72).

[25] 3GPP. *System architecture for the 5G System (5GS)*. Technical Specification (TS) 23.501. 3rd Generation Partnership Project (3GPP), 2018 (cit. on pp. 82, 100).

[26] 3GPP. *Technical Specification Group GSM/EDGE Radio Access Network; Cellular system support for ultra-low complexity and low throughput Internet of Things (CIoT) (Release 13)*. TS 45.820. 3rd Generation Partnership Project (3GPP), Nov. 2015 (cit. on pp. 80, 94).

[27] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio access capabilities (Release 13)*. TS 36.306 (cit. on pp. 71, 79, 90–92).

[28] 3GPP. *Technical Specification Group Services and System Aspects; Study on the security aspects of the next generation system (Release 14)*. TR 33.899 (cit. on pp. 1, 41).

[29] 3GPP. *Telecommunication management; Self-Organizing Networks (SON); Concepts and requirements*. TS 32.500. 3rd Generation Partnership Project (3GPP), Dec. 2009 (cit. on p. 19).

[30] 3GPP. *TS 36.133. Evolved Universal Terrestrial Radio Access (E-UTRA); Requirements for support of radio resource management; Version 8.23.0; Release 8*. URL: `http://www.3gpp.org/dynareport/36133.htm` (cit. on p. 36).

[31] 3GPP. *Universal Mobile Telecommunications System (UMTS); Numbering, addressing and identification (3GPP TS 23.003 version 12.5.0 Release 12)*. URL: `http://www.3gpp.org/dynareport/23003.htm` (cit. on pp. 10, 13).

[32] 3GPP. *Universal Mobile Telecommunications System (UMTS);LTE; 5G; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3 Release 15*. TS 24.301 (cit. on pp. 13, 71, 72, 79, 100).

[33] 3GPP. *Universal Terrestrial Radio Access (UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRA); Radio measurement collection for Minimization of Drive Tests (MDT); Overall description; Stage 2*. URL: http://www.3gpp.org/DynaReport/37320.htm (cit. on pp. 41, 55).

[34] 3GPP SA3. *S3-152498; LS on backoff timer*. URL: http://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_81_Anaheim/Docs/S3-152498.zip (cit. on p. 7).

[35] 3GPP SA3. *S3-192271; LS reply on Handling of UE radio network capabilities in 4G and 5G*. URL: https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_95Bis_Sapporo/Docs/S3-192271.zip (cit. on pp. 7, 101).

[36] 3GPP SA3. *SP-160580; Protecting against the modification of Attach/TAU Request attacks*. URL: http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_73/Docs/SP-160580.zip (cit. on p. 7).

[37] 4G Americas. *Self-Optimizing networks in 3GPP Release 11: The benefits of SON in LTE, Whitepaper*. Oct. 2013 (cit. on p. 46).

[38] *Amarisoft* (cit. on p. 28).

[39] Apple Inc. *Apple security updates*. https://support.apple.com/en-us/HT201222 (cit. on p. 82).

[40] Myrto Arapinis, Loretta Mancini, Eike Ritter, et al. „New Privacy Issues in Mobile Telephony: Fix and Verification“. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*. CCS '12. Raleigh, North Carolina, USA: ACM, 2012, pp. 205–216 (cit. on pp. 22, 82).

[41] Ben Wojtowicz. *OpenLTE Project Homepage*. URL: http://openlte.sourceforge.net/ (cit. on pp. 3, 28).

[42] Ravishankar Borgaonkar, Andrew Martin, Shinjo Park, Altaf Shaik, and Jean-Pierre Seifert. „White-stingray: Evaluating IMSI Catchers Detection Applications“. In: *Proceedings of the 11th USENIX Conference on Offensive Technologies*. WOOT'17. Vancouver, BC, Canada: USENIX Association, 2017, pp. 21–21 (cit. on pp. 40, 104).

[43] J.J. Caffery and G.L. Stuber. „Overview of radiolocation in CDMA cellular systems“. In: *Communications Magazine, IEEE* 36.4 (Apr. 1998), pp. 38–45 (cit. on p. 55).

[44] *Cell Mapper*. URL: https://play.google.com/store/apps/details?id=cellmapper.net.cellmapper (cit. on p. 52).

[45] Stephanie Clifford and Quentin Hardy. „Attention Shoppers: Store is Tracking Your Cell“. In: *New York Times* 14 (2013) (cit. on p. 84).

[46] David Nowoswiat. *Managing LTE core network signaling traffic*. URL: http://www2.alcatel-lucent.com/techzine/managing-lte-core-network-signaling-traffic/ (cit. on p. 18).

[47] David Pogue. *Two Tips for Facebook Users*. URL: %7Bhttp://pogue.blogs.nytimes.com/2013/07/18/two-tips-for-facebook-users/?src=twr&smid=tw-nytimes&_r=0%7D (cit. on p. 48).

[48] Whitfield Diffie, Paul C. Van Oorschot, and Michael J. Wiener. „Authentication and Authenticated Key Exchanges“. In: *Des. Codes Cryptography* 2.2 (June 1992), pp. 107–125 (cit. on p. 98).

[49] ENAiKOON. *OpenCellID*. URL: http://opencellid.org/ (cit. on p. 52).

[50] William Enck, Patrick Traynor, Patrick McDaniel, and Thomas F. La Porta. „Exploiting open functionality in SMS-capable cellular networks“. In: *Proceedings of the 12th ACM Conference on Computer and Communications Security, CCS 2005, Alexandria, VA, USA, November 7-11, 2005*. 2005, pp. 393–404 (cit. on p. 22).

[51] Tobias Engel. *Locating Mobile Phones using Signalling System 7*. https://berlin.ccc.de/~tobias/25c3-locating-mobile-phones.pdf (cit. on p. 86).

[52] Ettus. *USRP B210*. URL: http://www.ettus.com/product/details/UB210-KIT (cit. on pp. 3, 30).

[53] Facebook Inc. *Facebook Messenger*. URL: https://www.messenger.com/features (cit. on p. 35).

[54] Facebook Inc. *Facebook Mobile*. URL: https://www.facebook.com/mobile/ (cit. on pp. 35, 48).

[55] Dan Forsberg, Günther Horn, Wolf-Dietrich Moeller, and Valtteri Niemi. *LTE security*. John Wiley & Sons, 2012 (cit. on pp. 23, 65, 99).

[56] Dirk Fox. „Der IMSI-catcher“. In: *Datenschutz und Datensicherheit* 26.4 (2002), pp. 212–215 (cit. on pp. 27, 28).

[57] Gamry Instruments. *The Faraday Cage: What is it? How does it work?* URL: http://www.gamry.com/application-notes/instrumentation/faraday-cage/ (cit. on p. 39).

[58] Nico Golde, Kevin Redon, and Ravishankar Borgaonkar. „Weaponizing Femtocells: The Effect of Rogue Devices on Mobile Telecommunications“. In: *19th Annual Network and Distributed System Security Symposium, NDSS, 2012, San Diego, California, USA, February 5-8, 2012*. 2012 (cit. on pp. 22, 27, 29).

[59] Nico Golde, Kévin Redon, and Jean-Pierre Seifert. „Let Me Answer That for You: Exploiting Broadcast Information in Cellular Networks“. In: *Proceedings of the 22Nd USENIX Conference on Security*. SEC'13. Washington, D.C.: USENIX Association, 2013, pp. 33–48 (cit. on p. 52).

[60] Oded Goldreich. *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press, 2004 (cit. on p. 34).

[61] Google Inc. *Android Security Bulletin*. `https://source.android.com/security/bulletin` (cit. on p. 82).

[62] *GSM Map*. (Cit. on p. 28).

[63] GSMA. *3GPP Low Power Wide Area Technologies, GSMA white paper*. Tech. rep. GSMA, 2016 (cit. on pp. 80, 81).

[64] GSMA. *GSMA Announces New Initiatives Focusing On Creating More Efficient Mobile Applications*. URL: `http://www.gsma.com/newsroom/press-release/gsma-announces-new-initiatives-focusing-on-creating-more-efficient-mobile-applications/` (cit. on p. 64).

[65] GSMA. *GSMA Mobile Security Hall of Fame*. URL: `https://www.gsma.com/security/gsma-mobile-security-hall-of-fame/` (cit. on pp. 6, 40).

[66] GSMA. *The Mobile Economy 2019*. URL: `https://www.gsmaintelligence.com/research/?file=b9a6e6202ee1d5f787cfebb95d3639c5&download` (cit. on p. 1).

[67] *GSMArena.com*. `https://www.gsmarena.com/team.php3` (cit. on pp. 79, 80).

[68] GSMK. *GSMK Overwatch: IMSI Catcher Detection*. `https://www.gsmk.de/products/network-security/#overwatch` (cit. on p. 40).

[69] GSM-R. URL: `https://en.wikipedia.org/wiki/GSM-R` (cit. on p. 80).

[70] Mordechai Guri, Yisroel Mirsky, and Yuval Elovici. „9-1-1 DDoS: Attacks, Analysis and Mitigation“. In: *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. New York, NY, USA: IEEE, 2017, pp. 218–232 (cit. on p. 23).

[71] K. M. J. Haataja and K. Hypponen. „Man-In-The-Middle attacks on bluetooth: a comparative analysis, a novel attack, and countermeasures“. In: *2008 3rd International Symposium on Communications, Control and Signal Processing*. Mar. 2008, pp. 1096–1102 (cit. on p. 25).

[72] K. Haataja and P. Toivanen. „Two practical man-in-the-middle attacks on Bluetooth secure simple pairing and countermeasures“. In: *IEEE Transactions on Wireless Communications* 9.1 (Jan. 2010), pp. 384–392 (cit. on p. 25).

[73] Changhee Hahn, Hyunsoo Kwon, Daeyoung Kim, Kyungtae Kang, and Junbeom Hur. „A Privacy Threat in 4th Generation Mobile Telephony and Its Countermeasure“. In: *The 9th International Conference on Wireless Algorithms, Systems, and Applications* (2014), pp. 624–635 (cit. on p. 82).

[74] Harris Corporation. *Trademark registration for STINGRAY*. http://tsdr.uspto.gov/documentviewer?caseId=sn76303503 (cit. on pp. 2, 27, 28).

[75] S. Holtmanns and I. Oliver. „SMS and one-time-password interception in LTE networks“. In: *2017 IEEE International Conference on Communications (ICC)*. May 2017, pp. 1–6 (cit. on p. 27).

[76] B. Hong, S. Park, H. Kim, et al. „Peeking Over the Cellular Walled Gardens - A Method for Closed Network Diagnosis -“. In: *IEEE Transactions on Mobile Computing* 17.10 (Oct. 2018), pp. 2366–2380 (cit. on p. 70).

[77] Huang, Lin. „Forcing a Targeted LTE Cellphone into an Eavesdropping Network“. In: *Hack In The Box*. 2016 (cit. on p. 23).

[78] Huawei Technologies. *eRAN ANR Management Feature Parameter Description*. https://www.scribd.com/document/319018225/Huawei-ANR-Management-ERAN7-0-04. 2015 (cit. on pp. 44, 46).

[79] Huawei Technologies. *eRAN TDD MRO Feature Parameter Description*. http://www.honorcup.ru/upload/iblock/164/7.pdf. 2016 (cit. on p. 46).

[80] Huawei Technologies. *eWBB2.0 DBS3900 LTE TDD Product Description*. http://www.huawei.com/ilink/enenterprise/download/HW_205528. 2012 (cit. on p. 60).

[81] Huawei Technologies. *LTE eRAN3.0 Handover Fault Diagnosis*. https://www.scribd.com/document/138513253/Huawei-LTE-Handover-events. 2011 (cit. on p. 46).

[82] HUAWEI TECHNOLOGIES CO., LTD. *eRAN2.0 Feature Description*. https://www.scribd.com/document/132066434/Huawei-LTE-eRAN2-1-Feature-Description-doc. Sept. 2010 (cit. on p. 90).

[83] Huawei Technologies Co., Ltd. *All Bulletins - PSIRT*. https://www.huawei.com/en/psirt/all-bulletins (cit. on p. 82).

[84] Huawei-PSIRT. *Security Advisory - UE Information Leak*. URL: https://www.huawei.com/en/psirt/security-advisories/2016/huawei-sa-20160520-03-smartphone-en (cit. on pp. 6, 40).

[85] Huawei-PSIRT. *Security Advisory - UE Measurement Leak*. URL: http://www1.huawei.com/en/security/psirt/security-bulletins/security-advisories/hw-459832.htm (cit. on pp. 6, 40).

[86] S. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino. „LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE“. In: *Network and Distributed Systems Security (NDSS) Symposium 2018* (Feb. 2018) (cit. on p. 24).

[87] IDC. *Smartphone Vendor Market Share, Q1 2015*. URL: http://www.idc.com/prodserv/smartphone-market-share.jsp (cit. on p. 55).

[88] Jill Jermyn, Gabriel Salles-Loustau, and Saman Zonouz. „An Analysis of DoS Attack Strategies Against the LTE RAN“. In: *Journal of Cyber Security, 3(2):159–180*. 2014 (cit. on p. 23).

[89] Jim Forster. *OpenBTS and Range Networks*. URL: http://www.mastel.or.id/files/Open%5C%20BTS%5C_Jim%5C%20Foster.pdf (cit. on p. 39).

[90] R.P. Jover. „Security attacks against the availability of LTE mobility networks: Overview and research directions“. In: *Wireless Personal Multimedia Communications (WPMC), 2013 16th International Symposium on*. June 2013, pp. 1–9 (cit. on p. 96).

[91] JPL Wireless communications. *Cell Sizes*. URL: http://www.wirelesscommunication.nl/reference/chaptr04/cellplan/cellsize.htm (cit. on p. 52).

[92] H. Kim, J. Lee, E. Lee, and Y. Kim. „Touching the Untouchables: Dynamic Security Analysis of the LTE Control Plane“. In: *2019 IEEE Symposium on Security and Privacy (SP)*. May 2019, pp. 1153–1168 (cit. on p. 24).

[93] Hongil Kim, Dongkwan Kim, Minhee Kwon, et al. „Breaking and Fixing VoLTE: Exploiting Hidden Data Channels and Mis-implementations“. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*. 2015, pp. 328–339 (cit. on p. 23).

[94] Chris Paget aka Kristin Paget. *Practical cellphone spying*. 2010 (cit. on p. 27).

[95] KUHNE electronic. *KU PA BB 005300-3 A, RF Broadband Power Amplifier*. URL: https://shop.kuhne-electronic.com/kuhne/en/shop/industrial/prof-power-amplifier/prof-broadband/KU+PA+BB+0053003+A++RF+Broadband+Power+Amplifier/?card=507#_tab_content2 (cit. on p. 31).

[96] Denis Foo Kune, John Kölndorfer, Nicholas Hopper, and Yongdae Kim. „Location leaks over the GSM air interface“. In: *19th Annual Network and Distributed System Security Symposium,NDSS 2012, San Diego, California, USA, February 5-8, 2012* (2012) (cit. on pp. 22, 43, 48, 53).

[97] M. Labib, V. Marojevic, and J. H. Reed. „Analyzing and enhancing the resilience of LTE/LTE-A systems to RF spoofing“. In: (Oct. 2015), pp. 315–320 (cit. on p. 23).

[98] P.P.C. 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*. May 2007, pp. 1289–1297 (cit. on p. 22).

[99] Chi-Yu Li, Guan-Hua Tu, Chunyi Peng, et al. „Insecurity of Voice Solution VoLTE in LTE Mobile Networks“. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*. 2015, pp. 316–327 (cit. on p. 23).

[100] Marc Lichtman, Roger Piqueras Jover, Mina Labib, et al. „LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation“. In: *IEEE Communications Magazine* 54.4 (Apr. 2016), pp. 54–61 (cit. on p. 23).

[101] Lime Microsystems. *LimeSDR*. `https://www.crowdsupply.com/lime-micro/limesdr`. 2016 (cit. on pp. 30, 31).

[102] Advanced Card Systems Holdings Limited. *ACR38 Smart Card Reader* (cit. on p. 31).

[103] Alcatel Lucent. „Application Note, LTE SUBSCRIBER SERVICE RESTORATION“. In: (2014) (cit. on p. 69).

[104] M Ma. „Security Investigation in 4G LTE Networks“. In: *IEEE GLOBECOM*. 2012 (cit. on p. 23).

[105] Magdalena Nohrborg. *Self-Organizing Networks* (cit. on p. 19).

[106] Matt Ettus. *Ettus Research update*. URL: `%7Bhttp://static1.1.sqspcdn.com/static/f/679473/23654458/1381240753367/grcon13_ettus_products.pdf?token=ldHVQF0yAdZLWvdjhPjqLtrhB9I%5C%3D%7D` (cit. on p. 39).

[107] Melih Tufan. *Packet Networks Portfolio*. URL: `%7Bhttp://www.ericsson.com/ericsson/investors/doc/2011/ap_forum/ericsson_apac_forum_150911_packet_networks.pdf%7D` (cit. on p. 18).

[108] M. Miettinen, S. Marchal, I. Hafeez, et al. „IoT SENTINEL: Automated Device-Type Identification for Security Enforcement in IoT“. In: *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. Vol. 00. June 2017, pp. 2177–2184 (cit. on p. 22).

[109] Stig F Mjølsnes and Ruxandra F Olimid. „Easy 4G/LTE IMSI Catchers for Non-Programmers“. In: *International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security*. Feb. 2017, pp. 235–246. eprint: `1702.04434` (cit. on p. 28).

[110] Johan Moberg, Mattias Löfgren, and Robert Karlsson. „Throughput of the WCDMA Random Access Channel“. In: *IST Mobile Communications* (Jan. 2000) (cit. on p. 24).

[111] ABM Musa and Jakob Eriksson. „Tracking unmodified smartphones using Wi-Fi monitors“. In: *Proceedings of the 10th ACM conference on embedded network sensor systems*. ACM. 2012, pp. 281–294 (cit. on p. 84).

[112] *Nighthawk® M1 Mobile Router*. 2018 (cit. on p. 93).

[113] Karsten Nohl and Sylvain Munaut. „Wideband GSM Sniffing“. In: *27th Chaos Communication Congress*. 2010 (cit. on pp. 28, 48).

[114] Karsten Nohl and Chris Paget. „GSM - SRSLY?“ In: *CCC* (2009) (cit. on p. 28).

[115] Nokia Blog. *A signaling storm is gathering – Is your packet core ready?* URL: https://blog.networks.nokia.com/mobile-networks/2012/12/05/a-signaling-storm-is-gathering-is-your-packet-core-ready/ (cit. on pp. 48, 69).

[116] Nokia Networks. *Voice over LTE (VoLTE) Optimization*. URL: http://networks.nokia.com/sites/default/files/document/nokia%5C_volte%5C_optimization%5C_white%5C_paper%5C_071114.pdf (cit. on pp. 18, 51).

[117] P. O'Hanlon, R. Borgaonkar, and L. Hirschi. „Mobile Subscriber WiFi Privacy“. In: *2017 IEEE Security and Privacy Workshops (SPW)*. May 2017, pp. 169–178 (cit. on p. 22).

[118] Piers O'Hanlon, Ravishankar Borgaonkar, and Lucca Hirschi. „Mobile subscriber WiFi privacy“. In: *Proceedings of Mobile Security Technologies (MoST'17), held as part of the IEEE Computer Society Security and Privacy Workshops (SPW'17)*. To appear. 2017 (cit. on p. 3).

[119] OpenAirInterface. „History“. In: () (cit. on p. 32).

[120] Osmocom. *RACH flood DoS*. URL: http://security.osmocom.org/trac/ticket/1 (cit. on p. 96).

[121] Osmocom. *RTL-SDR*. URL: http://sdr.osmocom.org/trac/wiki/rtl-sdr (cit. on p. 31).

[122] Osmocom Project. *What is GSMTAP?* URL: http://bb.osmocom.org/trac/wiki/GSMTAP (cit. on p. 32).

[123] *OsmocomBB*. URL: http://osmocom.org/projects/baseband (cit. on pp. 3, 28).

[124] Shinjo Park, Altaf Shaik, Ravishankar Borgaonkar, and Jean-Pierre Seifert. „Anatomy of Commercial IMSI Catchers and Detectors“. In: *Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society*. WPES'19. London, United Kingdom: ACM, 2019, pp. 74–86 (cit. on pp. 29, 104).

[125] Qualcomm. *Qualcomm Product Security*. URL: https://www.qualcomm.com/connect/contact/security/product-security/hall-of-fame (cit. on pp. 6, 40).

[126] Qualcomm Research. *LTE Small Cell SON Test Cases; Functionality and Interworking*. https://www.qualcomm.com/media/documents/files/lte-small-cell-son-test-cases.pdf. 2015 (cit. on p. 45).

[127] Qualcomm Research. *Small Cells and UltraSON*. https://www.qualcomm.com/media/documents/files/small-cells-and-ultrason-presentation.pdf. 2014 (cit. on pp. 45, 66).

[128] Qualcomm Technologies, Inc. *Qualcomm Technologies, Inc. Security Bulletin*. https://www.qualcomm.com/company/product-security/bulletins (cit. on p. 82).

[129] Quectel. *LTE BC68 NB-IoT Module*. https://www.quectel.com/product/bc68.htm (cit. on p. 94).

[130] Arvind Raghavan R. Piqueras Jover Joshua Lackey. „Enhancing the security of LTE networks against jamming attacks“. In: *EURASIP Journal on Information Security*. 2014 (cit. on p. 23).

[131] Radmilo Racic, Denys Ma, Hao Chen, and Xin Liu. „Exploiting Opportunistic Scheduling in Cellular Data Networks“. In: *Proceedings of the Network and Distributed System Security Symposium, NDSS 2008, San Diego, California, USA, 10th February - 13th February 2008*. 2008 (cit. on p. 22).

[132] Ramtim Amin. *4G Wireshark dissector for Samsung USB stick*. URL: http://labs.p1sec.com/2013/08/18/4g-wireshark-dissector-based-on-samsung-usb-stick/ (cit. on p. 49).

[133] Raghunandan M Rao, Sean Ha, Vuk Marojevic, and Jeffrey Reed. „LTE PHY Layer Vulnerability Analysis and Testing Using Open-Source SDR Tools“. In: *IEEE Military Communications Conference* (2017) (cit. on p. 23).

[134] R.Borgaonkar, A.Shaik, N.Asokan ,V.Niemi, J.P.Seifert. *LTE and IMSI catcher myths; Blackhat EU*. https://www.blackhat.com/docs/eu-15/materials/eu-15-Borgaonkar-LTE-And-IMSI-Catcher-Myths.pdf. Nov. 2015 (cit. on p. 22).

[135] Kenneth van Rijsbergen. „The effectiveness of a homemade IMSI catcher build with YateBTS and a BladeRF“. In: *University of Amsterdam* (2016) (cit. on p. 28).

[136] David Rupprecht, Adrian Dabrowski, Thorsten Holz, Edgar Weippl, and Christina Pöpper. „On Security Research Towards Future Mobile Network Generations“. In: *IEEE Communications Surveys and Tutorials* PP (Oct. 2017) (cit. on p. 21).

[137] David Rupprecht, Kai Jansen, and Christina Pöpper. „Putting LTE Security Functions to the Test: A Framework to Evaluate Implementation Correctness“. In: *10th USENIX Workshop on Offensive Technologies (WOOT 16)*. Austin, TX: USENIX Association, Aug. 2016 (cit. on p. 24).

[138] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. „Breaking LTE on Layer Two“. In: *IEEE Symposium on Security & Privacy (SP)*. IEEE, May 2019 (cit. on pp. 22, 38).

[139] Samsung Electronics. *Android Security Updates - Samsung Mobile Security*. `https://security.samsungmobile.com/securityUpdate.smsb` (cit. on p. 82).

[140] *SELFNET - Framework for Self-Organized Network Management in Virtualized and Software Defined Networks*. 2016 (cit. on p. 23).

[141] Semtech Corporation. *LoRa Modulation Basics - AN1200.22*. 2015 (cit. on p. 25).

[142] Altaf Shaik, Ravishankar Borgaonkar, N. Asokan, Valtteri Niemi, and Jean-Pierre Seifert. „Practical attacks against privacy and availability in 4G/LTE mobile communication systems“. In: *23rd Annual Network and Distributed System Security Symposium, NDSS San Diego, California, USA, February 21-24, 2016*. 2016 (cit. on p. 86).

[143] Sigfox S.A. *Sigfox Technical Overview*. 2017 (cit. on p. 25).

[144] Signals Research Group. *VoLTE Performance Analysis*. URL: `http://www.signalsresearch.com/Docs/LTE%5C%20NA%5C%202014%5C%20VoLTE%5C%20Results%5C%20-%5C%20SRG%5C%20Presentation.pdf` (cit. on p. 53).

[145] Neil Sinclair, David Harle, Ian A. Glover, James Irvine, and Robert C. Atkinson. „Parameter Optimization for LTE Handover Using an Advanced SOM Algorithm“. In: *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*. New York, NY, USA: IEEE, 2013, pp. 1–6 (cit. on p. 67).

[146] Yuvraj Singh. „Article: Comparison of Okumura, Hata and COST-231 Models on the Basis of Path Loss and Signal Strength“. In: *International Journal of Computer Applications* 59.11 (Dec. 2012). Full text available, pp. 37–41 (cit. on p. 39).

[147] *SPEEDTEST* (cit. on p. 93).

[148] SR Labs. *SnoopSnitch*. URL: `https://opensource.srlabs.de/projects/snoopsnitch` (cit. on pp. 8, 48, 66).

[149] srsLTE. *Open source 3GPP LTE library*. URL: `https://github.com/srsLTE/srsLTE` (cit. on pp. 3, 28).

[150] Stoke. *Charting the Signaling Storms*. URL: http://www.slideshare.net/zahidtg/charting-the-signaling-storms-stoke (cit. on pp. 48, 69).

[151] Daehyun Strobel. „IMSI Catcher“. In: *Chair for Communication Security, Ruhr-Universität Bochum* (2007), p. 14 (cit. on pp. 33, 36).

[152] Da-Zhi Sun, Yi Mu, and Willy Susilo. „Man-in-the-middle Attacks on Secure Simple Pairing in Bluetooth Standard V5.0 and Its Countermeasure“. In: *Personal Ubiquitous Computing*. 22.1 (Feb. 2018), pp. 55–67 (cit. on p. 25).

[153] Sysmocom. *sysmoUSIM-SJS1* (cit. on pp. 31, 86).

[154] T. Engel. *SS7: Locate. Track. Manipulate. Chaos Communication Congress, 31C3*. 2014 (cit. on p. 27).

[155] Telit. *Machine to Machine Communication*. URL: http://www.telit.com/experience-telit/what-is-m2m/general-information/ (cit. on p. 10).

[156] Patrick Traynor, Michael Lin, Machigar Ongtang, et al. „On cellular botnets: measuring the impact of malicious devices on a cellular network core“. In: *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, November 9-13, 2009*. 2009, pp. 223–234 (cit. on pp. 22, 23, 27).

[157] Patrick Traynor, Patrick McDaniel, and Thomas La Porta. „On Attack Causality in Internet-connected Cellular Networks“. In: *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*. SS'07. Boston, MA: USENIX Association, 2007, 21:1–21:16 (cit. on p. 22).

[158] Udar Swapnil. *Darshak Framework*. URL: https://github.com/darshakframework/darshak (cit. on pp. 8, 48, 66).

[159] UmTRX. *Open source, cost optimised and future-proofing flexibility*. URL: http://umtrx.org/about/ (cit. on p. 39).

[160] Unwired Labs. *OpenCellID* (cit. on p. 58).

[161] H. Welte. *OpenBSC - Running your own GSM network*. 2009 (cit. on pp. 3, 27).

[162] WhatsApp Inc. *WhatsApp Messenger*. URL: http://www.whatsapp.com (cit. on p. 35).

[163] *WikiDevi*. https://wikidevi.com/wiki/Main_Page (cit. on pp. 79, 80).

[164] Wikipedia. *Software Defined Radio*. https://en.wikipedia.org/wiki/Software-defined_radio (cit. on p. 29).

[165] *Wireshark - network protocol analyzer*. URL: https://www.wireshark.org/ (cit. on p. 32).

[166] PC/SC Workgroup. *PC/SC Workgroup Specifications* (cit. on p. 31).

[167] XDA-Developers. *LG G3 Field Test Mode*. URL: http://forum.xda-developers.com/lg-g3/general/lg-g3-field-test-mode-how-to-check-lte-t3128275 (cit. on p. 50).

[168] Hojoon Yang, Sangwook Bae, Mincheol Son, et al. „Hiding in Plain Signal: Physical Signal Overshadowing Attack on LTE“. In: *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 55–72 (cit. on p. 24).