UHASSELT
KNOWLEDGE IN ACTION

Maastricht University

Faculteit Wetenschappen
School voor Informatietechnologie

master in de informatica

Masterthesis

5G Man-in-the-Middle Research Tool

Jelle Beerts

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :
Prof. dr. Peter QUAX

COPROMOTOR :
Prof. dr. Pieter ROBYNS

2024
2025

De transnationale Universiteit Limburg is een uniek samenwerkingsverband van twee universiteiten in twee landen: de Universiteit Hasselt en Maastricht University.

UHASSELT
KNOWLEDGE IN ACTION

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

Maastricht University

Faculteit Wetenschappen
School voor Informatietechnologie

master in de informatica

Masterthesis

5G Man-in-the-Middle Research Tool

Jelle Beerts

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :
Prof. dr. Peter QUAX

COPROMOTOR :
Prof. dr. Pieter ROBYNS

Hasselt University

Thesis submitted for obtaining a master's degree in computer science

5G Man-in-the-Middle Research Tool

Author:

Beerts Jelle

Promotor:

Prof. Dr. Peter Quax

Co-promotor:

Prof. Dr. Pieter Robyns

Academic year 2024-2025

Acknowledgements

First and foremost, I want to thank Prof. Dr. Pieter Robyns, for his guidance and feedback during the entire process of delivering this result. I also want to thank my promotor, Prof. Dr. Peter Quax, for his valuable insights and support. I am grateful for the support I received from my friends, helping me to stay motivated in tough times. I thank my family, especially my parents, for providing me with the opportunity to achieve my goals. Finally, a special thanks to Lila for always being there for me.

Abstract

5G is the latest generation in a long-standing line of wireless communication standards and has rapidly become a popular way for users to access the Internet on their mobile devices. 5G has improved on its predecessors in many ways to enhance its connectivity experience, and security is one of the areas that 5G attempted to improve upon. Even so, there are still ways to attack this standard, leading to privacy concerns, service issues, and more. A key requirement of many of these attacks is a Man-In-The-Middle (MITM) position in which an attacker is able to place themselves in between the user on one side, and the network on the other, forwarding and possibly manipulating the communication between the two parties.

As this scenario is often a requirement to execute an attack, researching vulnerabilities that depend on the MITM position first requires implementing the MITM itself. This process often results in closed-source solutions that seem to be tailored towards specifically testing the given attack, reducing their usefulness in other projects, and creating unnecessary duplicate work. This thesis provides a general-purpose and configurable solution that provides the researcher with a tool to immediately start testing the actual vulnerability, without having to implement the MITM setup themselves. We evaluate our approach by executing different known vulnerabilities using our tool, showing that the tool would have been usable for those attacks and can be used for similar attacks.

Nederlandstalige Samenvatting

Opzet

5G is de meest recente generatie van de verzameling draadloze communicatiestandaarden die ons toelaten in het dagelijkse leven om overal waar we ons bevinden met het internet te verbinden. Vergeleken met zijn voorgangers probeert 5G op drie verschillende manieren een verbetering te brengen. Ten eerste was het de bedoeling om de mobiele breedband ervaring te verbeteren: men wou een sneller netwerk voorzien met meer capaciteit. Ten tweede was het de bedoeling om de vertraging die men oploopt tijdens het versturen van signalen te verlagen en de betrouwbaarheid van de connectie te verhogen. Ten slotte was het de bedoeling om op extreem grote schaal communicatie te ondersteunen: door ondersteuning te bieden voor Internet of Things (IoT) toestellen zoals sensoren, rekening houdend met stricte eisen rond bijvoorbeeld batterijverbruik.

Naast deze hoofddoelen, is er ook gefocust op het verbeteren van de beveiliging van 5G. Zo is er bijvoorbeeld ingezet op het verhogen van het niveau van privacy dat een gebruiker heeft in het netwerk aan de hand van een beschermde identiteit. Desondanks deze pogingen tot het verhogen van de beveiliging van 5G, is het nog altijd mogelijk om deze netwerken aan te vallen. Er wordt voortdurend onderzoek gedaan naar manieren om 5G aan te vallen, en een opvallend kenmerk van veel van deze aanvallen is dat ze uitgaan van een zogenaamde Man-In-The-Middle (MITM) positie. Deze opstelling is een scenario waarin een aanvaller erin slaagt om de netwerkgebruiker ervan te overtuigen met infrastructuur van de aanvaller te verbinden in plaats van met het echte netwerk. De aanvaller verbindt op zijn beurt met het netwerk, en doet alsof hij de netwerkgebruiker is. Het resultaat is dan dat alle communicatie tussen de netwerkgebruiker en het netwerk via de aanvaller verloopt, wat hem toelaat om de berichten te bekijken en te manipuleren. Op Figuur 1 is dergelijk scenario afgebeeld.

De MITM opstelling is dermate krachtig dat hij vaak ook een vereiste is om een bepaalde aanval uit te voeren. Hierdoor moeten onderzoekers vaak een MITM ontwikkelen vooraleer ze hun eigenlijke aanval kunnen testen, met als gevolg dat deze implementaties vaak erg gericht zijn op wat er precies nodig is om de aanval te bereiken en ook niet publiek beschikbaar zijn. Het opzet van deze thesis is om een implementatie te maken die dit probleem aanpakt. We

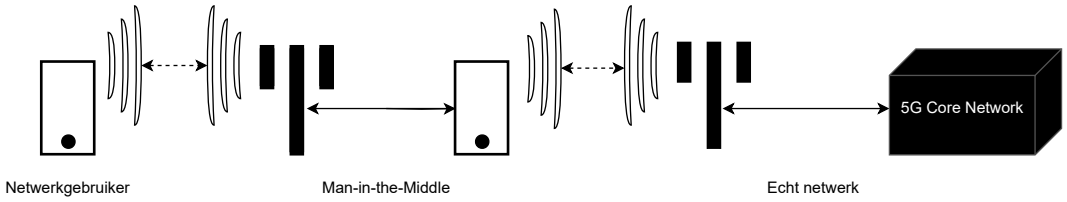

Figure 1: Een MITM opstelling in een draadloos netwerk zoals 5G.

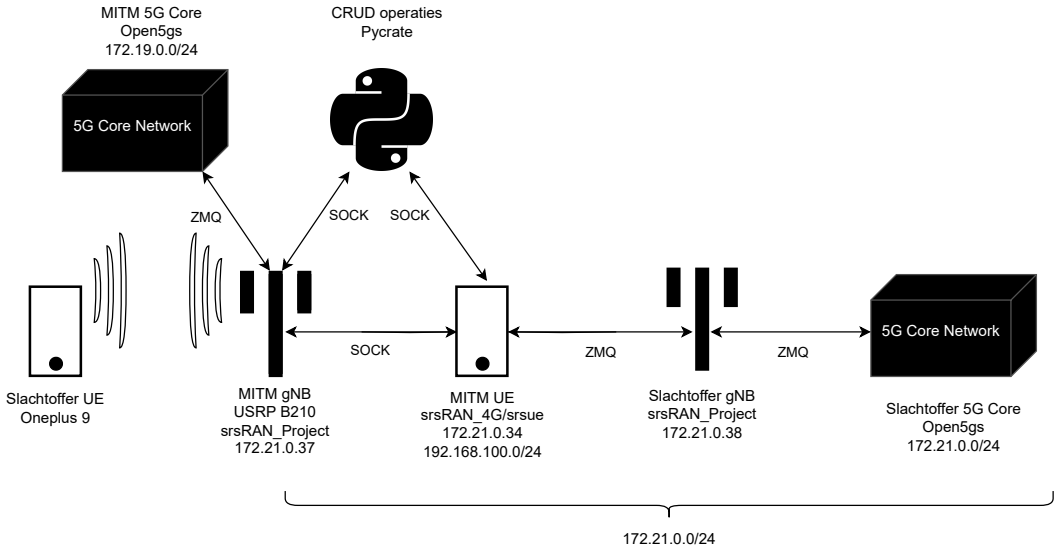

Figure 2: De opstelling geïmplementeerd in dit onderzoek.

voorzien een generieke en vrij instelbare MITM tool die toelaat om meteen over te gaan tot de eigenlijke aanval te implementeren en te testen zonder dat men tijd verliest aan het maken van een MITM. Verder toetsen we ook de toepasbaarheid van deze tool door reeds ontdekte aanvallen die steunen op een MITM uit te voeren met onze eigen tool.

We definiëren voor dit werk vier onderzoeksvragen, namelijk: “is het mogelijk om met bestaande software een generieke en aanstuurbare MITM te implementeren?”, “kunnen we met deze generieke implementatie bestaande aanvallen implementeren?”, “zijn ontdekte aanvallen nog steeds een probleem?” en ten slotte: “kan een MITM in eender welke netwerkomgeving in gebruik genomen worden?”.

Implementatie

Overzicht

In de uitgewerkte opstelling zitten vier componenten:

- De User Equipment (UE): een eindgebruikerstoestel dat kan communiceren met het netwerk (telefoon, smartwatch, etc.).
- De 5G Node B (gNB): de netwerkinfrastructuur waarmee de UE verbindt wanneer het met het netwerk wil communiceren.
- De 5G Core (5GC): het achterliggende core netwerk dat alle logica, communicatie, etc. voorziet samen met de UE.
- Python modules: deze stukken code laten ons toe om berichten in het netwerk te manipuleren, aan te maken, etc.

Om de tool te maken hebben we gebruikgemaakt van bestaande software: srsRAN [Sys25m] voor de gNB, srsUE [Sys25i] voor de UE en Open5GS [Ope25a] voor het core netwerk. Verder hebben we gebruikgemaakt van extra tools zoals Pycrate [Pyc25] om pakketten aan te passen, aan te maken, etc. Dit had als resultaat de opstelling getoond in Figuur 2.

De tool is in staat om zowel passief als actief te zijn in het netwerk, wat betekent dat de tool gebruikt kan worden om berichten die beide partijen sturen uit te lezen of gewoon door te

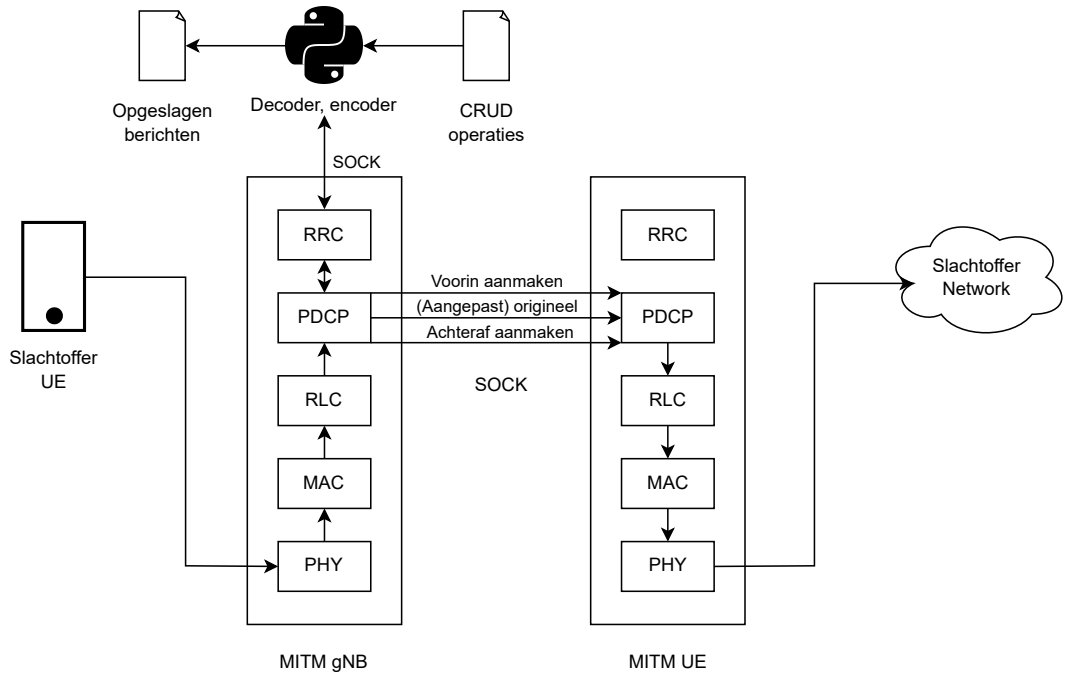

Figure 3: Het pad dat een bericht doorloopt in het geval van een actieve MITM instelling.

sturen. In dat geval heeft de MITM geen effect op de communicatie en kan hij gebruikt worden voor bv. privacy-gerichte aanvallen. Indien men wenst de MITM een meer actieve rol te geven in het netwerk kan dit door aan de hand van configuratie bestanden vier soorten acties aan te geven: de Create, Retrieve, Update, Delete (CRUD) operaties. Deze operaties verwijzen naar het aanmaken van berichten, het opvragen van berichten, het aanpassen van berichten en het verwijderen van berichten.

Om een operatie te definiëren, gebruiken we een JSON formaat waarin de gebruiker duidelijk moet maken welke soort operatie hij wil uitvoeren, op welk soort bericht dit moet uitgevoerd worden en wanneer in de communicatie dit moet gebeuren. De MITM zal deze commando's dan gebruiken om de communicatie actief te manipuleren met aandacht voor het bewaren van de communicatie. Er zijn mechanismen aanwezig in de protocollen van 5G die een derde partij beletten om zomaar berichten aan te passen en dergelijke. De tool omzeilt deze zolang als mogelijk, om de aanvaller zo veel mogelijk opties te geven. Zodra dit niet meer mogelijk is, zal de communicatie falen.

Werking

De geïmplementeerde tool werkt door binnenkomende berichten in beide richtingen door te sturen naar hun eigenlijke ontvanger. Alvorens dit te doen, wordt er gekeken of er een bepaalde operatie gedefiniëerd is voor dit bericht. Zoals afgebeeld op Figuur 3, doorloopt een bericht van bijvoorbeeld de UE naar het netwerk de normale 5G protocol stack tot op het niveau van het Packet Data Convergence Protocol (PDCP). Op dat moment stopt de MITM de uitvoering van de normale srsRAN code, en wordt de code vervangen met die wat de MITM functionaliteiten voorziet. Er wordt nagekeken of dit bericht moet opgeslagen worden (`RETRIEVE`), daarna wordt er gekeken of het bericht moet tegengehouden worden (`DELETE`), daarna wordt er gekeken of er berichten aangemaakt moeten worden die voor het origineel verstuurd moeten worden (`CREATE`), daarna wordt er gekeken of er een aanpassing gedaan moet worden aan het bericht (`UPDATE`) en ten slotte wordt er gekeken of er berichten aangemaakt moeten worden na het versturen van

het origineel (`CREATE`).

Met deze implementatie beantwoorden we de eerste onderzoeksvraag: het is wel degelijk mogelijk een generieke MITM opstelling te maken met bestaande software tools.

Experimenten

Om te bewijzen dat onze tool wel degelijk bruikbaar zou zijn om onderzoek uit te voeren, hebben we een selectie gemaakt van onderzoek dat in het verleden gedaan is waarin een MITM een vereiste was. We hebben voor elke aanval in deze selectie een configuratie proberen zoeken die het doel van de aanval kon bereiken, wat dan aantoont dat onze tool ook gebruikt had kunnen worden.

Naast onderzoeken hoe bruikbaar onze tool zou zijn voor het ontdekken van deze aanvallen, hebben we ook getest of de aanvallen nog steeds een probleem vormen na hun ontdekking. Een positief resultaat betekent dat we een gelijke uitkomst kregen als het voorgaande onderzoek, en bewijst dat de geteste aanval nog steeds een probleem is. In het geval van een negatief resultaat kunnen we enkel stellen dat met de componenten die getest zijn de aanval niet meer zou lukken. Dit sluit echter niet uit dat het nog steeds mogelijk zou zijn met andere toestellen.

De resultaten van onze studie worden getoond in Tabel 1. We zien dat onze tool in staat is om bijna alle geselecteerde aanvallen uit te voeren. Dit komt doordat dit allemaal aanvallen zijn die gebruikmaken van de vier ondersteunde CRUD operaties. De laatste aanval, NReplay [FSP24] vereist ook nog dat er een XOR-operatie wordt uitgevoerd op een bericht, wat onze tool niet ondersteunt. Een aanval zoals NReplay toont dus aan dat er ook scenario's zijn waarin een gespecialiseerde tool wel degelijk een vereiste is. Met deze resultaten is het mogelijk om onderzoeksvragen 2 en 3 te beantwoorden: een generieke tool zoals geïmplementeerd in dit onderzoek is in staat om gekende aanvallen uit te voeren, en er zijn nog steeds aanvallen die een probleem vormen na hun ontdekking.

Aanval	MITM Ondersteuning	Resultaat	Opmerking
SUCI catcher [Chl+21]	Volledig	Positief	
ETWS spoofing [BP22]	Volledig	Positief	
Registration reject: 5GS [Kar+23]	Volledig	Positief	
Identity request: SUCI [Kar+23]	Volledig	Positief	
Identity request: IMEI [Kar+23]	Volledig	Positief	Getest op srsUE.
		Negatief	Getest op smartphone.
Registration reject: N1 [Kar+23]	Volledig	Negatief	
RRC release w/ redirect [Kar+23]	Volledig	Negatief	
NReplay [FSP24]	Enkel berichten	Geen resultaat	XOR Operator niet ondersteund.

Table 1: Een overzicht van alle geteste reeds bekende aanvallen, hoe goed ze te implementeren vielen met onze tool en of het test resultaat al dan niet positief was, inclusief een nota over de test over hoe de test gedaan was, of waarom ze niet gedaan kon worden.

Praktisch

Een MITM zou op verschillende manieren gebruikt kunnen worden in testen:

1. Volledig gesimuleerd: het volledige netwerk wordt gesimuleerd op één toestel, met artificiële connecties. Elk component is volledig geïmplementeerd in software.

2. Deels draadloos: de UE van het slachtoffer is een commercieel aankoopbaar toestel dat draadloos verbindt met het netwerk dat deels opgebouwd is uit fysieke hardware.

3. Volledig realistisch: alle toestellen werken draadloos en er wordt een verbinding opgesteld met een commercieel netwerk aan de hand van fysieke hardware.

In onze testen was het mogelijk om versie 1 en 2 in gebruik te nemen. We ondervonden geen moeilijkheden met de volledig gesimuleerde opstelling, noch met een deels realistische connectie met een commercieel toestel. Deze twee opstellingen zijn dus volledig bruikbaar. De laatste opstelling in gebruik nemen lukte ons echter niet doordat er op het moment van schrijven geen aanbod is van 5G Standalone (SA) in onze directe omgeving. Concreet betekent dit dat er geen commercieel netwerk aanwezig is op de plek van testen waarmee onze setup effectief zou kunnen werken. Het alternatief, 5G Non-Standalone (NSA), is wel aanwezig, maar dit wordt niet volledig ondersteund door de componenten die gebruikt zijn om de tool te implementeren. Op dit moment is dit probleem ook niet uniek aan onze omgeving, wat maakt dat de setup volledig verbinden met een commercieel netwerk nog werk zou vergen om ook dit soort netwerken te ondersteunen.

Conclusie

Met het geleverde werk voorzien we een generieke en aanstuurbare onderzoekstool die gebruikt kan worden in het onderzoek naar aanvallen op 5G. We hebben ons uitdrukkelijk gericht op een tool die herbruikbaar is zonder specialisaties, met als gevolg dat niet alle aanvallen uitvoerbaar zullen zijn. Onze experimenten met bestaande aanvallen geven echter wel aan dat een groot deel van de aanvallen wel uitvoerbaar zouden zijn met onze tool, wat wijst op een goede bruikbaarheid ervan. Onze ervaringen met het in gebruik nemen van de tool leren dat afhankelijk van welke versie men wenst te gebruiken, er al dan niet gekeken moet worden naar andere oplossingen. Commerciële netwerken die nog geen state of the art 5G SA voorzien worden niet ondersteund door deze tool, maar we voorzien een alternatief dat toch toelaat om testen uit te voeren op commercieel aankoopbare toestellen. Met deze resultaten is het opzet van de thesis bereikt, en hebben we onze onderzoeksvragen beantwoord.

Contents

Chapter 1

Introduction

Cellular communication has become an integral part of our lives. We've seen multiple generations of mobile networks starting with 1G in the 1980s, up to the present day with 5G, which has officially become the fastest-growing of these technologies. GSMA reports 5G to have reached over 1.5 billion connections since the end of 2023, outpacing its predecessors [AB24]. 5G is the successor to 4G, which itself has enjoyed widespread success. Combining 5G's connection numbers with those of 4G, GSMA reports that 80% of mobile Internet subscribers are using a 4G or 5G connection, showcasing how relevant the last two mobile network generations have really been. This trend will continue as well, with about 4.6 billion users accounting for 57% of the population using a mobile device to access the Internet [SB24].

The fast adoption of 5G comes as no surprise given the improvements it offers over 4G to both consumers and enterprises. The user experience was improved according to the three 5G design use cases: Enhanced Mobile Broadband (eMBB), Ultra Reliable Low Latency Communication (URLLC), massive Machine-Type Communications (mMTC) [Int19], and its security was enhanced, responding to the many vulnerabilities found in previous generations [GSM25].

Despite all these improvements, there remain vulnerabilities that can be found and exploited in 5G. Vulnerabilities have many possible origins: they can be introduced by incorrect interpretation of the specifications/unclear specifications as seen in [Kli+23] or by abusing the semantics of valid messages, for example [Chl+21]. A subdomain of vulnerabilities requires the attacker to be in the so-called Man-In-The-Middle (MITM) position. In those scenarios, the attacker is in a powerful position where they can intercept and/or partake in the communications between two parties [Mal19].

In this thesis, we will focus on such MITM attacks in 5G, with a specific eye for implementing a configurable passive and/or active MITM setup. Our primary research question is: "can we use existing open-source solutions to implement a configurable 5G MITM tool?". Based on this question, we can further research related and more specific questions: "would such a tool be usable to implement known vulnerabilities?" and "are there vulnerabilities that are still a problem after their discovery?" and finally "how deployable is such a MITM setup in the real world?".

This thesis will answer these questions by executing practical experiments using the implementation we made as an answer to our first research question. Each experiment will focus on a specific kind of known attack, described in detail along with the required background knowledge throughout the thesis. Of course, it would be impossible to implement each known attack on 5G due to timing constraints, so we selected the ones we found most interesting to show the tool's capabilities. Experiments that showed the tool lacks the required customizability to implement them will also be shown to provide an open and honest look at the implementation we made.

We start the thesis by looking at the 5G System in general, providing a thorough look at the design goals of 5G and how well they were achieved in practice. Then, Chapters 3 and 4 will treat all background knowledge required to understand how to implement a MITM for 5G networks. Chapter 4 also details known attacks on 5G, which will be demonstrated using our tool later in Chapter 5. Chapter 5 describes the tool implemented in this thesis and how it can be used to implement existing attacks to demonstrate its usability as a research tool. Finally, we will form a conclusion regarding the work that was done and provide options for future work in Chapter 6.

Chapter 2

5G System Overview

3GPP first defined 5G in Release 15 [3GP19], with, as of writing this thesis, the most recent fully developed standard being Release 18 [3GP25a]. As a stepping stone for the research done in this thesis, this chapter will focus on how the 5G System (5GS) is built, the goals of the overall design, as well as whether or not it has accomplished these goals. This high-level overview of the 5GS will be followed by an in-depth look at how 5G works with a focus on security.

2.1 5G Use Cases

The design of 5G is heavily governed by the desire to succeed in three key areas. These are the so-called 5G use cases [DPS23]:

- Enhanced Mobile Broadband (eMBB)
- Ultra Reliable Low Latency Communication (URLLC)
- massive Machine-Type Communications (mMTC)

2.1.1 Enhanced Mobile Broadband

When thinking about how a mobile network can be improved compared to its predecessor, improvements such as faster data rates immediately come to mind. eMBB is the use case that focuses on these kinds of logical improvements upon the provided services by 4G. The focus of eMBB is on further improving the user experience by increasing end-user data rates, capacity, etc., compared to previous generations [DPS23]. In comparison with 4G, the following IMT-2020 5G requirements were set [Int19]:

- 10-100x Data rates: a peak downlink rate of 20Gb/s, peak uplink of 10Gb/s and a user experience downlink rate of 100 Mb/s and 50Mb/s uplink.
- 1000x more capacity: 10 Mbit/s/m^2 connecting 1,000,000 devices per km^2, which is 10-100x more connected devices.

However, due to the already extensive use of the low-band spectrum, these capacity and bandwidth improvements had to come from a 5G deployment in all bands to provide the necessary frequency allocations. In reality, this deployment has been slow, with the European Union (EU) assigning three pioneer bands in 2019 meant to show the promise of 5G and entice further investment and expansions into the new technology [Ook23]. These pioneer bands are the following:

- 700MHz for long-range applications
- 3.6GHz for a large-scale, common implementation of 5G

- 26GHz for short-range applications

It took until October 2023 for all EU member states to assign at least one pioneer band, with Poland being the last [Obs23].

With the lack of spectrum allocation comes a smaller than anticipated bandwidth, and with this limited bandwidth comes a disappointing performance effect: the median 5G speeds in Europe have actually decreased over time rather than increased. This is mainly due to network congestion since we have more and more 5G systems accessing a stagnant spectrum allocation, which in turn is caused by an unwillingness to move to a higher spectrum (due to the associated costs, for example) or free up lower-band frequencies and thus severely limiting the available bandwidth for 5G [Ook23].

To keep up with the fast-growing 5G market, the EU has created a so-called '5G strategy in the digital decade' [Com25], which sets targets to be reached by 2030 surrounding 5G. Despite this strategy, the EU is behind on its goals compared to others, such as the United Arab Emirates and South Korea. Progress on targets such as availability is still being made, however and availability is gradually going up [Eve24].

Throughout the EU, mid-band has been the most popular [Ook23]. Although each country has allocated at least one pioneer band, the number of countries assigning spectrum across all three pioneer bands has remained low (8 in February 2023). In that same year, 17 countries allocated spectrum across both low- and mid-band frequencies, showing how unpopular high-band frequencies remain. Finally, as of February 2023, just eight countries have allocated millimeter Wave (mmWave) spectrum, which is staggeringly low considering the mmWave spectrum was meant to be one of the great successes of 5G [Ook23]. As for the low-band frequencies, operators are choosing to shut down legacy networks in favor of 4G Long Term Evolution (LTE) or 5G New Radio (NR) [BIP24]. The primary 'victim' of this phasing-out process is 3G. 2G is harder to phase out as it is often still needed to support M2M/IoT use cases [Ook23].

2.1.2 Ultra Reliable Low Latency Communication

Latency and reliability should be a key focus of wireless networks, as numerous applications such as traffic safety in Vehicle-to-Vehicle (V2V) use cases and factory automation require the network they utilize to deliver data reliably with as little delay as possible [DPS23]. In the IMT-2020 5G requirements, we can see the following [Int19]:

- 1ms-4ms latency for the User-plane (UP), 20ms for the Control-plane (CP).
- perception of 99.999% availability, which again refers to the high capacity of 5G but also the reliability requirement of a 99.999% success probability successfully transmitting a Packet Data Unit (PDU) under certain circumstances.

If we define 5G availability as the number of users that spend most of their time accessing a 5G network on their 5G-capable device, we can say that the perception of 99.999% availability has not been achieved yet. As of February 2023, the U.S. has the highest level of 5G availability, topping the charts at 56%. In Europe, availability is even further from 99.999%, only exceeding 40% in Cyprus, Switzerland, and Denmark [Ook23].

A case study done in [Yai23] saw that 5G is more supportive of online gaming due to its faster download and upload speeds compared to 4G, but can still be underwhelming latency-wise. Mobile gaming is one of the domains targeted by URLLC, due to its strict latency requirements that weren't supported on previous mobile network generations. The observed latency values in the case study ranged from 72.01ms to 115.46ms, which means some multiplayer games could suffer from a deteriorated experience. Sub-100-ms latency was considered acceptable, and the observed values all passed this bar except for one, but the observed difference between 5G and 4G was relatively modest compared to the increased download and upload speeds.

2.1.3 Massive Machine-Type Communications

Internet of Things (IoT) devices, such as sensors for example, are devices that get deployed at a massive scale and typically have strict performance requirements, such as low energy consumption, to increase their battery life. These devices often do not require high data rates, nor low latencies, meaning that they do not fit in the set of devices targeted by the previously mentioned eMBB and URLLC use cases. To cater to these types of devices, the mMTC use case was designed, focusing on improving the number of devices that can be connected to a network, as well as the efficiency of the connection [DPS23]. In the IMT-2020 5G requirements, we can see the following [Int19]:

- 1000x more capacity: 10 Mbit/s/m^2 connecting 1,000,000 devices per km^2, which is 10-100x more connected devices.
- high energy efficiency: 10 year battery life for sensors, 90% reduction in energy use leading to a high focus on spectral efficiency: an average of 3.3-9 bits/s/Hz in downlink and 1.6-6.75bits/s/Hz in uplink.

As already mentioned, the bandwidth dedicated to 5G is still limited. Due to this limit, the capacity also becomes limited since a given amount of bandwidth can only handle so many users. The efficiency of the capacity part of this use case, therefore, is directly dependent on the eMBB use case, relying on its bandwidth improvements.

2.2 Spectrum Use

The 5G Radio Access Network (RAN) uses the wireless spectrum, which is divided into different frequency bands, each with its own characteristics and designated type of communication or services. Each band can be further divided into carriers. The total spectrum covered by 5G NR ranges from sub-1GHz, 1GHz to 6GHz and above 24GHz, also known as mmWave starting at 30GHz [Int19]. We can classify the frequency range into three main areas:

- Low-band (sub-1Ghz): The low-band is the lowest operating range for 5G, using lower frequencies that have better propagation characteristics than their higher frequency alternatives (see Figure 2.1). The better the propagation characteristics of a signal, the better it can penetrate obstacles on its path, such as buildings, which makes them interesting for large-scale deployment. As an example, the difference in path loss between 700 MHz and 1800 MHz is 13.4 dB, which corresponds to a 2.6 times increase in the distance the signal can cover (in case of an unobstructed path) [GL22]. As one might gather from the example, path loss means "the reduction in signal strength as it propagates through the wireless channel" [IOO23], and the higher the frequency, the bigger the path loss due to, for example, atmospheric absorption [IOO23]. Due to the lower path loss of lower frequencies, they are more cost-effective at covering a wide area, making this deployment especially interesting for rural areas [GL22]. The issue with this band is the very limited amount of bandwidth it still has available due to other allocations [LG21], causing a maximum of 20 MHz channel bandwidth [DPS23].
- Mid-band (1GHz-6GHz): The mid-band sits between the coverage benefits of the low-band and the capacity benefits of the high-band. Offering a bit of both, the mid-band can be used for most forms of deployment [Int19]. Mid-band is considered to have a city-wide coverage area, with higher capacity than low-band [SC22], especially in the upper mid-bands (3.3-4.2, 4.5-5, 5.925-7.125 GHz), there is a capacity for much wider bandwidth compared to the low-band [LG21], reaching channel bandwidths up to 100 MHz [DPS23].
- High-band (Above 24GHz): Due to the already mentioned propagation properties of the different frequency ranges, the high-band spectrum becomes hard to use for wide-area coverage. Due to its large channel bandwidth, though (up to 400MHz), the high band should still be utilized in cases where high capacity and data rates are required. Examples

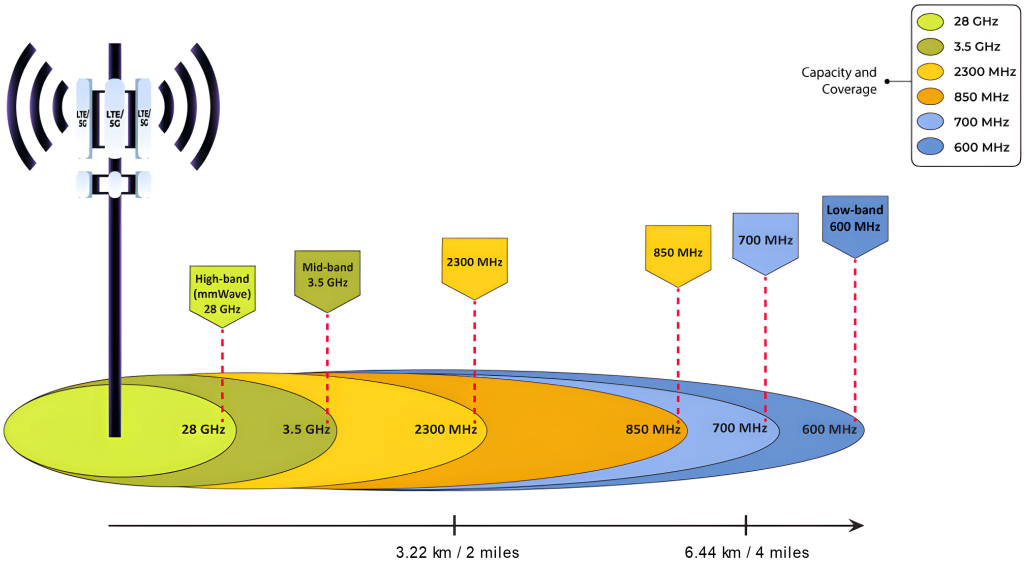

Figure 2.1: The 5G operating frequencies and their respective (theoretical) signal range, from [NYB23].

of those deployment scenarios are targeted hotspots in which users are densely packed into a relatively small area [DPS23].

2.3 The 5G System

2.3.1 Composition

As of writing this thesis, the latest 5GS is standardized in [3GP24b] (release 18). Schematically, the 5GS is similar to previous generations. In its simplest form, a typical 5G system looks as depicted in Figure 2.2.

This schematic conforms with the definition of the 5GS given in the specification: "the 3GPP system consisting of the 5G Access Network (AN), 5G Core (5GC), and User Equipment (UE)" [3GP24b]. The different components in the 5GS will now be explained in further detail.

- UE: In its most abstract form, a UE is "a device composed of a Mobile Station (MS) and a Universal Subscriber Identity Module (USIM)" [Sul22]. A MS comprises all technology needed for wireless communication with a mobile network, such as antennae. The USIM contains the information enabling users to access and authenticate themselves to the mobile network as valid subscribers [DPS23]. In our daily lives, we see UE's everywhere: from smartphones and smartwatches to specialized IoT devices to even untethered Extended Reality (XR) glasses [Ame23].
- 5GC: At the basis of the 5GC lies a Service-based Architecture (SBA) that implements the standard core network functionalities in a more modular manner. These services and the 5GC in general are further explained in 3.1. For now, it suffices to describe the 5GC as a black box that is responsible for essential network functionalities such as confidentiality and integrity protection, user identification and authentication, and end-to-end connection setup, relying on the 5G Node B (gNB) to provide the radio access to the network [DPS23].
- gNB: The gNB handles the communication between the UE and the 5GC by forwarding traffic between the two parties. The gNB provides the NR-Uu interface to 5G capable de-

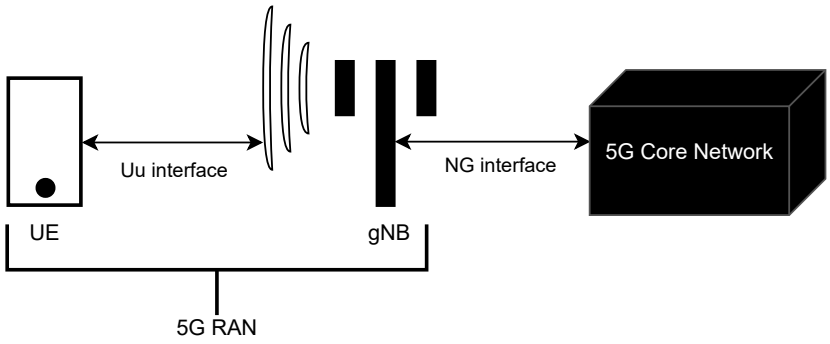

Figure 2.2: A basic 5G System (5GS), adapted from [Sul22].

vices and uses the Next Generation (NG) interface to connect to the 5GC [Sul22]. Besides providing connectivity between the UE and 5GC, the gNB also handles the scheduling of the data transmissions of those parties in both uplink (UE to 5GC) and downlink (5GC to UE) for all UEs in its cell range. The gNB also collects measurement data of the UEs and Quality of Service (QoS) measurements to make sure that the correct UE's traffic gets prioritized if necessary [Kor23].

- RAN: The RAN handles all radio-related functionalities, relying on the 5GC to provide the network functionalities. The RAN is connected with the 5GC through either a gNB serving 5G NR devices using the NR protocols or through a Next Generation Evolved NodeB (ng-eNB) that uses the 4G LTE air interface to connect devices to the network [DPS23].

2.3.2 5G Deployment Options

After working on the 5G network architecture, 3GPP created several architecture options that described possible 5G deployments. These options were based on three important decisions surrounding 5G [Rom+19]:

- What, if any, support for LTE would be provided in the new architecture.
- How LTE and NR access could be combined.
- How LTE and its Evolved Packet Core (EPC) could be of use in specifying an alternative 5G architecture.

Combining LTE and NR Radio Access Technology (RAT) was done through having one of the RAT technologies act as the RAT with the higher geographical coverage. This 'dominant' RAT is responsible for handling the signaling surrounding connection control (i.e., connection setup, authentication, etc.). The other RAT is responsible for boosting user data traffic capacity, enhancing their experience overall [Rom+19]. If both LTE and NR are deployed to aid each other, the term 5G Non-Standalone (NSA) is used.

5G Non-Standalone

As launching 5G in its pure form would require significant investments and infrastructure changes, these concerns raised by the telecom industry were a key driver for the development of 5G NSA, requesting a less drastic way to provide a first way to offer 5G services [Rom+19]. In response, NSA was developed, combining both 5G and 4G in different ways (so-called 'options'), with one of the two generations taking responsibility for a specific part of the architecture (connection control signaling or user data traffic). Due to LTE being used in these options, the existing infrastructure could remain in use, maximizing the value of the investments made in this

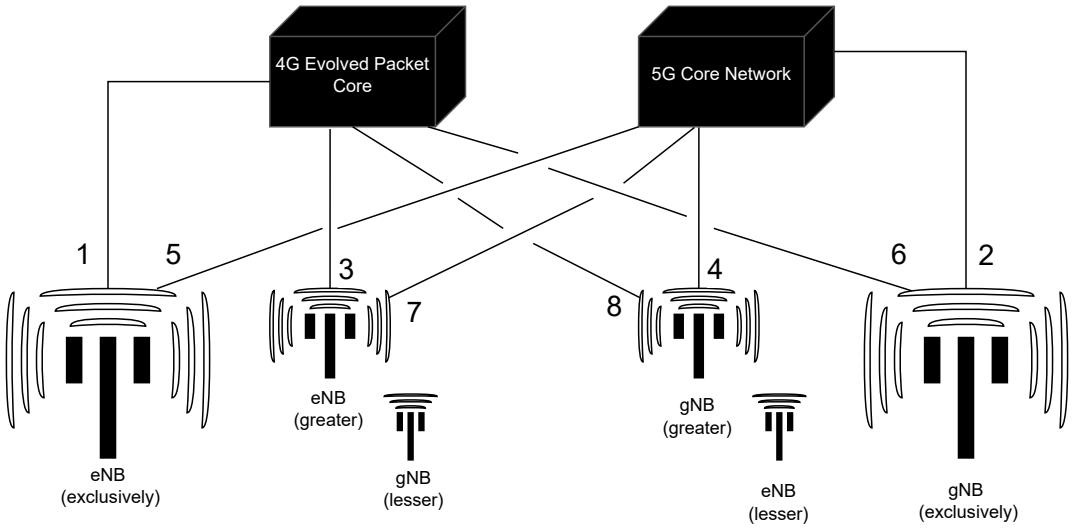

Figure 2.3: All eight 5G NSA deployment options, from [Rom+19].

infrastructure and speeding up the deployment of 5G, with the possibility to later implement the full upgrade to 5G Standalone (SA) [GSM20].

As should be obvious, the main requirement of this architecture is an existing LTE coverage. The main drawback of using NSA is that two of the three 5G use cases mentioned in Section 2.1 are not supported. Due to LTE being present in the architecture, only LTE's supported network features are provided [Rom+19]. New features that are introduced in the 5GC will therefore not be available, generally only achieving improvements related to eMBB as the bandwidth-related benefits of NR would still be available.

The four different deployment options of this combination of LTE and NR are as follows [Rom+19]:

- LTE control signaling and data traffic (no gNB).
- NR control signaling and data traffic (no Evolved Node B (eNB)).
- LTE control signaling, NR data traffic. LTE has the larger coverage.
- NR control signaling, LTE data traffic. NR has the larger coverage.

Adding the two possible cores (EPC and 5GC) to this set of four options, we get eight options total, which are displayed in Figure 2.3. As shown in the figure, there are three subparts that together denote one option. Using dual connectivity makes an option fall under the NSA umbrella, whereas using only one RAT makes an option SA in the case of 5G. Besides dual connectivity, the RAT used by the node for control signaling is a factor. The node that does not serve for control signaling will be used to boost user data traffic capacity and utilize the other RAT. Finally, each RAT configuration has two possible core networks it can be connected to: the 4G EPC and the 5GC. Both setups denote a different option [GSM19].

It should be noted that the term 'NSA' also often refers specifically to option 3, and 'SA' is synonymous with option 2, as this is the only option with exclusive NR deployment. In this thesis, we will utilize this convention as well, unless specifically focusing on different deployment options.

Out of all the options, some were not fleshed out during the design phase. Options 6 and 8 were abandoned as connecting the 5G gNB to the 4G EPC would impose too many limitations on the features of NR to justify the investment. Option 1 refers to the original LTE architecture

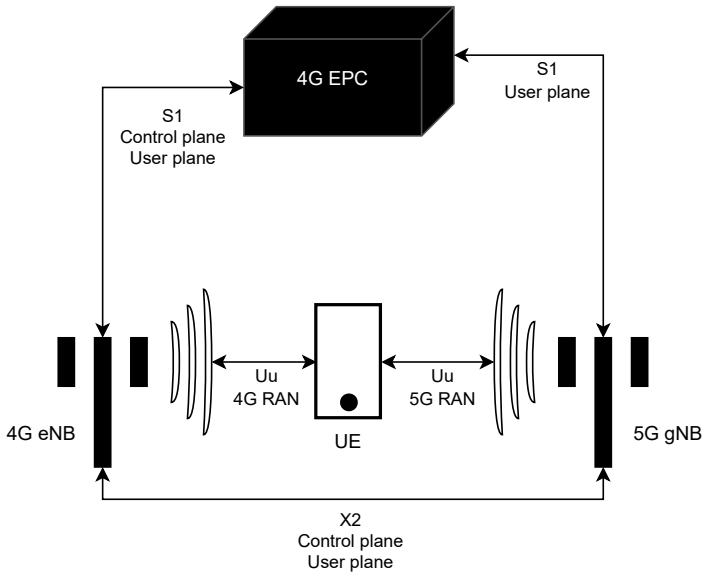

Figure 2.4: 5G NSA option 3 deployment [Rom+19].

(1 eNB connected to the 4G EPC) and is therefore not relevant for 5G design. The remaining options were therefore 2, 3, 4, 5 and 7, out of which 2 (full NR) and 3 (LTE the dominant RAN technology, with NR as supplement) were favored due to their higher perceived market value [Rom+19].

Option 3 is the NSA architecture that is shown in Figure 2.4 and was the first 5G network architecture to be commercially deployed. As previously mentioned, this rapid deployment was made possible due to its reuse of existing LTE infrastructure, requiring only a deployment of 5G gNB's for boosting data traffic capacity [Rom+19].

Despite the popularity of option 3, other NSA deployments also have their merits. Those go beyond the scope of this thesis due to their lower popularity, however.

5G Standalone

5G SA comprises the full 5G vision, making exclusive use of 5G RAN and the 5GC. In Figure 2.3, this denotes option 2. If implemented correctly, it provides all 5G use case benefits and supports all value enablers, such as edge computing. When comparing 5G to 4G and mentioning the benefits of 5G, 5G SA is always the basis for comparison. Figure 2.2 in fact denoted a 5G SA deployment.

Chapter 3

5G Architecture

In this chapter, we will take a more detailed look at the components that make up the 5GC and how it was built with regard to security. We will also look at the large set of identifiers a UE can have in the 5GS in order to lay the foundations for a deeper dive into the protocols that make up the 5G protocol stack in the next chapter.

3.1 5G Core

The 5GC introduced two large enhancements compared to its predecessor: SBA and support for network slicing [DPS23]. A CP/UP split was also added, but this is also present in the 4G EPC since Release 14 [SLY17]. The design of the 5GC architecture was oriented towards providing an access-independent interface instead of providing backwards compatibility with the previous generations. This means that instead of creating a new interface between the core and RAN meant for that specific generation as had been the case up until then, the 5GC provides two interfaces, N2 and N3, that are meant to be as generic and future-proof as possible, allowing future RAN technologies to utilize these interfaces as well. For example, the LTE specifications were complemented to support the N2/N3 interfaces, allowing it to utilize the 5GC as well [Rom+19].

The major difference between the 5GC and the EPC is the SBA. In the SBA, all functionalities provided by the core are split up into different services provided by different Network Function (NF)s. These NFs interact with each other using the HTTP/2 Representational State Transfer (REST) paradigm, in which one of the NFs is a service producer and the other one is a service consumer [Rom+19].

Since the 5GC is built using services that discover and communicate with one another, some services can be deemed essential, whereas others are optional. Without the essential NFs, it is impossible for a 5GC to execute all its responsibilities to provide a UE with connectivity. The following NFs/components are therefore essential [Rom+19]:

- Access and Mobility Management Function (AMF)
- User Plane Function (UPF)
- Session Management Function (SMF)
- Authentication Server Function (AUSF)
- Unified Data Management Function (UDM)
- Unified Data Repository (UDR)
- Network Repository Function (NRF)

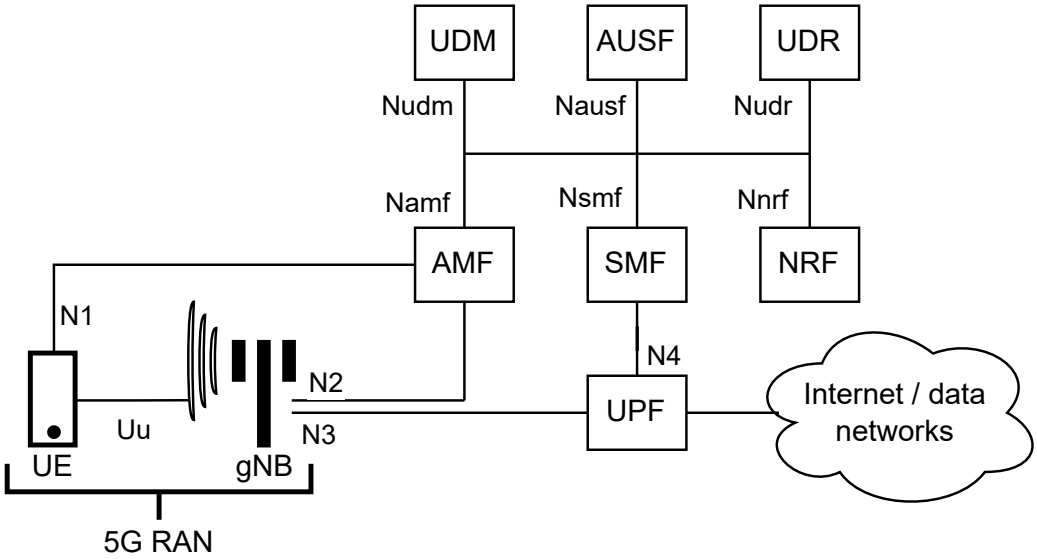

Figure 3.1: Essential components of a 5GC in a basic 5G setup, adapted from [Rom+19]

The NFs are interconnected as shown in Figure 3.1 and shed some light on the black box that represented the 5GC in Figure 2.2. We will now look into the services these functions provide and the planes to which they belong.

3.1.1 Access and Mobility Management Function

The AMF is connected with its UEs using the N1 interface. The connection with the gNB is made through the N2 interface. The main role of the AMF is to handle Non-Access Stratum (NAS) messages sent by a UE, serving as its main contact point with the core [Ryu25d]. This means the AMF takes part in most communication in the network, supporting encrypted signaling with UEs, allowing UEs to register themselves within the network as well as authenticate themselves when necessary. The AMF also allows a UE to move between different cells in the network. Idle devices are activated by the AMF when there is data waiting for them in the core. These supported features make the AMF responsible for 5G Mobility Management (MM) [Rom+19].

MM provided by the AMF is necessary to ensure that [Rom+19]:

- The UE can be notified about incoming messages and calls.
- The UE is able to communicate with others or with the Internet, for example.
- UE mobility is possible, allowing a user to move within or between different access technologies without loss of connectivity.
- The UE can be identified.
- The connection with the UE can be secured.
- The UE is able to communicate with the 5GC.

Connection Management States

Connection management states model the state of the NAS connection between the UE and the AMF. These states are [3GP24b]:

- `CM_IDLE`

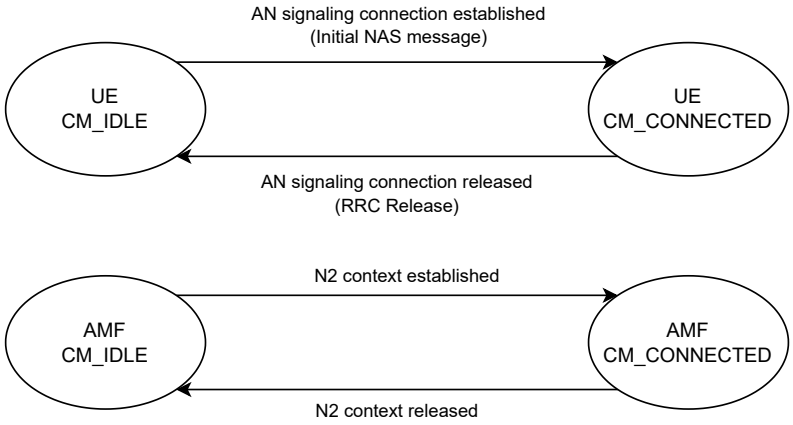

Figure 3.2: The Connection Management (CM) state machine as determined in [3GP24b].

- `CM_CONNECTED`

The UE starts without a NAS connection with the AMF, which corresponds to `CM_IDLE`. In this state, the UE will perform cell selection/reselection and Public Land Mobile Network (PLMN) selection to eventually start setting up its connection with the selected network. Regular NAS messages, such as authentication, are not yet sent, as the UE is still searching for a connection. The UE enters the `CM_CONNECTED` state whenever the N2 connection between the UE and AMF is established. Receiving the initial N2 message for registration (see Section 4.4.4) means the transition is made to the `CM_CONNECTED` state. When the UE enters the `RRC_IDLE` state (see Section 4.3.1), the UE will also enter the `CM_IDLE` state. Upon this transition to `CM_IDLE`, the UP connection of established PDU sessions is deactivated [3GP24b]. See Figure 3.2 for the full CM state machine.

Procedure Categories

5G MM procedures can be divided into three categories [Rom+19]:

1. Common procedures: can be initiated as long as the UE is in the `CM-CONNECTED` state.
2. Specific procedures: only one of these can be active at a time for each UE, for each access type.
3. CM procedures: these procedures establish security between the UE and the network and are used to reserve resources for data transmissions as well.

Connectivity Management

In order to receive session management services from the SMF, which provides the UE with data connectivity, the UE must first establish a connection with the network, which it does through the MM functions provided by the AMF. The UE achieves this by first selecting a network, which is done through deciding on a PLMN and a 5G AN. The UE then connects with the gNB by establishing a Radio Resource Control (RRC) connection with it as explained in more detail in Section 4.3.6. Then, an AMF must be selected for the UE to connect with. This is done through the parameters (selected PLMN, network slice information, etc.) chosen by the UE during connection establishment with the gNB. Now that the UE has an active RRC connection and the proper AMF has been chosen, the first NAS MM message can be sent over the N2 interface [Rom+19]. This marks the start of the registration procedure described in Section 4.4.4. This procedure is used to register the UE in the 5GC and establish the NAS connection with the AMF. The UE can then use this connection with the AMF as an entry

point for all interaction it needs to have with the 5GC, such as setting up a data session with the SMF [Rom+19].

In order to keep track of a UE with regard to its location, cells get grouped into Tracking Area (TA)s, which, as a group, are assigned to the UE in the form of a Registration Area (RA). Each gNB possesses a Tracking Area Identity (TAI) that matches one of the TAs. A gNB periodically broadcasts its TAI such that the UE can check if said TAI is equal to one of the TA in the RA it belongs to. If so, the UE is considered to have remained within the same RA. If the UE has moved to a cell with a TAI not equal to one of the TA's inside its RA, the UE is considered to have moved and will receive a new RA. The UE starts a registration update procedure, which, upon receipt by the AMF of the new network, will be checked to see if the new network already has an established context for this UE. In case it does not, the new AMF contacts the previous AMF to retrieve the context it had established with the UE. In order to keep the AMF up to date with idle UE's as well, the UE also periodically starts a registration update based on a timer. As the UE has not moved, the AMF is aware of its presence in the RA and is no longer left in the dark, as would be the case with an idle UE not providing any updates [Rom+19].

The AMF also allows for any authorized NF (such as the SMF) to subscribe to events related to the mobility of the UE. As mentioned, the AMF receives location reports by the gNB in the form of a cell identity, TA, and an optional timestamp describing the last time the UE was known to be in that location [Rom+19].

As for the actual implementation of session management, the AMF does not handle this itself: it forwards all of this signaling between the UE and the SMF. Authentication is also one of those ordered services in the AMF: the UE will attempt to authenticate itself through the AMF, but the actual authentication will happen in the AUSF [Rom+19].

Reachability Management

When a UE is idle, it is no longer actively communicating with the network. If the UPF were then to receive a packet that is meant for such an idle UE, the UE must first be forced to re-establish a connection with the network [Rom+19]. This is achieved through paging, and in the case of the UPF example, would follow these steps [Rom+19]:

1. The UPF buffers the packet.
2. The UPF notifies the SMF that a downlink packet has arrived.
3. The SMF informs the AMF it should set up UP resources for the data session.
4. The AMF sends a paging request to the gNB that owns the cell of the RA the UE is in.
5. The gNB pages the UE.
6. The UE responds to the AMF with a service request, re-establishing the connection such that the buffered packet can be sent to the UE.

Other AMF Functions

Besides the core functionalities described above, the AMF also handles the following functionalities in a 5G network [Emb22]:

- Next Generation Application Protocol (NGAP): The protocol to communicate between the gNB and 5GC.
- Allocating the Globally Unique Temporary Identifier (GUTI) (see Section 3.3.5).
- Selecting appropriate NFs: AUSF, UDM, Policy Control Function (PCF), SMF (depending on configuration or reports by the NRF).
- Support for Short Message Service (SMS)

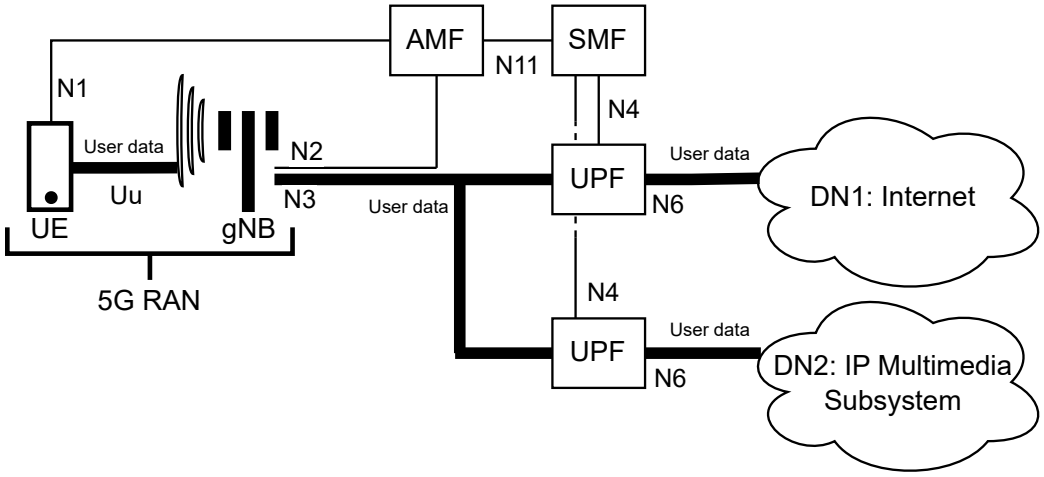

Figure 3.3: A UE with two established PDU sessions to two different DNs, adapted from [Rom+19].

3.1.2 User Plane Function

As a UE is free to move within the network, it often moves cells, which could make it difficult to reach the UE from the outside world due to changing IP addresses, for example. Besides that, a UE could be inactive, meaning that it would not receive messages addressed towards it since the outside world is not aware of the UE's current state. The UPF addresses these two problems by acting as a "stable IP anchor" [Rom+19] the outside world can use to address the UE. In that way, the mobility of the UE inside the network is abstracted away, as the outside world only needs to communicate with the UPF, which handles the further delivery of the packet [Rom+19]. As such, the UPF can be seen as the "gateway between the AN and external networks" [DPS23].

More concretely, the UPF's responsibility is to make sure that IP packets from the outside world to the UE will always be routable to that UE, despite its mobility, as these messages in reality are sent to the UPF, which then takes care of further processing to deliver the package. This is the case even if the UE is idle and not immediately reachable: the UPF will temporarily buffer the packet, request the AMF to page the UE, forcing it into a connection in order to finally deliver the buffered packet [Rom+19].

Besides these essential functions, the UPF also inspects the packets to analyze their content. Depending on the content of the packets, various network or user policies can be enacted, such as limiting data rates. Besides that, the UPF also provides traffic reporting for the SMF and QoS marking of packets to provide priority handling in congestion scenarios [Rom+19].

The UPF is the only NF that is part of the UP. All other NFs are part of the CP [DPS23].

PDU Sessions

The Session Management (SM) functionalities of the 5GS are necessary to provide a UE with a connection to external networks, known as Data Network (DN)s, such as the Internet. To facilitate this, the connection must be set up between the UE and the DN as well as maintained to ensure a connection with the correct QoS. These features are provided by the UPF, working together tightly with the SMF [Rom+19]. One such connection between a UE and DN is called a 'PDU session'. For such a PDU session, the UE is assigned an IP address which is managed by the UPF, which, as previously mentioned, acts as the UE's anchor point to the outside world [Ryu25g]. These sessions are established through a NAS procedure, described in detail in Section 4.4.10. Figure 3.3 shows a UE with two established PDU sessions.

Connecting with a DN is done through a PDU session establishment request as described in Section 4.4.10. In this request, the UE can provide a Data Network Name (DNN) such as 'Internet' (for general internet connectivity) to clarify which DN the UE wants to connect to. The UP connection will transport PDUs in both directions. An example of such a PDU is an IP packet [Rom+19].

A PDU session has the following main properties [3GP24b]:

- PDU session Identifier: identifies the PDU session for both the UE and network.
- Slice identifier: identifies the network slice of the PDU session.
- DNN: name of the DN the UE is connected to through the PDU session.
- PDU session type: which protocol type the PDU session uses. This can be IP-based (IPv4, IPv6, Dual-stack IPv6), Ethernet, or unstructured.
- Service and session continuity mode: how long the PDU session anchor point will remain valid and whether or not it can be re-allocated.
- UP security enforcement information: determines if UP ciphering and/or integrity protection should be activated.

These properties are determined when the PDU session is established and do not change afterwards [Rom+19].

3.1.3 Session Management Function

The SMF manages the UE's sessions: the PDU sessions mentioned in Section 3.1.2 in which the UE has a connection with the outside world. Communication with the UE happens indirectly through the AMF, using the N11 interface. The SMF is responsible for establishing, modifying, and releasing the data sessions, and the UE is able to open multiple of these on the same SMF [Rom+19].

Besides the basic management functions, the SMF works together tightly with the UPF using the N4 interface. As the UE requires an IP address for its data session, the SMF provides this, having the UPF act as the anchor point for this connection using that IP address. The SMF also works with the PCF for policy control, such as data rate limitations for a user session [Rom+19].

User Plane Management

Together with the UPF and UDM, the SMF manages the UP. The PDU sessions contain user data, and as such are part of the UP. The PDU session only goes through one SMF, but can have multiple UPF's in a chain, in which each UPF executes different responsibilities depending on its location in the chain. The SMF is responsible for selecting the UPF for such a UP connection, as well as what rules it should follow such as whether to buffer idle UE packets or not. The selection of a UPF happens by considering which UPF's are available and what their capabilities are [Rom+19]. Learning about the existence of a UPF can be done in different ways [Rom+19]:

- Configuring the available UPFs on the SMF.
- Discovering the available UPFs using the NRF.
- Exchanging information when setting up the N4 interface between the SMF and a UPF.

Together with the UDM, the SMF verifies if the UE requests are compliant with the user subscription using update notifications on the subscription data like QoS information [3GP24b].

Besides simply providing the UE with an IP through the UPF, the SMF also acts as the IP address manager for the PDU sessions. The UE can acquire information such as addresses of

Domain Name Service (DNS) servers, the Maximum Transmission Unit (MTU) the UE should consider, etc [3GP24b]. In essence, the SMF acts as a "5GC equivalent of an IP DHCP server" [Ryu25f].

Quality of Service Management

The SMF is also responsible for managing QoS flows. These flows define the QoS a PDU session receives, using a Quality of Service Flow ID (QFI) to identify a flow, with identical IDs meaning identical QoS. The QoS parameters for a PDU session are managed by the SMF, with specific values for the QoS either preconfigured on the SMF or configured during PDU establishment [3GP24b], which is described further in Section 4.4.10. Besides managing the QoS flows, the SMF also creates the QoS flows for each new PDU session that gets established [Cho24].

The possible QoS parameters are the following [3GP24b]:

- Resource type: determines if a QoS flow's resources are permanently allocated.
- Priority level: determines levels of priority for a QoS flow.
- Packet delay budget: sets an upper bound on packet delay between the UE and UPF.
- Packet error rate: sets an upper bound on how many PDU's are not successfully delivered to the receiver.
- Averaging window: defines the amount of time over which the guaranteed and maximum flow bit rate is calculated. The shorter this window, the better the service, as there is less room for errors.
- Maximum data burst volume: the maximum amount of data that should be handled within a one packet delay budget.

3.1.4 Authentication Server Function

The AUSF has the important responsibility of authenticating a UE using the credentials it provided during registration, for example, by working with the UDM to verify these [Rom+19]. Crucial for this functionality is that the AUSF also manages the session key that is used to derive other keys throughout the session, as further described in Section 4.4.5 [NKP22].

3.1.5 Unified Data Management Function

The UDM is used by the AMF to access user subscription data and by the AUSF to generate the authentication data that it uses to authenticate users and generate keying material. The SMF works with the UDM to verify policies with regard to the subscription of the UE, such as different access rules depending on roaming status [Rom+19].

A more concrete listing of the functionalities the UDM provides can be found in [3GP24b]. What follows is an excerpt of the most important ones [3GP24b]:

- Generation of 5G Authentication and Key Agreement (5G-AKA) credentials (see Section 4.4.5).
- User identification handling (storage and management of the Subscription Permanent Identifier (SUPI) for each subscriber, see Section 3.3.2).
- Support for de-concealing the Subscription Concealed Identifier (SUCI) (see Section 3.3.3).
- Access authorization based on subscription data (roaming restrictions).
- Registering the UE's serving NF (storing the serving AMF, SMF for each UE).
- Subscription management.
- SMS management.

The subscription data required by the AMF and AUSF is stored in the UDR in records. Each record contains the identity of the user, provided by its USIM when the UE needs to authenticate itself. This identity is the 5G SUPI, which is the 5G equivalent to the 4G International Mobile Subscriber Identity (IMSI) as described in Section 3.3.2. The UDR also provides network and user policy data, required by the SMF to implement QoS, for example. The UDR is interacted with through the UDM [Rom+19].

3.1.6 Network Repository Function

As the 5GC is built using an SBA, the services need to collaborate with each other to achieve a functioning core. As previously mentioned, the core uses the HTTP/2 REST system with a producer and consumer scheme in which the consumer utilizes the services provided by the producer to achieve its goals. In order to do so, the consumer needs a means of locating the producer, which is provided by the NRF. Services register themselves with the NRF, which keeps a record of this registration such that when a consumer requires the services of a producer that it has not located yet, it can query the NRF on how to reach it. Of course, this does require that the NRF is universally known to the other NFs to function [Rom+19].

More concretely, the NRF supports the functionalities listed below, which are a selection of the ones listed in [3GP24b]:

- An NRF bootstrapping service that enables other NFs to discover the services offered by the NRF
- Providing information about discovered NF to other NFs that request this.
- Maintaining a profile of discovered NFs, their supported services and their health status.
- Receiving notifications about newly registred/updated/deregistered NFs along with their potential services.

3.2 5G Security Architecture

With the 5GS being the successor to the 4G System (4GS), it should at least provide 4G equivalent security. On top of this, there were enhancements made to protect against attacks such as replay attacks, bidding-down attacks, etc [Pra+18]. These enhancements were made possible through the new security architecture shown in Figure 3.4, which contains the following security domains [3GP24f]:

1. Network access security: provides the UE with the means to authenticate itself and securely access network services, with a particular focus on protecting the (radio) interfaces.
2. Network domain security: allows the nodes in the network to securely exchange data.
3. User domain security: secures user access to the MS.
4. Application domain security: enables applications in the user and provider domains to exchange messages securely.
5. SBA domain security: allows the NFs to securely communicate within the Serving Network (SN) domain and with other network domains.
6. Visibility and configurability of security: informs the user about whether a security feature is in use or not (not in Figure 3.4).

As this thesis focuses on MITM attacks mainly targeting the user during the registration phase, network access security, network domain security, and application domain security are the most relevant for this subject. The MITM implementation of this thesis implements a rogue base station (attacking the network access security domain) to intercept or manipulate data sent

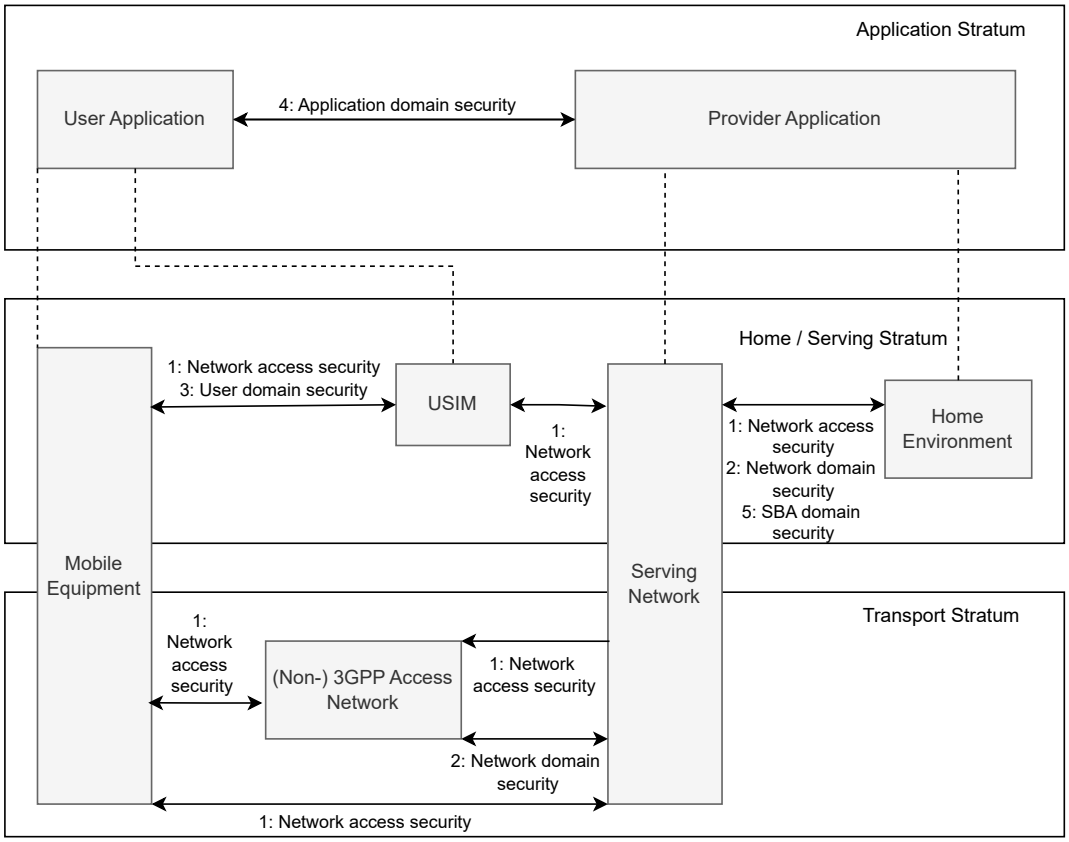

Figure 3.4: Overview of the 5G security architecture, as determined in [3GP24f]

between the UE and the 5GC (network domain security and application domain security).

3.2.1 Security Entities in the 5G Core

The 5GC contains the following security entities, part of the 5G Security Architecture (5GSA) [3GP24f]:

- AUSF: Described in Section 3.1.4. Handles the actual authentication of a UE using credentials created by the UDM.
- Authentication credential Repository and Processing Function (ARPF): The ARPF is a service offered by the UDM and is responsible for generating the 5G Home Environment Authentication Vector (5G HE AV), which is used during the Authentication and Key Agreement (AKA) procedure (see Section 4.4.5).
- Subscription Identifier De-concealing Function (SIDF): The SIDF is responsible for de-concealment of the SUCI (see Section 3.3.3) and is another service offered by the UDM. 'de-concealment' means the SIDF is able to reverse the encryption that turned the SUPI into a SUCI using both the used protection scheme and Home Network (HN) private key as its tools.
- Security Anchor Function (SEAF): The SEAF is a part of the AMF and is able to authenticate a subscriber using their provided SUCI. Besides that, the SEAF maintains the Visited Network's Security Anchor Key (K_{SEAF}), which is used to further derive other essential keys (see Section 4.4.5).

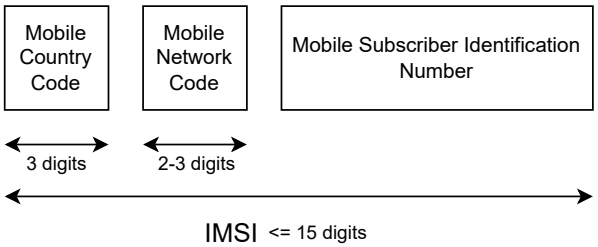

Figure 3.5: Composition of an IMSI as defined in [3GP24a]

3.3 Identifiers

5G gives the user a large list of different identifiers that all have their own purpose. This section will list the most relevant ones, detailing their composition and use according to the specification in [3GP24a].

3.3.1 International Mobile Subscriber Identity

Up to and including 4G, each mobile subscriber had a unique IMSI that was allocated to them, typically through the USIM. This identifies both the device and the subscription information and, as such, is used by the UE to identify itself during registration and authentication [Ryu25k]. The IMSI is composed as shown in Figure 3.5.

The parts that make up an IMSI are [3GP24a]:

- The Mobile Country Code (MCC) consisting of three digits. The MCC uniquely identifies the home country of the mobile subscription.
- The Mobile Network Code (MNC) consisting of two or three digits identifying the home PLMN of the mobile subscription, with the PLMN being a mobile network offered by a specific operator in the home country of the mobile subscription.
- The Mobile Subscriber Identification Number (MSIN) uniquely identifies the subscription within a given PLMN such that even if the MCC and MNC match, the subscriber is still uniquely identifiable.

3.3.2 Subscription Permanent Identifier

The SUPI has the same goal as the IMSI in previous generations: uniquely identifying a subscriber in the 5GS. The SUPI can take different forms, though, most notably that of the IMSI [3GP24a]. The other forms go beyond the scope of this thesis.

3.3.3 Subscription Concealed Identifier

The SUCI is a 5G unique identifier that contains the SUPI of the subscriber in an encrypted form. High level, the SUCI is the SUPI encrypted with a key provided on the USIM, that allows the UE to share its identity with the core without leaking it to other users on the network by sending it in plain text as was the case in previous generation networks [3GP24a]. The SUCI is defined as shown in Figure 3.6.

The parts that make up a SUCI are [3GP24a]:

- SUPI type: determines the type of SUPI that was used by the subscriber, such as the IMSI, for example.
- Home Network Identifier (HNI): identifies the HN of the subscriber, which can be done through combining the MCC and MNC as described above.

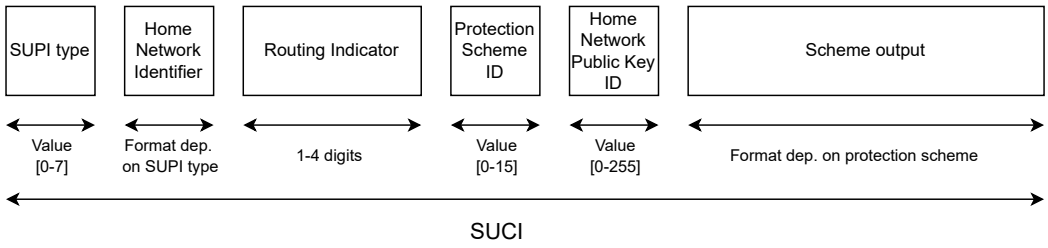

Figure 3.6: Composition of a SUCI as defined in [3GP24a]

- Routing Indicator (RI): combined with the HNI this allows for routing the signaling that includes the SUCI to the AUSF and UDM instances in order to serve the subscriber.
- Protection scheme identifier: determines which protection scheme was used to encrypt the SUPI to create the SUCI. A special protection scheme is the null scheme, which provides no encryption of the SUPI such that the SUCI and SUPI are equal.
- HN public key identifier: informs the network of which key was used by the UE to encrypt the SUPI, as typically multiple are available.
- Scheme output: the result of performing the encryption on the SUPI. In case the null scheme was used on a SUPI of type IMSI, the scheme output would be the MSIN of the IMSI. The information of the MCC and MNC parts of the IMSI is included in the HNI already, and is therefore not part of the scheme output.

As the SUCI is an encrypted form of the IMSI that the network itself is still able to decrypt, but a malicious actor could not, the SUCI is a privacy improvement for the user if it is used instead of the IMSI. By regenerating a SUCI each time, the UE is able to turn its permanent identity into a temporary one, making it harder to attack the privacy of the UE [Ryu25d].

3.3.4 Temporary Mobile Subscriber Identity

To support the goal of providing confidentiality to a subscriber (i.e., not revealing their identity), a UE may receive a Temporary Mobile Subscriber Identity (TMSI) from the core. This is a local and temporary identifier for the subscriber, assigned by the network, which only has significance in the network that assigned it. This means that the same TMSI will not be usable in a different network other than the one that assigned it. The TMSI consists of four octets in hexadecimal format. Often, the TMSI contains a part that is related to its creation time to avoid double allocation [3GP24a].

3.3.5 Globally Unique Temporary Identifier

Like the TMSI, the 5G GUTI is meant to provide a temporary identification to the UE in order to protect its identity. By assigning a 5G GUTI, the UE is able to use this instead of its SUPI in case it does not support SUCI encryption. As such, the 5G GUTI can be used by the UE to uniquely identify itself within the network without leaking its permanent identity [3GP24a].

The 5G GUTI has two main components [3GP24a]:

- Globally Unique AMF Identifier (GUAMI): identifies the AMF which allocated the 5G GUTI. The GUAMI is created by combining the MCC, MNC, and AMF ID.
- 5G TMSI: uniquely identifies the UE to the AMF(s) that allocated the 5G GUTI (see Section 3.3.4).

A full breakdown of the parts that make up the 5G GUTI can be seen in Figure 3.7.

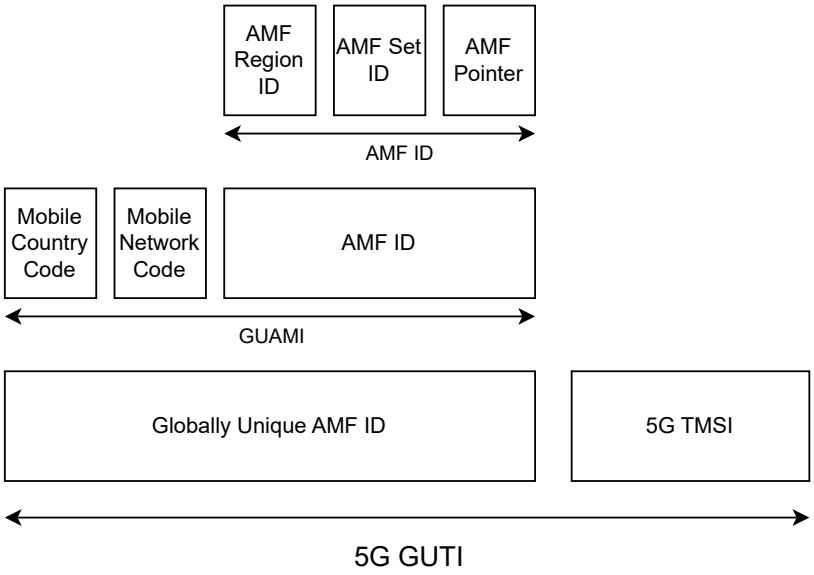

Figure 3.7: Composition of a 5G GUTI as defined in [3GP24a]

3.3.6 Shortened Temporary Mobile Subscriber Identity

The Shortened Temporary Mobile Subscriber Identity (S-TMSI) shortens the 5G GUTI to optimize message length for procedures such as paging [3GP24a].

The 5G S-TMSI is constructed using the following components, all present in a 5G GUTI [3GP24a]:

- AMF Set ID
- AMF Pointer
- 5G TMSI

3.3.7 Radio Network Temporary Identifier

Most identifiers above are an exclusive identity of the subscriber or device, some even permanent. This exclusivity and especially permanence mean these identifiers can be considered sensitive with regard to privacy. Temporary identifiers like the TMSI are a partial solution to this problem, but these are identifiers used by the core network, not accessible to the gNB as it is not active on the highest protocol layers. The gNB still needs some way then to refer to its connected with UEs without revealing their identity.

Importantly, as the temporary identifiers are used by the core network, they should be unique throughout the entire network using that core network. This is an important difference with an identifier used by a gNB: a gNB only needs to uniquely identify each UE connected to it. If a UE with the same temporary identifier as another UE is connected to a different gNB than that UE, there is no confusion on either gNB about which UE is which, as they are only connected to one of them. This fact allows gNBs to introduce a different temporary identifier: the Radio Network Temporary Identifier (RNTI). These identifiers are temporary, can belong to any UE and expire after inactivity (released RRC connection) or a handover (UE gets connected to a different gNB). There are many different RNTI's, among which the Cell Radio Network Temporary Identifier (C-RNTI) which a UE receives from its gNB at the end of the random access procedure [Att+22].

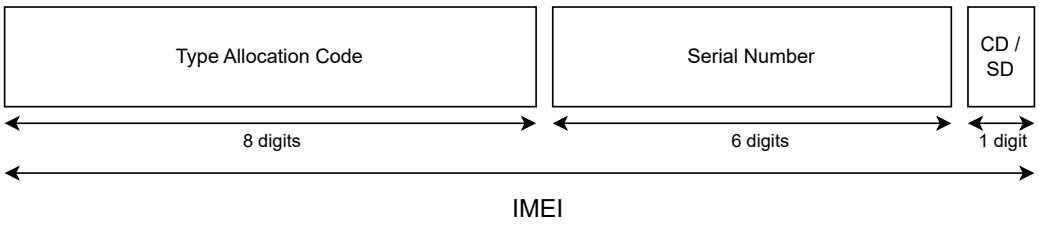

Figure 3.8: Composition of an IMEI as defined in [3GP24a]

3.3.8 Equipment Identifiers

International Mobile Station Equipment Identity

An International Mobile station Equipment Identity (IMEI) identifies a specific UE (the actual hardware itself) [3GP24a]. The IMEI is made up of the following components [3GP24a]:

- Type Allocation Code (TAC): identifies the manufacturer and the model of the UE. For example, TAC 86492106 denotes the Oneplus 11 smartphone [Wik24].
- Serial Number (SNR): the serial number uniquely identifies each equipment within the TAC.
- Check Digit (CD)/Spare Digit (SD): The CD is used to avoid manual transmission errors, but is not part of the transmitted digits of the IMEI. Alternatively, the SD always has the same value when transmitted by the MS: 0.

International Mobile Station Equipment Identity and Software Version Number

The International Mobile station Equipment Identity and Software Version number (IMEISV) is a slight variation on the IMEI, which instead of the CD/SD fields includes the Software Version Number (SVN) field, denoting the software version number the Mobile Equipment (ME) is using [3GP24a].

Permanent Equipment Identifier

The Permanent Equipment Identifier (PEI) can take on multiple forms just like the SUCI. Examples of PEI's are IMEI's, IMEISV's, or classic 48-bit MAC addresses [3GP24a].

Chapter 4

5G Protocol Stack and its Vulnerabilities

This chapter will examine the 5G protocol stack, focusing on how each protocol functions and which procedures it implements. Every protocol will also be analyzed with regard to security, explaining state-of-the-art attacks specifically targeting vulnerabilities in a given protocol.

As mentioned in Section 3.1, the 5GS has a functional split between signaling data and user data. This divides the protocols into two sets [3GP25e]:

1. UP protocols: these carry the user data through the Access Stratum (AS) for data sessions with the Internet, for example. The UP protocol stack can be seen in Figure 4.1.
2. CP protocols: these control the connection between the UE and the network (such as providing a means of authentication). The CP protocol stack can be seen in Figure 4.2.

4.1 Medium Access Control (MAC)

The Medium Access Control (MAC) layer is responsible for translating between the physical channels used to transmit data and the logical channels meant to separate network entities' different forms of traffic. Besides that primary responsibility, the MAC layer is also responsible for retransmissions and scheduling [DPS23].

The MAC layer is the lowest layer in the protocol stack that this thesis will treat. We will only briefly touch on MAC as our focus is on higher-position protocols in the protocol stack.

4.1.1 Logical Channels

The first part of the main service MAC provides to its upper layer is the availability of logical channels. When handling data in the upper layers, a separation can be made between user data (packets such as IP messages) and control data (packets carrying information such as NAS authentication data). This separation is purely logical, as from the MAC layer and below, these packets are treated equally. The MAC layer provides a means of translating between the logical channels used by the upper layers and the transport channels used by the lower layers [DPS23]. The set of logical channels MAC provides to the upper layers is the following [DPS23]:

- Broadcast Control Channel (BCCH): carries System Information (SI) messages (downlink only).
- Paging Control Channel (PCCH): carries paging messages (downlink only).

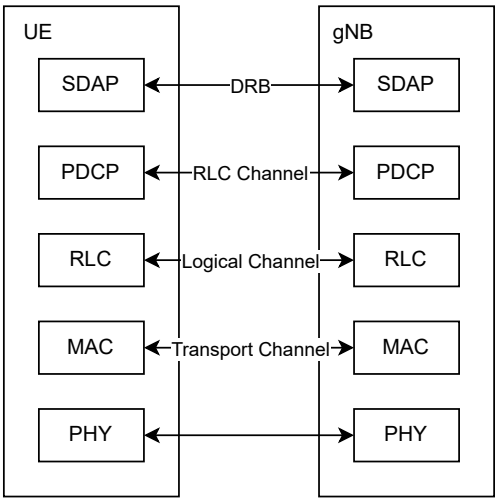

Figure 4.1: The 5G UP protocol stack as determined in [3GP25b].

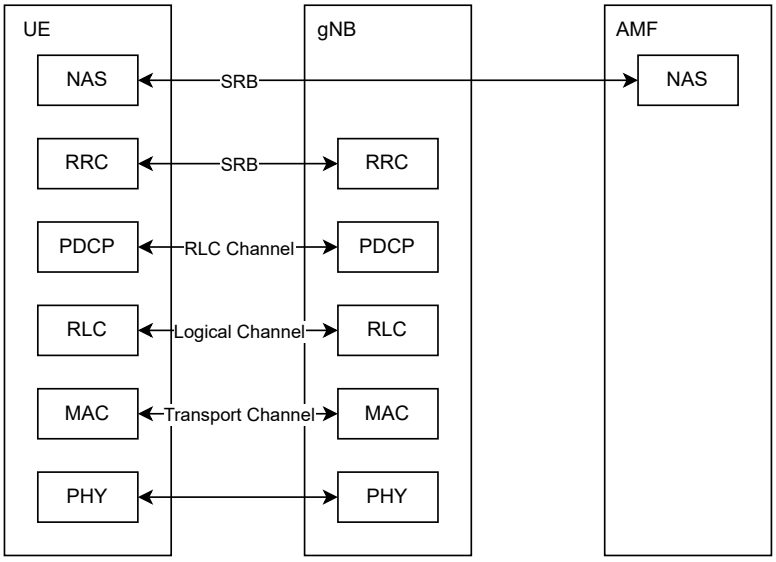

Figure 4.2: The 5G CP protocol stack as determined in [3GP25b].

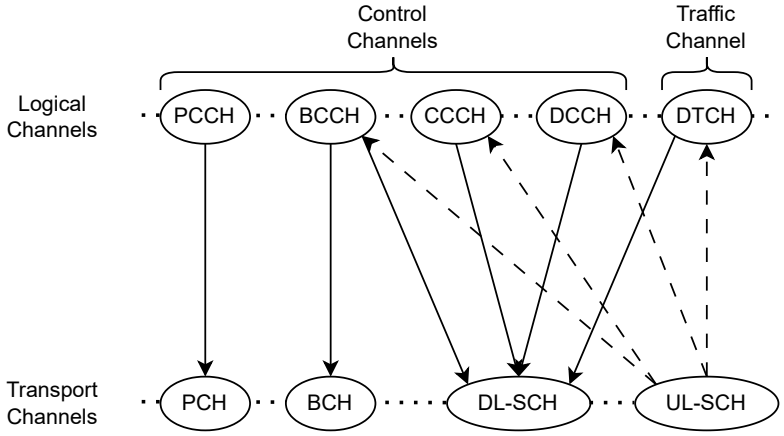

Figure 4.3: The translation between logical channels and transport channels provided by the MAC layer [Ryu25a]

.

- Common Control Channel (CCCH): carries the control messages and random access messages before an RRC connection is established.
- Dedicated Control Channel (DCCH): carries the control messages after an RRC connection is established ('dedicated' due to the established RRC connection with one UE).
- Dedicated Traffic Channel (DTCH): carries user data.

4.1.2 Transport Channels

The second part of MAC's service to upper layers is utilizing the transport channels provided by the physical layer below it in the protocol stack, based on the logical channels used by the layers above.

Data on the transport channels is organized into transport blocks, described using a transport format. In the transport format, information is given about how the transport blocks should be transmitted, defining the size of the transport block as well as the modulation and coding scheme, and the antenna mapping [DPS23].

The transport channels at MAC's disposal are [DPS23]:

- Broadcast Channel (BCH): carries a subset of the BCCH SI messages, specifically the Master Information Block (MIB).
- Paging Channel (PCH): carries paging information from the PCCH logical channel.
- Downlink Shared Channel (DL-SCH): carries most downlink data, including all CCCH, DCCH, DTCH as well as the SI that is not part of the BCH data.
- Uplink Shared Channel (UL-SCH): has the same responsibilities as the DL-SCH, but in the uplink instead.
- Random Access Channel (RACH): carries random access data.

The concrete translation of each logical channel to each downlink channel and vice versa can be seen in Figure 4.3.

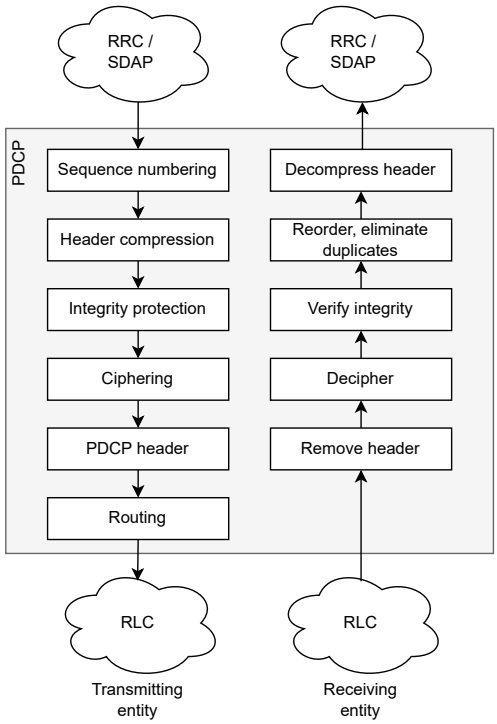

Figure 4.4: The functional view of the PDCP layer, adapted from [3GP25d].

4.2 Packet Data Convergence Protocol (PDCP)

The Packet Data Convergence Protocol (PDCP) layer acts as a management layer for all data that must be transmitted to/from both the UE and gNB in the network. It manages reliability and confidentiality for all data provided by the upper layers, sequential delivery, and duplicate removal.

4.2.1 Services

The PDCP protocol provides the following services to upper layers [3GP25d]:

- Transfer of UP and CP data.
- Header compression.
- Uplink data compression.
- Ciphering.
- Integrity protection.

The PDCP protocol expects the following services from lower layers [3GP25d]:

- Data transfer with acknowledgements to indicate successful delivery of PDCP PDUs.
- Unacknowledged data transfer.

4.2.2 Functioning

The PDCP layer has a predefined pipeline of operations it goes through for PDUs that need to be transmitted or were received. Both pipelines can be seen in Figure 4.4.

In general, the transmission pipeline can be described as follows [3GP25d]:

1. A sequence number is added to the incoming RRC/Service Data Adaption Protocol (SDAP) PDU, which allows the receiving party to determine if a particular message has already been received or not. Reception of duplicate PDU's is generally handled by discarding them.
2. UP data then goes through header compression, if enabled. CP data is not handled in this step.
3. Integrity protection includes both the presence of a Message Authentication Code (MAC) field and the actual integrity verification. Subsection 4.2.3 further treats this topic.
4. Besides integrity protection, PDCP also provides ciphering of the messages. If enabled, messages are ciphered as explained in Subsection 4.2.3 to provide confidentiality.
5. Finally, a PDCP header is added consisting of fields such as the PDCP sequence number and the MAC.

The pipeline triggered when receiving a PDCP PDU is mostly the same as the transmission pipeline, only reversing the operations done during transmission.

4.2.3 PDCP Security

PDCP ciphering includes both the ciphering of the MAC field and the data-part of the PDU. The upper layers configure the ciphering algorithm used by PDCP and the keys used for ciphering. The upper layers are also responsible for the (de-)activation of ciphering, based on the procedures they have to execute [3GP25d].

The required parameters that are used for ciphering are the following [3GP25d]:

- `COUNT`: the sequence number of the PDU.
- `DIRECTION`: direction of the transmission.
- `BEARER`: the identity of the used Radio Bearer (RB). A RB is a logical channel to transfer data [Ryu25i].
- `KEY`: ciphering keys generated in the AKA-procedure (see Section 4.4.5). CP data is ciphered using the RRC Encryption Key (K_{RRCenc}), and UP data with the UP Encryption Key (K_{UPenc}).

PDCP integrity protection protects both the PDU header and the data before ciphering. Again, the upper layers configure the integrity protection algorithm, the key that should be used, and the (de-)activation of the protection. Integrity protection uses the same required parameters to function as ciphering, except for the `KEY` parameter, which becomes the RRC Integrity Key (K_{RRCint}) and the UP Integrity Key (K_{UPint}) for CP and UP data, respectively [3GP25d].

The actual integrity verification happens by computing the MAC-field of the PDCP header using the parameters above after deciphering the PDU. The receiving party calculates an X-MAC based on the same parameters above: this is the MAC the receiver expects it will receive. In case a PDU was unaltered after transmission, the MAC calculated by the transmitting party and the expected MAC calculated by the receiving party will match. A mismatch means integrity verification failed, and the PDU should be discarded [3GP25d].

4.3 Radio Resource Control (RRC)

RRC can be defined as "the common language that should be understood by both the network and the UE" [Ryu25h]. RRC provides the means for the UE and network to configure how they will communicate together, reconfigure the established connection, and transmit NAS

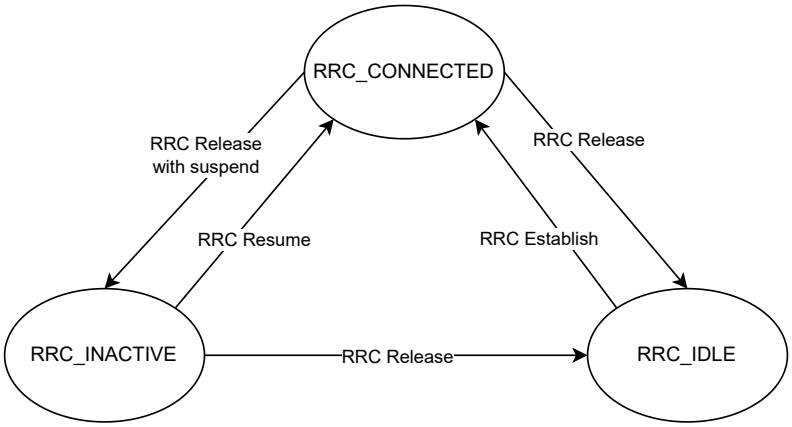

Figure 4.5: The RRC state machine as specified in [3GP24g].

messages for the CP signaling [Ryu25h]. To do so, RRC uses the services provided by the PDCP layer.

4.3.1 UE States

RRC implements three possible states a UE can find itself in regarding the current connection status of the UE to a gNB. If no RRC connection with the network has been established, the UE is in the `RRC_IDLE` state. Otherwise, the UE can be in either the `RRC_CONNECTED` or the `RRC_INACTIVE` depending on its activity in the network [3GP24g]. The following subsections will further characterize these RRC states.

RRC_IDLE

The UE is in the `RRC_IDLE` state as long as it has not established an RRC context with the network. While idle, the device periodically checks for paging messages which are the only downlink messages it could receive. In the uplink, the device can only send random access messages in order to start establishing a connection with the network. Since the UE is not yet connected to a gNB, it is also not yet a part of any cell [DPS23].

RRC_CONNECTED

When the UE and a gNB have established an RRC connection, the UE is in the `RRC_CONNECTED` state. The connection parameters have been agreed upon, which make up the RRC context for the connection. In `RRC_CONNECTED`, the UE is a part of the cell that is under the control of the gNB the UE is connected with [DPS23].

RRC_INACTIVE

The `RRC_INACTIVE` state is a new state that was added to 5G compared to the RRC states of 4G. The purpose of this state is to allow the UE to go idle to save battery life, for example, without losing the established RRC context. In early versions of 4G, when the UE went idle, it would have to re-establish its RRC context with the eNB before being able to transmit a message, incurring overhead that costs battery life and increases delay. Later versions of 4G did add an extension that provided similar features as `RRC_IDLE`, though, but the UE did still lose its connection to the core network. 5G's `RRC_INACTIVE` allows the UE and gNB to maintain the RRC connection when the UE goes idle, including the connection it made with the core network [DPS23].

4.3.2 Signaling Radio Bearers

An RB is a logical channel that is used to carry data [Ryu25i]. There is a distinction between the type of RB depending on the type of data it carries. Signaling Radio Bearer (SRB)s carry RRC and NAS messages, both part of the CP signaling. Data Radio Bearer (DRB)s carry user data, part of the UP signaling [3GP24g].

The following SRBs are defined in [3GP24g]:

- SRB0: carries RRC messages on the CCCH logical channel (see Section 4.1 for logical channels).
- SRB1: carries RRC and NAS messages on the DCCH logical channel, before establishing SRB2.
- SRB2: carries RRC and NAS messages on the DCCH logical channel and has a lower priority than SRB1.
- SRB3: carries RRC messages on the DCCH logical channel when the UE is in dual connectivity mode.
- SRB4: carries RRC messages such as measurement report information on the DCCH logical channel and has a lower priority than SRB1.
- SRB5: carries RRC messages such as measurement report information on the DCCH logical channel when the UE is in dual connectivity mode and has a lower priority than SRB1 and SRB3.

When messages are transmitted by a UE or gNB, they will be sent over a specific SRB using services of the PDCP, Radio Link Control (RLC), MAC, and Physical (PHY) layers as shown in the protocol stacks in Figures 4.1, 4.2 [DPS23].

4.3.3 Services

The RRC protocol provides the following services to upper layers [3GP24g]:

- Broadcast of common control information: refers to the transmission of SI using System Information Block (SIB)s, which a UE requires to know how to connect to the network.
- Notification of UEs in `RRC_IDLE`: RRC is responsible for the transmission of paging messages sent by the AMF to notify UEs of incoming services.
- Notification of UEs about Earthquake and Tsunami Warning System (ETWS) and/or Commercial Mobile Alert Service (CMAS): the paging system is used by the Public Warning System (PWS) to re-establish the connection between the UE and the network, putting it back in the `RRC_CONNECTED` state [BP22]. “Once active, the UE can receive warning messages that are transmitted through SIB messages” [BP22].
- Transfer of dedicated signaling: refers to the messages RRC will help transmit while the UE is in the `RRC_CONNECTED` connected state.
- Broadcast of positioning assistance data.
- Transfer of application layer measurement configuration and reporting regarding signal strength and latency parameters.

The RRC protocol expects the following services from lower layers [3GP24g]:

- Integrity protection: making sure that messages cannot be altered by third parties, which requires security to be activated for the connection.
- Ciphering: making sure that messages can not be read by third parties, which requires security to be activated for the connection.

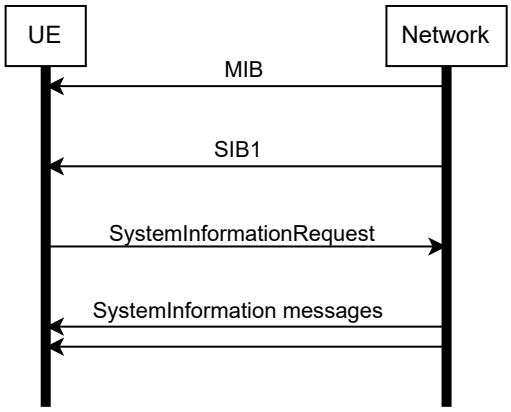

Figure 4.6: SI acquisition according to [3GP24g].

- Lossless, in-sequence delivery of information without duplication: making sure that messages actually arrive at their destination (using acknowledgements) and that they do so in sequence (using sequence numbering), without duplicates.

4.3.4 Functions

In what follows, we will list the most important functions provided by RRC regarding this thesis.

Broadcast of System Information

With regards to broadcasting of SI, the RRC protocol provides the following functions [3GP24g]:

- Information such as cell (re-)selection parameters and common channel configuration.
- Emergency notifications meant to warn users of UEs of incoming natural disasters etc. through the ETWS and CMAS notification systems.

RRC Connection Control

With regards to connection control, the RRC protocol provides the following functions [3GP24g]:

- Paging.
- Management of the RRC connection, meaning establishment, modification, suspension, resumption, and release of the connection. This includes assigning and modifying UE identities (C-RNTI for example), as well as establishing, modifying, suspending, resuming, and releasing of SRBs (excluding SRB0, which is used for SIB's).
- Access barring: can prevent certain UEs from accessing the network during high load, for example [Rom+19].
- AS security activation, meaning the configuration of both ciphering and integrity protection of the SRBs and DRBs.
- RRC connection mobility management, including intra- and inter-frequency handover, as well as maintaining the security of an already established connection during handover.

4.3.5 System Information Acquisition Procedure

SI refers to SI the UE requires to connect to the network, or inform the UE about network configurations, etc. It can be divided into two different types: the MIB and the SIBs. The MIB

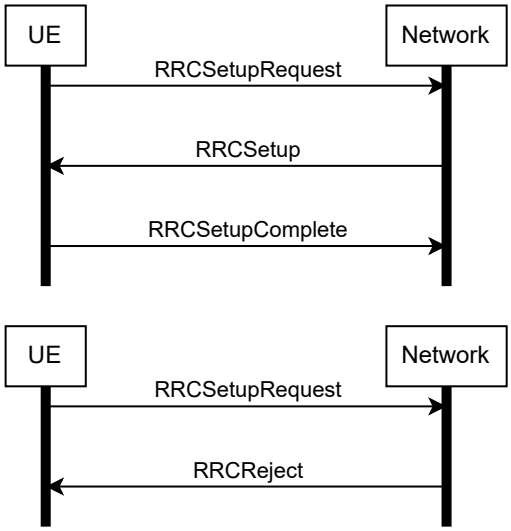

Figure 4.7: The RRC connection establishment procedure according to [3GP24g].

contains the essential information a UE needs in order to retrieve the equally essential SIB1, which it requires to attach to the network [DPS23].

Specifically, SIB1 contains information about the availability of other SIBs and how they are scheduled. Some SIBs are only provided on-demand, and the UE can only learn this through SIB1. Due to the nature of the information SIB1 provides, it is only applicable within the cell that broadcasts it. If a UE is in the `RRC_CONNECTED` state, the SI can also be delivered through an `RRCReconfiguration` message [3GP24g].

The UE starts the SI acquisition procedure when it wishes to connect to a cell (e.g., when powering on the device, for example). SI acquisition is also necessary when the UE wants to execute cell-reselection or return to its original cell after moving out of its coverage. It also starts this procedure when it receives a PWS notification, informing it about the transmission of a SIB that warns the UE about incoming dangers (SIB6, for example) [3GP24g].

The first SI the UE acquires when it is in the `RRC_IDLE` or `RRC_INACTIVE` state is the MIB, which is broadcast regularly by the gNB (periodicity of 80ms). The UE also acquires the MIB when it attempts to re-establish its RRC connection. After this, the UE uses the information provided by the MIB to acquire SIB1. On-demand SIBs can be requested by the UE using the `SystemInformationRequest` message [3GP24g]..

The general message flow of the SI acquisition procedure can be seen in Figure 4.6.

4.3.6 RRC Connection Establishment Procedure

The RRC connection establishment procedure is meant to establish an RRC connection between the UE and the gNB after the UE received the SI as per the SI acquisition procedure described in Section 4.3.5. RRC connection establishment involves the establishment of SRB1, over which most of the NAS registration procedure described in Section 4.4.4 will take place [3GP24g].

The general message flow of the RRC connection establishment procedure can be seen in Figure 4.7.

The UE initiates this procedure after it has acquired the necessary SI and is still in the `RRC_IDLE` state. The UE does not utilize this procedure to re-establish its RRC connection if it wants

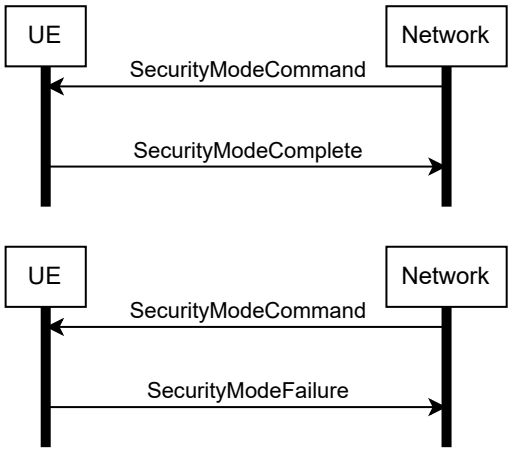

Figure 4.8: Security mode activation according to [3GP24g].

to move from the `RRC_INACTIVE` to the `RRC_CONNECTED` state. The procedure is started using the `RRCSetupRequest` message, in which the UE uses a 5G S-TMSI (see Section 3.3.6) if it has one (i.e., if the UE is registered in the TA of the current cell). If not, the UE uses a random 39-bit value. Besides this identity, the UE provides a cause as to what triggered the UE to want to establish an RRC connection. Example causes are `emergency` and `mo-signalling` [3GP24g].

Upon reception of the `RRCSetupRequest`, the gNB allocates a C-RNTI for the UE and generates the `RRCSetup` message [3GP25e]. This message will contain information such as the RB configuration for SRB1 [3GP24g].

Upon reception of the `RRCSetup` message, the UE sends a confirmation of the successful RRC connection establishment back to the gNB using the `RRCSetupComplete` message. This message has many Information Element (IE)s, but only two are non-optional: the `selectedPLMN-Identity` field, which denotes which PLMN the UE selected of the list provided by the gNB in SIB1 and the `dedicatedNAS-Message`, which contains the first NAS message the UE will send to the network, starting the NAS registration procedure described in Section 4.4.4 [3GP24g].

4.3.7 Initial AS Security Activation Procedure

The initial AS security activation procedure is meant to activate AS security protection after the RRC connection has been established and the UE and network have agreed on their security parameters (keys, etc.) [3GP24g].

The general message flow of the initial AS security mode activation procedure can be seen in Figure 4.8.

The network only initiates this procedure with a UE in the `RRC_CONNECTED` state. Besides that, this procedure is only initiated when SRB1 has already been established (i.e., the RRC connection establishment procedure was successfully executed) and before the establishment of SRB2 and/or DRBs as these require security to be activated first [3GP24g].

The `SecurityModeCommand` message sent by the gNB triggers the UE to derive the gNB Security Key (K_{gNB}), K_{RRCint}, and K_{RRCenc} based on the information provided in the `SecurityModeCommand`. These keys get derived depending on the selection of integrity and encryption algorithms provided by the gNB. Upon successful verification of the message sent by the gNB and successful derivation of the keys, the UE and gNB can now consider the AS security to be activated [3GP24g]. What that means in more concrete terms will now be explained.

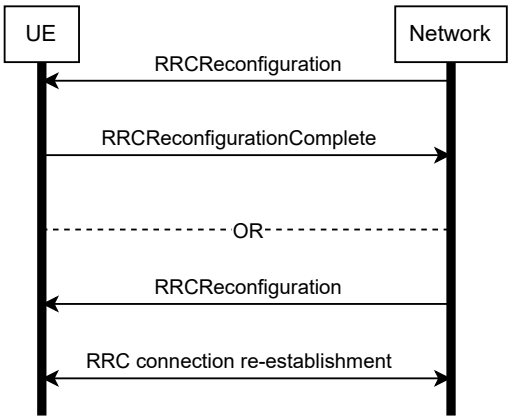

Figure 4.9: The RRC reconfiguration procedure as determined in [3GP24g].

RRC Security Mechanisms

RRC uses the security and protection services provided by the PDCP layer (see Section 4.2.3) to protect its messages. Since the NAS messages are encapsulated within the RRC messages, they are implicitly protected as well, but they do still provide their own protection. Layers below the PDCP layer are not integrity protected, however. When a message fails the RRC integrity check, the message should be discarded. Besides providing confidentiality and integrity protection, replay protection is also present. This ensures that no messages with the same PDCP `COUNT` value are accepted twice [3GP24f].

The input parameters for the 128-bit New Radio Integrity Algorithm (NIA)s are the following [3GP24f]:

- The RRC message itself.
- K_{RRCint} (128-bit integrity key).
- RB identity which corresponds to that of the used RB (SRB1, for example).
- The direction of transmission (uplink or downlink).
- The PDCP `COUNT` field representing the PDCP `COUNT` the sender currently has.

The input parameters for the 128-bit New Radio Encryption Algorithm (NEA)s are mostly the same as those for the NIA algorithms: the RB identity, transmission direction, and PDCP `COUNT` are once again included, but now aK_{RRCenc} is used as key and the length of the required keystream is an input as well [3GP24f].

4.3.8 RRC Reconfiguration Procedure

The RRC reconfiguration procedure is meant to modify an RRC connection by establishing, modifying, or releasing RBs, RLC channels, and more. The procedure is initiated by the network with UE's in the `RRC_CONNECTED` state [3GP24g].

The general message flow of the RRC reconfiguration procedure can be seen in Figure 4.9.

The procedure starts with the network sending an `RRCReconfiguration` to the UE, which modifies the RRC connection depending on the contents of the message fields [3GP24g]. Possible (re-)configuration options are [3GP24g]:

- RBs, including DRBs.
- Measurements.

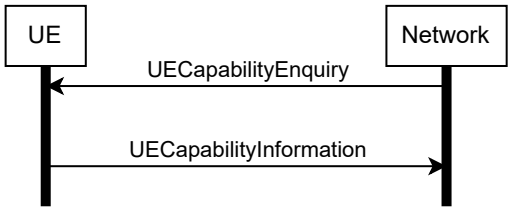

Figure 4.10: UE capability transfer as determined in [3GP24g].

- Secondary cells or cell groups.
- RLC channels.
- Other.

The `radioBearerConfig`-field, for example, determines the configuration of RBs that should be used after the reconfiguration. Reconfiguring RBs means establishing new RBs according to the provided configuration, modifying them (changing PDCP/RLC configurations) or releasing them. Reconfigurations are also able to update the security keys for ciphering and integrity protection [3GP24g].

Upon reception of the `RRCReconfiguration`, the UE will execute the required steps to update its RRC connection. Upon completion, the UE will respond to the network with an `RRCReconfigurationComplete`, indicating success. Should the reconfiguration fail, the UE will re-establish the RRC connection with the network, resetting the connection [3GP24g].

4.3.9 UE Capability Transfer Procedure

The UE capability transfer procedure is a way for the network to learn about the UE's supported features, such as supported frequency bands and modulation schemes. Doing so allows the network to allocate resources according the the actual capabilities of the UE, further enhancing the efficiency of the communication [Ryu25j]. The network initiates this procedure when the UE is in the `RRC_CONNECTED` state and when it requires information about the UE's supported features. This procedure should only be executed after security activation, but if the network does retrieve UE capabilities before security activation, it should not forward them to the core network [3GP24g].

The general message flow of the RRC UE capability transfer procedure can be seen in Figure 4.10.

The procedure starts with the network transmitting a `UECapabilityEnquiry` to the UE, which includes a `ue-CapabilityRAT-RequestList` with a `rat-Type` that describes what the network expects to be answered by the UE. Depending on the `rat-Type` value, the UE will include different information in its response. In the context of this thesis, the most relevant `rat-Type` would be `NR (0)`, which requires the UE to respond with the `supportedBandCombinationList`, `featureSets`, and `featureSetCombinations` [3GP24g]. The contents of these fields can be summarized as follows [3GP25c]:

- `supportedBandCombinationList`: defines the supported NR band combinations by the UE. For each band combination, the UE provides the associated feature set combination in `featureSetCombinations`.
- `featureSets`: downlink and uplink features defining capabilities such as bandwidth capacity, for example.
- `featureSetCombinations`: pools of feature sets that the UE supports on the different NR band combinations.

Figure 4.11: The RRC connection release procedure as determined in [3GP24g].

4.3.10 RRC Connection Release Procedure

The RRC connection release procedure is meant to release the RRC connection between the gNB and UE by releasing all radio resources and RBs. Besides that, this procedure is also used to suspend the RRC connection, including the RBs, but only if SRB2 and at least one DRB were set up [3GP24g].

The general message flow of the procedure can be seen in Figure 4.11.

The network uses this procedure in case it wants to move the UE from the `RRC_CONNECTED` state to either the `RRC_IDLE` state, completely releasing the connection, or the `RRC_INACTIVE` state, temporarily suspending the connection to save battery life, etc. (requires that SRB2 and at least one DRB were set up). This procedure is also used to keep a UE in `RRC_INACTIVE` or move it to `RRC_IDLE` if the UE wants to resume the RRC connection (i.e., the UE wants to transition back to `RRC_CONNECTED`), but the network refuses. When releasing the UE, the network can redirect the UE to another frequency and RAT (LTE only), should it wish to do so [3GP24g].

Upon receiving the `RRCRelease`, the UE will make the state transition as mentioned and follow the instructions (such as redirection) provided in the `RRCRelease` in case security was activated before the release. If security was not yet activated, all fields except `waitTime` should be ignored by the UE, including the redirection [3GP24g].

4.3.11 Known Attacks on RRC: Bidding-Down Attacks

Bidding-down attacks are a type of attack in which the goal is to downgrade the security of a connection by keeping the UE and network in the same generation (intra-generation bidding-down), but downgrading the security of the connection itself (forcing a weaker encryption scheme, for example) or by moving the UE to an older generation mobile network (inter-generation bidding-down). Moving the UE to an older generation network can be quite effective for an attacker, as these generations often provide less security, which can then be exploited. Usually, these inter-generational bidding-down attacks exploit legitimate protocol functionalities, making them hard to detect or mitigate [Kar+23].

Research done by [Kar+23] shows that even in 5G, there are multiple possible attack vectors for bidding-down attacks, especially in the case of an incorrect implementation of the standard by a UE or network. Their research centered around specific RRC and NAS procedures, and we will detail those RRC vulnerabilities here.

UE Security Capabilities

During connection establishment, the UE informs the gNB about which security algorithms it supports for ciphering and integrity protection. There are multiple possible schemes, among which is the null scheme, which provides no protection. As the UE must first inform the gNB of which algorithms it supports before security can be activated, this information is sent before security activation, meaning that it can be manipulated by a MITM to reflect different algorithms than what the UE originally sent, especially selecting only the null scheme as a supported algorithm. If the MITM succeeds in altering these capabilities, the security of the connection is downgraded [Kar+23].

To protect against this otherwise glaring vulnerability, the standard requires the following [Kar+23]:

1. Mandatory security algorithms beyond the null scheme. UE's that do not support these should be rejected.
2. The network should replay the capabilities it received to the UE after security activation, such that the UE is able to verify if the capabilities the network received matched those it originally sent.

Tests done by [Kar+23] showed that the majority of the tested core networks accepted UE security capabilities that should have been rejected. In response to these invalid capabilities, the core networks that failed the test utilized the null scheme for the connection, meaning the security of the connection was downgraded. The tested capabilities were: supporting only the null scheme, supporting only the non-mandatory algorithms, and supporting no algorithm at all. Furthermore, it was shown that two tested UEs did not verify the replayed security capabilities, not noticing the connection had been downgraded to the null algorithm. Finally, it was also discovered that all the tested UEs continued connecting with the network if it did not replay the capabilities it received at all, opening the UE up to the risk that if its capabilities were downgraded to the null scheme, it would not even know about it.

RRC Release with Redirection

During the RRC release procedure, the network is able to redirect the UE to a different cell on another frequency. In 5G, the redirection can be done to either the same RAT (NR) or to LTE. If the UE receives this message before security activation, it should ignore the redirection instruction and just release the connection. Should the UE not ignore the field however, and redirect itself to the cell a malicious actor provided in the release, it will be downgraded to 4G. From 4G, the attacker is free to downgrade the connection even further to 3G or 2G and exploit the vulnerabilities of the older generation networks [Kar+23].

Testing done by [Kar+23] showed that none of the tested UEs could be downgraded using this approach in 5G. All UEs ignored the redirection as they should.

4.3.12 Known Attacks on RRC: Warning and Emergency System

Using the PWS and ETWS system, authorities are able to quickly notify a large number of users in a large area of an incoming emergency, such as a flood or earthquake. Each country has some PWS-system or equivalent for this purpose, with options being SMS, smartphone apps, or the 5G integrated ETWS and CMAS, for example. Japan is one of the countries that uses the ETWS system [Ame18].

Given the severity of the circumstances that trigger such a notification, attacking the PWS could seriously impact the affected users. Possible ways to attack the PWS are to block the passage of this kind of message, exposing users to the danger they would otherwise be warned about, or to craft PWS messages, possibly causing mass panic among the recipients [BP22].

The PWS relies on paging to re-establish the connection between the network and UEs in the `RRC_INACTIVE` state. These paging messages contain a PWS-indication, informing the UE that it should monitor for SIB messages about the emergency. Possible SIBs are: SIB6 for primary ETWS notifications, SIB7 for secondary ETWS notifications, and SIB8 for CMAS messages [BP22].

The AMF is informed about an emergency through a `Write-Replace-Warning Request`, which it forwards to the RAN, such that it can be translated into SIB messages. These SIB messages are then broadcast within the specific cells that received the request. The devices inside the RAN get paged, with an emergency cause. This page moves the inactive UE's from `RRC_IDLE` to `RRC_CONNECTED` [BP22]. "Once active, the UEs can receive warning messages through the SIBs" [BP22].

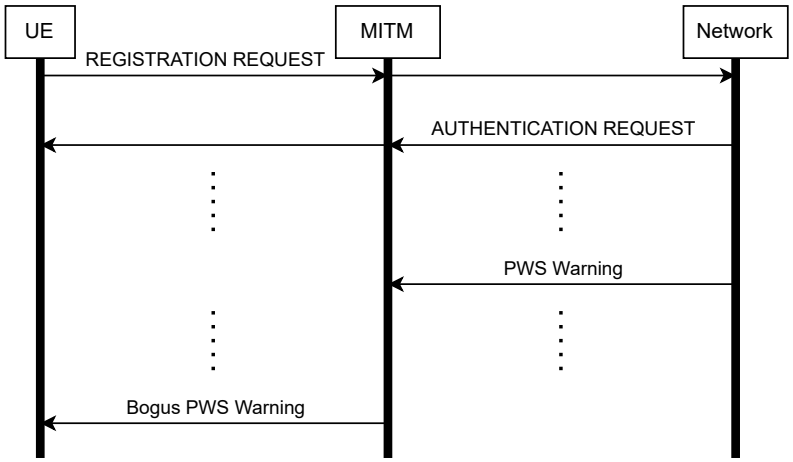

Figure 4.12: The PWS attack possibilities of a MITM setup: not forwarding legitimate warnings, and creating bogus warnings.

The issue that was exploited in research done in [BP22] is that these PWS systems lack security properties in that they do not have a verification of the legitimacy of the warning, nor do they provide integrity protection. We will now take a deeper look at the vulnerabilities of the PWS system.

Insecure Broadcast Messages

Since MIB and SIB messages are broadcast messages meant to be acquired by a UE possibly without a security context, these messages are sent without security protection (neither ciphering nor integrity protection). This means that there is no way for a UE that receives such a warning to verify whether it was sent by the real network or not. An attacker could send spoofed SIB warnings to the UE, and the UE would display them to the user as it would normally do so [BP22].

No Acknowledgements in ETWS/CMAS Delivery

Neither of the paging procedure and SIB messages provides a method to the network to verify if a UE actually received the message they sent. Upon reception of a SIB6 message, the UE displays the warning to the user, but never responds to the network to inform it that it received the message. This means that attackers have an easier time suppressing PWS messages, as the network is unaware of which of its UEs received their warning [BP22].

PWS Attacks Using a MITM

The research done in [BP22] shows that there are two main ways a MITM setup can attack PWS messages: PWS spoofing attack and PWS warning suppression. A message flow of a MITM setup exercising both of these methods can be seen in Figure 4.12.

If the MITM wishes to spoof PWS warnings, it can do so as long as the UE remains connected to it. The MITM is able to craft bogus PWS warnings and send them to the UE, which, due to the lack of message verification, will accept them and display the warnings to the user [BP22].

As the UE is connected to the rogue base station, the MITM is in charge of deciding what gets forwarded to the UE and what does not. Typically, deciding not to forward a message as a MITM can disrupt the communication in general, as sequence numbers might no longer line up, or one of the parties keeps retransmitting the message that was not forwarded. For the PWS

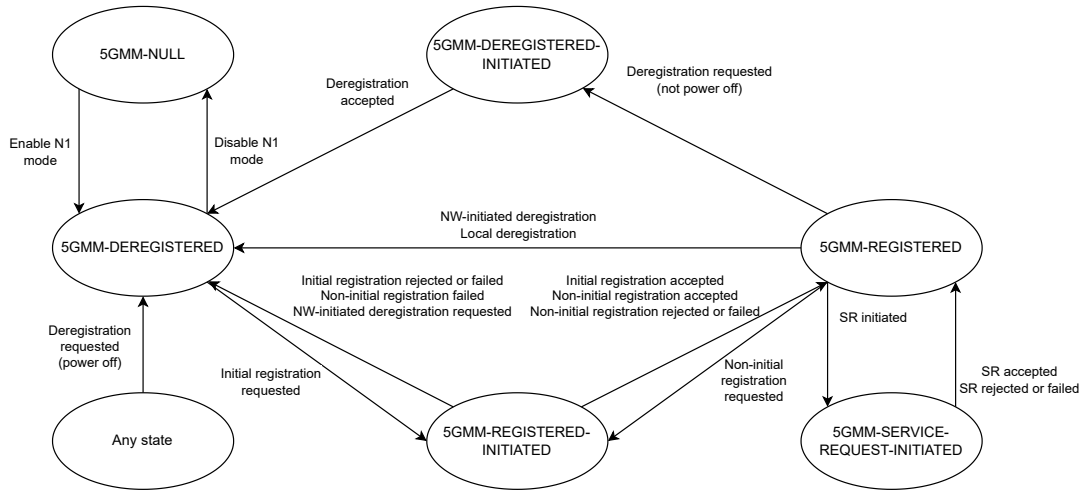

Figure 4.13: The 5GMM main states in the UE as per [3GP24d].

messages, however, this is not the case. Due to the network never knowing if a UE actually received the messages, the MITM can decide not to forward the paging messages sent to the UE, causing it to remain in the `RRC_INACTIVE` state [BP22]. As for how long the UE would remain in the dark, [BP22] described the following: "the suppression continues until the UE disconnects from the attacker and connects to the real network appropriately" [BP22].

4.4 Non-Access Stratum (NAS)

NAS is the protocol that is used when the UE and AMF need to exchange messages. NAS messages are sent by the UE and AMF (through the gNB) encapsulated within RRC messages, and they provide the bulk of the communication within the CP signaling.

NAS has three main responsibilities [3GP24d]:

1. UE mobility management such as authentication, identification and security activation.
2. UE session management which allows the UE to establish data connections with the outside world.
3. Transport of other services such as SMS.

To achieve its responsibilities, NAS provides two sets of procedures (the NAS sublayers): those for 5GS Mobility Management (5GMM) and 5GS Session Management (5GSM) [3GP24d].

4.4.1 NAS Sublayers

The 5GMM sublayer is responsible for providing identification, security, connection, and mobility management for the UE as well as support for generic messages. The 5GSM sublayer is responsible for handling the PDU sessions between the UE and the outside world. The 5GSM sublayer provides authentication, authorization, establishment, modification and release of PDU sessions as well as support for handovers of existing PDU sessions [3GP24d].

5GMM Sublayer States

The 5GMM sublayer of the UE and the network can be described through state machines in which the states are managed per access type (3GPP or non-3GPP) independently [3GP24d]. The whole state machine containing the 5GMM states in the UE can be seen in Figure 4.13, and the state machine containing those of the network can be seen in Figure 4.14.

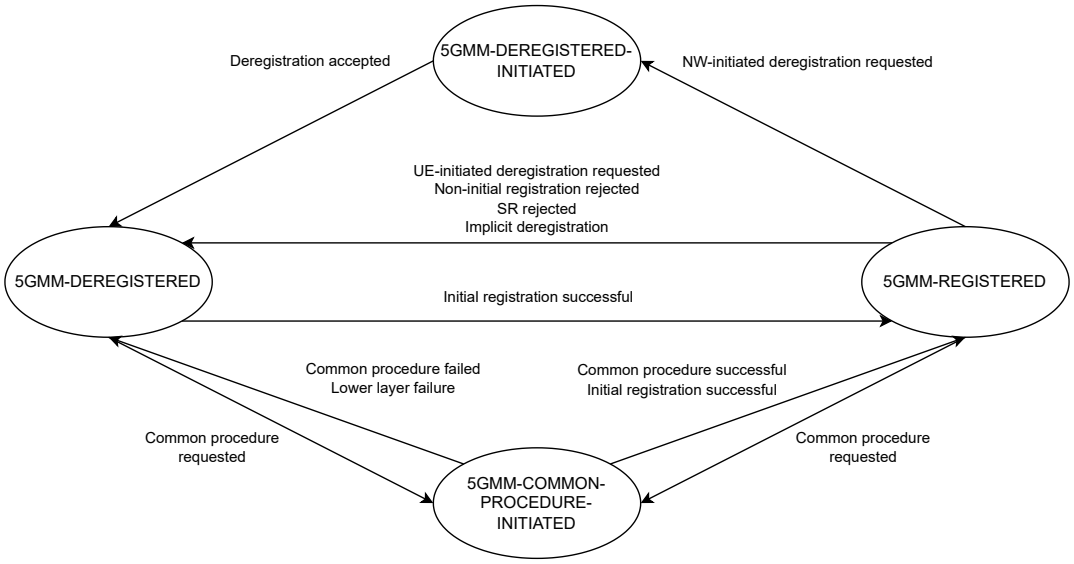

Figure 4.14: The 5GMM main states in the network as per [3GP24d].

5GMM Main States in the UE

The 5GMM main states in the UE can be described as follows [3GP24d]:

- `5GMM-NULL`: 5GS services are disabled in the UE and there is no MM happening.
- `5GMM-DEREGISTERED`: the UE has not yet established a 5GMM context, which it can start using the NAS registration procedure described in Section 4.4.4.
- `5GMM-REGISTERED-INITIATED`: the UE has started the registration procedure and is waiting for a response from the network.
- `5GMM-REGISTERED`: a 5GMM context has been established and the UE is able to establish PDU sessions.
- `5GMM-DEREGISTERED-INITIATED`: the UE has initiated the deregistration procedure and is waiting for a response from the network.
- `5GMM-SERVICE-REQUEST-INITIATED`: the UE has started the service request procedure and is waiting for a response from the network.

The main 5GMM states are subdivided into several substates, which can be found in [3GP24d]. However, these are beyond the scope of this thesis and will not be explained here.

5GMM Main States in the Network

The 5GMM main states in the network can be described as follows [3GP24d]:

- `5GMM-DEREGISTERED`: the UE is currently deregistered and no 5GMM has been established. The network is still in this state when it receives the `REGISTRATION REQUEST` from the UE.
- `5GMM-COMMON-PROCEDURE-INITIATED`: the network finds itself in this state after it starts a 5GMM procedure and is waiting for a response from the UE.
- `5GMM-REGISTERED`: the network is in this state if a 5GMM context has been established and one or more PDU sessions can be established.
- `5GMM-DEREGISTERED-INITIATED`: the network enters this state after it has started the deregistration procedure and is waiting for the UE to respond.

4.4.2 Transmission of 5GSM messages

Transporting 5GSM messages is done through piggybacking them in specific 5GMM transport messages using a dedicated IE for this purpose after a 5GMM context has been established. If this happens, both the AMF and SMF will execute the procedures carried in the 5GMM message, with the AMF executing the 5GMM part, and the SMF executing the part contained within the piggybacked 5GSM message. These procedures are executed independently of one another, without the failure of either one influencing the success of the other [3GP24d].

Before 5GSM procedures can be executed, a 5GMM context with activated security must be established between the AMF and UE. This is done through the security control procedure described in 4.4.6 [3GP24d].

4.4.3 Procedures Overview

As both 5GMM and 5GSM provide many procedures, we start with an overview of them. After that, we will take a deeper dive into the procedures provided by the two sublayers that are most relevant for this thesis.

Types of 5GMM Procedures

There are three types of 5GMM procedures [3GP24d]:

1. 5GMM common procedures: can be initiated as long as the UE is in the `5GMM-CONNECTED` state. Procedures of this type are:
   - Network-initiated procedures such as AKA or security activation.
   - UE-initiated procedures such as NAS transport.
   - UE or network 5GMM status updates.
2. 5GMM specific procedures: only one 5GMM specific procedure can run at the same time for one UE. Procedures of this type are:
   - UE-initiated: the registration procedure.
   - UE- or network-intiated: the deregistration procedure.
3. 5GMM connection management procedures:
   - UE-initiated: the service request procedure.
   - Network-initiated: the paging and notification procedures.

5GMM Procedures in General

Before the UE can communicate with the AMF, it must first establish an RRC connection with the network. The RRC connection establishment procedure is described in Section 4.3.6. Once the connection has been established, the UE can transmit its first NAS message [3GP24d]. There are four options regarding the initial NAS messages [3GP24d]:

1. `REGISTRATION REQUEST`
2. `DEREGISTRATION REQUEST`
3. `SERVICE REQUEST`
4. `CONTROL PLANE SERVICE REQUEST`

When the UE needs to identify itself to the network during the 5GMM procedures, it can use multiple identifiers depending on its history within the network. The SUPI is of the highest interest privacy-wise, as this is the UE's globally unique permanent identifier (see Section 3.3.2). Depending on whether the UE uses the null-scheme or not for encrypting the SUPI, the SUCI

can work as a privacy-preserving identifier for the UE. As the SUCI achieves the same as the SUPI without leaking the user's identity if encrypted, the UE uses the SUCI as its identifier in NAS signaling, not the SUPI [3GP24d]. Specifically, the SUCI is used by the UE under the following 5GMM circumstances [3GP24d]:

1. the UE has to send a `REGISTRATION REQUEST` and does not yet have a valid 5G GUTI.
2. the UE receives an `IDENTITY REQUEST` for its SUCI.
3. the UE has to send a `DEREGISTRATION REQUEST` and does not yet have a valid 5G GUTI.

As can be seen from the items above, if the UE does possess a 5G GUTI, it will use this when identification is necessary. Using the 5G GUTI helps with privacy concerns as this is a temporary identifier instead of a permanent one, making it more difficult to link a user to their 5G GUTI [3GP24d].

To summarize, 5GMM procedures handle all NAS signaling between the UE and AMF, allowing the two parties to establish a connection that is ready to be used for user data between the UE and external DNs.

Types of 5GSM Procedures

There are three types of 5GSM procedures [3GP24d]:

1. PDU session procedures: used for the manipulation of PDU sessions by the network or for establishing a PDU session by the UE.
   - PDU session establishment (UE-initiated).
   - PDU authentication and authorization (network-initiated).
   - PDU session modification (network-initiated).
   - PDU session release (network-initiated).
2. Transaction procedures: used by the UE to request the manipulation of a PDU session by the network as described in the list above.
   - UE-requested PDU session modification.
   - UE-requested PDU session release.
3. Common procedure:
   - 5GSM status procedure.

5GSM Procedures in General

5GSM supports the following PDU session types [3GP24d]:

1. IPv4
2. IPv6
3. IPv4v6
4. Ethernet
5. Unstructured

These session types are included by the UE in the `PDU SESSION ESTABLISHMENT REQUEST` according to its IP stack capabilities [3GP24d].

As mentioned in Section 3.1.3, the SMF handles the session management of the UE, allowing it to connect to external networks through PDU sessions. The SMF allows for the UE to establish multiple PDU sessions with different or the same DNs. For its connection with the

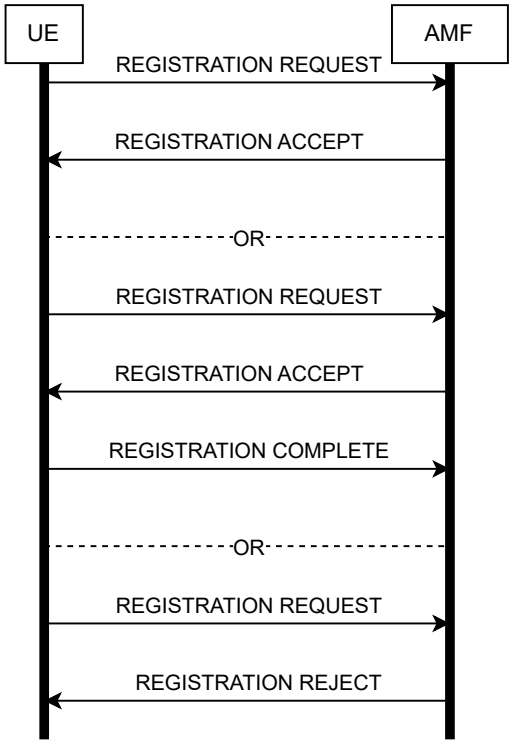

Figure 4.15: The registration procedure for initial registration as per [3GP24d].

outside world, the UE requires an IP address, which it gets from the SMF based on the PDU session type the UE requested (IPv4, for example). Sending the allocated IP to the UE is done through NAS, as a reply to the `PDU SESSION ESTABLISHMENT REQUEST` sent by the UE [3GP24d].

Besides the establishment, termination and management of the PDU sessions, 5GSM also provides QoS management for the connection. This enables the network to apply different forwarding treatments of the user traffic based on the agreed parameters of the QoS flow [3GP24d].

In short, 5GSM procedures enable the UE to communicate with the outside world through the SMF and UPF's managed PDU sessions.

4.4.4 5GMM: Registration Procedure for Initial Registration

The registration procedure is meant to allow a UE to connect to the network by registering itself for 5GMM and, later, 5GSM services. The registration procedure is the first NAS procedure a UE goes through in a typical connection setup. The general message flow of this procedure can be seen in Figure 4.15.

The UE initiates the procedure by sending a `REGISTRATION REQUEST` to the AMF while in the `5GMM-DEREGISTERED` state with the intent to register for 5GS services, emergency services, or SMS over NAS [3GP24d]. The `REGISTRATION REQUEST` contains the following (for this thesis relevant) IEs [3GP24d]:

- `5GS registration type`: indicates the type of registration procedure the UE is performing. These reflect the intents mentioned earlier, taking values such as 'initial registration' or 'emergency registration' [Ryu25c].

- `5G Key Set Identifier (ngKSI)`: denotes the security context the UE is wishing to utilize for its connection, to "help the core correlate the UE with its existing security keys" [Ryu25c]. If no (valid) security context was selected, a new one will be established for the UE [3GP24d].
- `5GS mobile identity`: denotes both the type of identifier used by the UE to identify itself as well as the value for that identifier. This can be the UE's SUCI or a temporary identifier such as the 5G GUTI [Ryu25c].
- `5GMM capability`: provides the network with information about the UE's supported features related to 5GMM [Ryu25c].
- `UE security capability`: carries the supported NAS and AS ciphering and integrity protection algorithms [Ryu25c].

The UE has to follow strict specifications about which identifier types it chooses for the `5GS mobile identity` field. The following decisions must be evaluated, in the given order [3GP24d]:

1. If the UE was previously registered in NSA mode with a valid native 4G GUTI and security context available and the network indicates that N26 (interface between AMF and its 4G equivalent) is supported [Ryu25e], then the UE creates a mapped 5G GUTI from the 4G GUTI which it uses as its 5GS mobile identity.
2. If the UE possesses a valid 5G GUTI given by PLMN it is registering in, it should use this as an identifier.
3. If the UE possesses a valid 5G GUTI given by an equivalent PLMN as the one it is registering in, it should use this as an identifier.
4. If the UE possesses a valid 5G GUTI given by a different PLMN than the one it is registering in, it should use this as an identifier.
5. If a SUCI is available on the UE, it should use this SUCI as identifier.
6. If the UE does not hold any valid 5G GUTI nor any SUCI but is registering for emergency services, it should use the PEI as identifier.

When the AMF receives the `REGISTRATION REQUEST`, it must determine what the next procedure to trigger is. It can choose from the 5GMM common procedures (identification, authentication, ...), and this decision is made using the information provided in the `REGISTRATION REQUEST`. The most important field that determines the outcome of this decision is the `5GS registration type`, which, if set to 'emergency services', allows the AMF to forego the authentication procedure even if no 5G NAS security context has previously been established. The AMF will not check for mobility or access restrictions in this case, nor subscription restrictions, etc.

If the network accepts the initial registration request, the AMF sends a `REGISTRATION ACCEPT` message to the UE, which contains the following (most relevant) IE's [3GP24d]:

- `5GS registration result`: indicates the outcome of the registration procedure and the type of registration that was accepted [Ryu25c].
- `5G GUTI`: upon registration acceptance, the UE receives a new 5G GUTI [Ryu25c].
- `Equivalent PLMNs`: contains PLMN codes (MCC+MNC) that represent PLMNs onto which the UE can roam without additional registration steps [Ryu25c].
- `TAI list`: defines the coverage area in which the UE can remain registered with this AMF [Ryu25c]. See Section 3.1.1 for more information.

If the network cannot accept the initial registration, the AMF sends a `REGISTRATION REJECT` message to the UE indicating the cause of the failure.

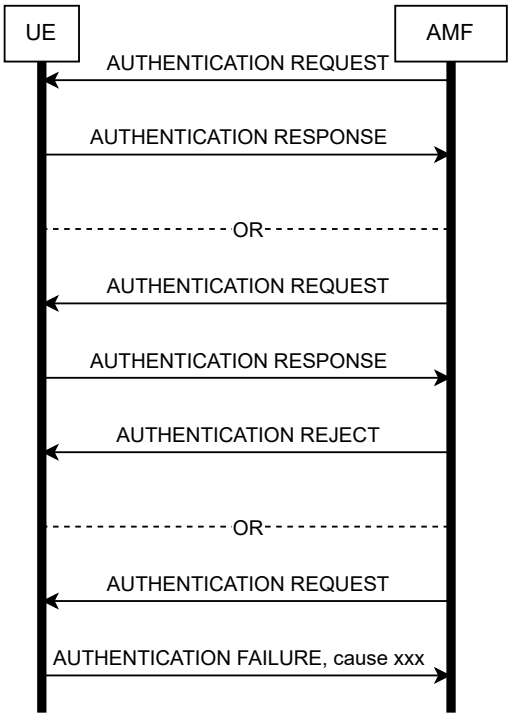

Figure 4.16: The 5G-AKA procedure as per [3GP24d].

If the UE sent a common `REGISTRATION REQUEST`, with `5GS registration type` set to 'initial registration', the AMF and UE need to start the process of mutually authenticating each other, which is described in Section 4.4.5 below.

4.4.5 5GMM: Primary Authentication and Key Agreement (AKA) Procedure

The AKA procedure is one of the most important procedures the UE and network go through during their connection setup. It allows the UE and network to authenticate one another: mutual authentication, but also provides both parties with the keying material they need to secure the signaling after this procedure [3GP24d]. Specifically, the keying material this procedure outputs is the anchor key: K_{SEAF} that allows the two parties to derive keys they need for security contexts without needing to re-authenticate for each context [3GP24f].

[3GP24d] defines two methods to implement the primary AKA procedure:

1. Extensible Authentication Protocol (EAP) based primary AKA.
2. 5G-AKA based primary AKA.

This thesis will focus on 5G-AKA to remain within scope. However, it is essential to note that both methods are always supported on the UE and AMF, making both methods a valid choice.

The general message flow of the 5G-AKA procedure can be seen in Figure 4.16

In what follows, we assume the following:

1. The UE connects to a SN, not its HN (if the UE connects to its HN the procedure is still the same, but with fewer steps as there is no SN that also requires the keying material).

2. The UE uses a SUCI as its identity.

The mutual authentication between the two parties starts upon reception of the `REGISTRATION REQUEST` sent by the UE to the AMF. The subsequent steps work as follows in case our assumptions above hold [Cox21]:

1. The SN's AMF which received the `REGISTRATION REQUEST` asks the HN's AUSF to authenticate the UE. To do so, the AUSF receives the following information from the AMF:
   - SN name (consists of MNC and MCC).
   - The subscriber's identity (SUCI in this case).
2. When the HN's AUSF receives the request to authenticate the user for that SN, it first checks if the AMF that made the request can use the given SN name. If so, the AUSF forwards the information to the UDM that allocates a new resource for the subscriber's identity.
3. The ARPF now asks the SIDF to return the SUPI that belongs to the provided SUCI. After this, it looks up the user-specific key K to generate an Authentication Vector (AV) that contains the following parameters:
   - `RAND`: a random number that serves as an authentication challenge and is used to compute the other three parameters below, combined with K and the SN's name.
   - `XRES*`: the expected response a UE will compute if it is the owner of K. If the UE does not possess K, it will compute a response different from `XRES*`.
   - `AUTN`: an authentication token that proves to the UE the network has the same value of K as it does. `AUTN` also includes a sequence number that can be used to protect against replay attacks.
   - Home Network's Security Anchor Key (K_{AUSF}): the HN's security anchor key, which can be used to serve the SN's security anchor key.
4. After generating the AV, the ARPF returns this to the AUSF together with the retrieved SUPI.
5. The AUSF stores `XRES*` so that it can be used when the HN has to perform the authentication procedure, hashes `XRES*` to get `HXRES*`: a hashed variant of `XRES*` that the SN can use to check the UE's response. The AUSF also stores the SUPI it received and computes the K_{SEAF} security anchor key for the SN.
6. All authentication material is now forwarded to the SN's AMF, which can now send the AV to the UE to let it authenticate itself.

A full overview of the `REGISTRATION REQUEST` triggering 5G-AKA to compute the `AUTHENTICATION REQUEST` and the expected `AUTHENTICATION RESPONSE` can be seen in Figure 4.17.

After receiving the `AUTHENTICATION REQUEST`, the UE will handle it as follows [Cox21]:

1. The UE receives the `RAND` and `AUTN` values and checks if the value of K that was given by the network matches the one the UE expects. The sequence number is also verified to see if it has not yet been parsed.
2. If these conditions are satisfied, the UE calculates the `AUTHENTICATION RESPONSE` containing `RES*` by calculating `RES` (combining `RAND` with K) and combining this with the SN's name.
3. The UE now derives its security keys K_{AUSF}, K_{SEAF} and AMF Security Key (K_{AMF}), which are derived in the that order. Using K_{AMF}, the UE and AMF will be able to agree on a NAS Integrity Key (K_{NASint}) and a NAS Encryption Key (K_{NASenc}) for the protection of NAS signaling.

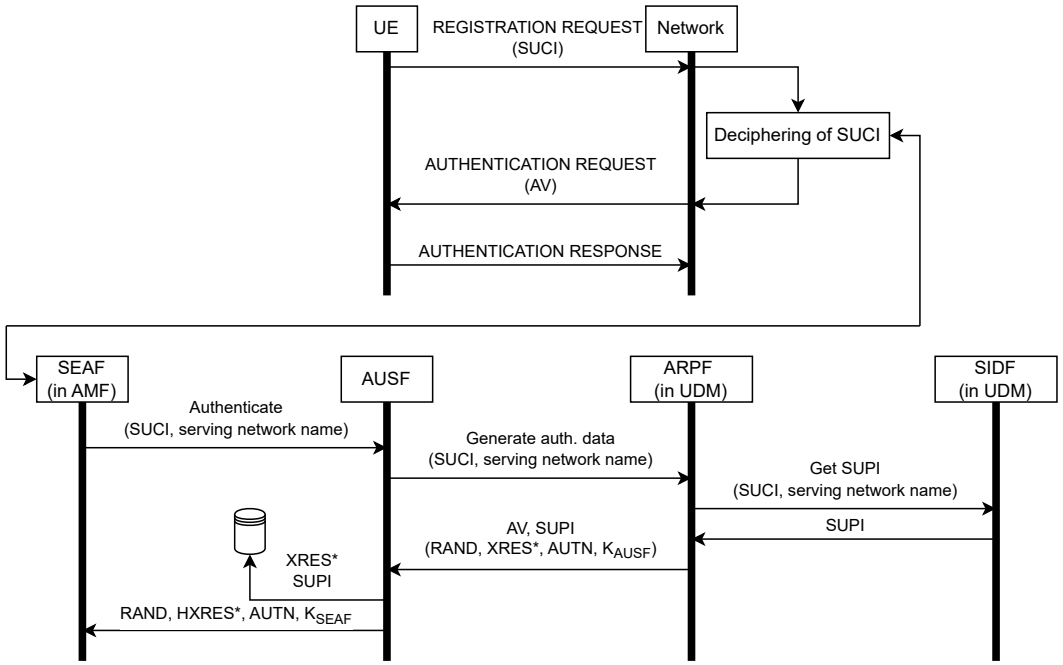

Figure 4.17: The UE `REGISTRATION REQUEST` triggering the 5G-AKA procedure, adapted from [Cox21].

4. Finally, this information is included in the `AUTHENTICATION RESPONSE`, which the UE sends to the AMF.

Upon receiving the `AUTHENTICATION RESPONSE` from the UE, the AMF of the SN will start the procedure shown in Figure 4.18 by hashing the received `RES*` to get `HRES*` and comparing it to the stored `HXRES*`. If these match, the AMF concludes that the UE has the correct value of K and can therefore be authenticated successfully. To fully confirm the authentication, the AMF of the SN sends the `RES*` to the AUSF of the HN. The AUSF compares `RES*` with `XRES*`, which should also match. If so, the HN also concludes that the UE can be authenticated, but also that the SN is valid. The match between `RES` and `XRES*` on the AUSF can only be achieved if the SN was truthful to both the HN and the UE about its own identity (as this is used by both the UE and AUSF to calculate `RES*` and `XRES*`). Now that the AUSF knows the SN is valid and the UE is authenticated, it answers with the SUPI of the UE, allowing the SN to calculate its own copy of K_{AMF}. Finally, the AUSF sends a confirmation to the UDM that the UE has been authenticated successfully. The UDM can then link that confirmation to subsequent procedures [Cox21]. Figure 4.18 shows these final steps of 5G-AKA.

After AKA has taken place, the UE and gNB can start setting up integrity and confidentiality protection using the agreed-upon keys, starting the security mode control procedure explained in Section 4.4.6

Authentication not Accepted by the Network

If the UE provides an `AUTHENTICATION RESPONSE` with an incorrect `RES*`, the network can handle this in two ways depending on the identity the UE provided [3GP24d]:

1. In the case of a SUCI, the network rejects the authentication and sends an `AUTHENTICATION FAILURE`. This will cause the UE to abort its registration.

2. In case of a GUTI, the network sends an `IDENTITY REQUEST` (see Section 4.4.7) to retrieve the UE's SUCI. After this, the network initiates the 5G-AKA procedure again, using the

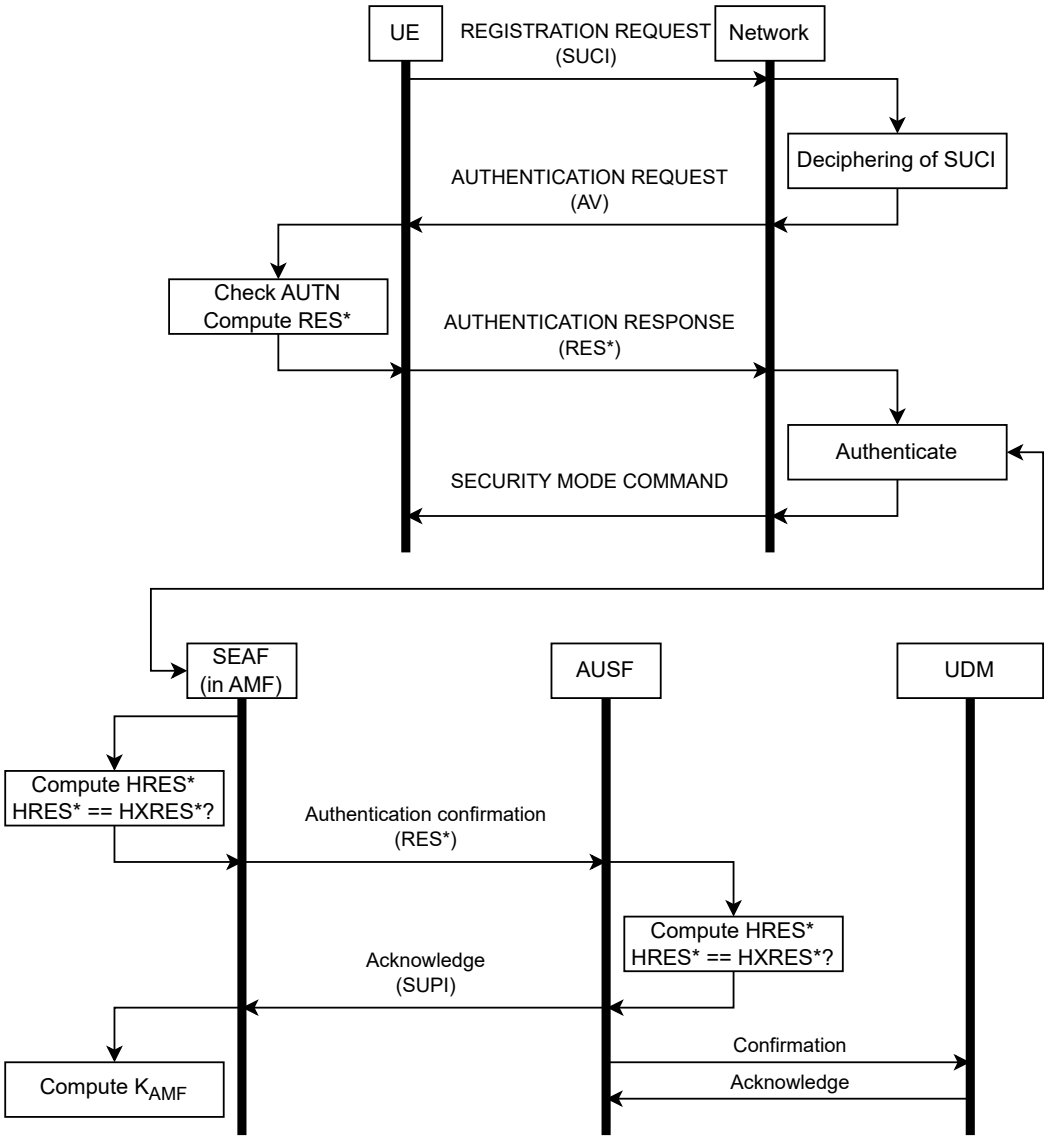

Figure 4.18: Handling the `AUTHENTICATION RESPONSE` sent by the UE, adapted from [Cox21].

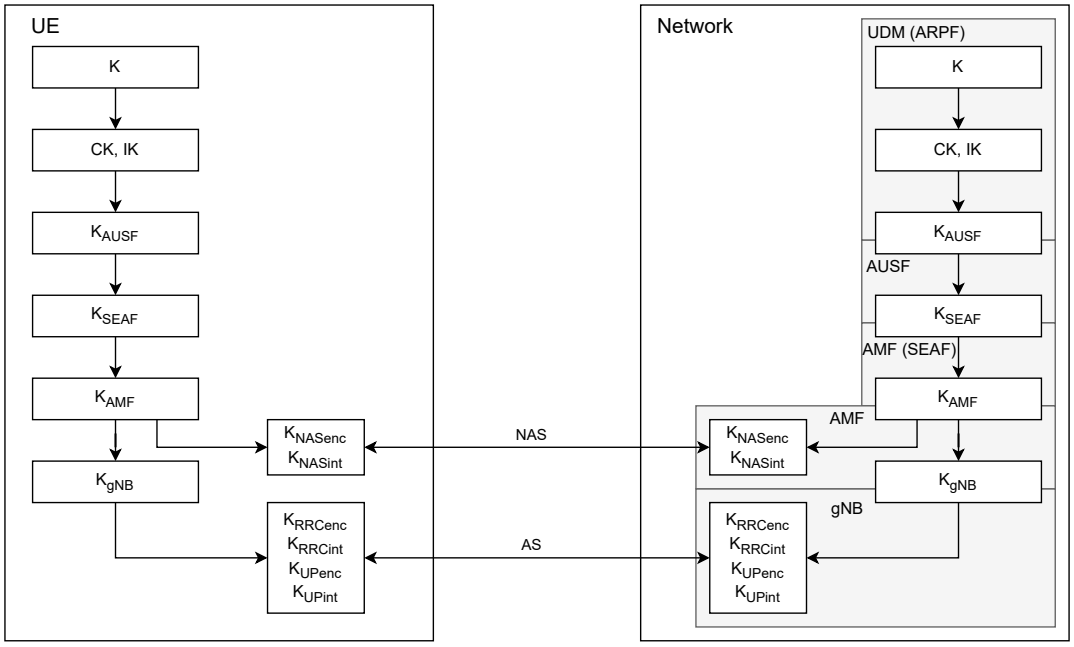

Figure 4.19: The network access key hierarchy, from [Cox21].

SUCI as described in the previous sections.

Authentication not Accepted by the UE

As previously mentioned, the UE checks data sent by the network in the `AUTHENTICATION REQUEST` (`RAND`, `AUTN`, and `ngKSI`) to verify the authenticity of the network it is trying to connect to. If incorrect parameters are provided, the UE can reject the core network. If it has to do so, the UE answers with an `AUTHENTICATION FAILURE` message, with a cause value describing what caused the failure [3GP24d].

There are four possible causes for authentication failure [3GP24d]:

1. MAC failure: if the MAC (contains the `AUTN` parameter) is invalid, the UE sends an `AUTHENTICATION FAILURE` with cause `#20: MAC failure`. This failure can happen when a UE receives an AV that was meant for a different SUCI, for example.
2. Non-5G authentication unacceptable: if the UE receives an `AUTN` which has the Authentication Management Field (AMF) separation bit set to "0" (meaning a connection to the 4G Evolved Packet System (EPS) [3GP24e]), the UE sends an `AUTHENTICATION FAILURE` with cause `#26: non-5G authentication unacceptable`.
3. ngKSI already in use: if the ngKSI sent by the core is already in use on the UE, it sends an `AUTHENTICATION FAILURE` with cause `#71 ngKSI already in use`.
4. Sequence number failure: if the UE detects an incorrect sequence number in the `AUTN`, it sends an `AUTHENTICATION FAILURE` back to the network with cause `#21 synch failure` as well as a re-synchronization token `AUTS`.

Key Hierarchy

During the AKA procedure, many keys are derived on both the UE and network side. These keys are generated in a fixed order, as these keys are all dependent on a predecessor. This order creates a so-called 'hierarchy of keys'. The generation of these keys relies on shared knowledge between the network and the UE of a user-specific key K. This key is found within

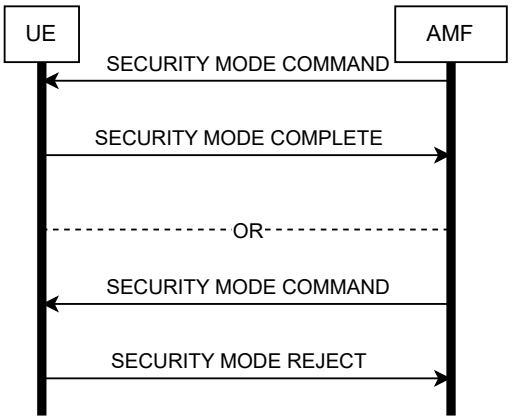

Figure 4.20: The security mode control procedure as per [3GP24d].

the ARPF as well as within the UE on the USIM. During the authentication procedure, both parties check with one another whether they have the right copy of K. If so, they compute a hierarchy of lower-level keys that they will use to secure their session, which is shown in Figure 4.19 [Cox21].

Starting at the top of the hierarchy with K, the UE and ARPF compute CK and IK, which are legacy 3G keys that are used to derive the next key in the chain: K_{AUSF}. K_{AUSF} then serves as the basis for K_{SEAF}, which in turn serves as the basis for K_{AMF} [Cox21].

K_{AMF} allows for the derivation of three keys: K_{NASint}, K_{NASenc}, which are used for the ciphering and integrity protection of NAS messages and K_{gNB} which gets sent to the primary gNB [Cox21].

K_{gNB} further allows for the derivation of two key sets: K_{RRCenc} and K_{RRCint}, which are used for the ciphering and integrity protection of RRC signaling and K_{UPenc}, K_{UPint} which are used for ciphering and integrity protection of UP session data [Cox21].

4.4.6 5GMM: Security Mode Control Procedure

The NAS security mode control has the responsibility of initializing and starting the NAS security (integrity and/or confidentiality) protection for the connection. The protection will be provided using the keys established for this purpose in the AKA procedure. When this procedure has finished, both parties will take a NAS security context into use that describes the keys, etc., used to protect the signaling [3GP24d].

The general message flow of the security mode control procedure can be seen in Figure 4.20.

The procedure is started by the AMF by sending a `SECURITY MODE COMMAND` message to the UE. The `SECURITY MODE COMMAND` is sent unciphered, but integrity protected using K_{NASint} that was established in the AKA procedure. Precisely which K_{NASint} was used will be described using the `ngKSI` so that the UE is able to use the same key. As the messages sent before this procedure were all unprotected, this is the first message that gets the 'integrity protected with new 5G NAS security context' value for the `security header type` field in the NAS message [3GP24d].

The following (most relevant) IEs are present in the `SECURITY MODE COMMAND` message [3GP24d]:

- `ngKSI`: informs the UE about which key set it should use.
- `Replayed UE security capabilities`: the AMF replays the security capabilities (supported algorithms, for example) it received from the UE during the registration procedure.

Since this message is now integrity protected, it can no longer be altered by a third party, such that if the UE verifies that these replayed capabilities match the ones it sent during registration, it knows they have not been tampered with.

- `Selected NAS security algorithms`: indicates the algorithms to be used for NAS integrity and ciphering.

When the UE receives the `SECURITY MODE COMMAND`, it performs an integrity check using the key set indicated by the `ngKSI` and a verification of the replayed security capabilities. Both these tests should pass for the UE to accept the message [3GP24d].

NAS Security Mode Command Accepted

When the `SECURITY MODE COMMAND` passes the tests done by the UE, the UE will start using the NAS security context that was described by the message to protect its signaling. Upon doing so, the UE responds with a `SECURITY MODE COMPLETE` message that is now fully protected using the integrity and ciphering algorithms and keys that were indicated by the AMF in the `SECURITY MODE COMMAND`. As this message is now both ciphered and integrity protected, this is the first message in the signaling with the `security header type` set to 'integrity protected and ciphered with new 5G NAS security context'. All subsequent UE messages will be ciphered and integrity protected with the selected algorithms and keys [3GP24d].

The following (most relevant) IEs are present in the `SECURITY MODE COMPLETE` message [3GP24d]:

- `IMEISV` The IMEISV of the UE if it was requested by the AMF in the `SECURITY MODE COMMAND`.
- `NAS message container`: contains a replay of the originally sent `REGISTRATION REQUEST` by the UE.

Upon receiving the `SECURITY MODE COMPLETE`, the AMF will start protecting all of its signaling with ciphering as well (the `SECURITY MODE COMMAND` was not yet ciphered, since the UE first needs to know which key set it should use for decryption). If the UE included a `REGISTRATION REQUEST`, the AMF will now use this version of the `REGISTRATION REQUEST` instead of the one it received during the initial registration procedure as the basis for further completing the registration. The `REGISTRATION REQUEST` sent by the UE before and after security activation does not need to match. The UE needs to determine which IEs can be sent without protection for its initial `REGISTRATION REQUEST` and expand this with the IEs that require protection in the security protected `REGISTRATION REQUEST` of this procedure. The purpose of this should be clear: ensuring that the AMF works with a fully protected `REGISTRATION REQUEST` instead of the one that could possibly have been manipulated before security activation [3GP24d].

NAS Security Mode Command not Accepted

If the UE determines that the `SECURITY MODE COMMAND` is invalid, the UE sends a `SECURITY MODE REJECT` message back to the AMF. This message contains a cause for the failure, with the following typical values [3GP24d]:

- `#23: UE security capabilities mismatch`.
- `#24: security mode rejected, unspecified`.

Cause #23 is used when the replayed security capabilities from the AMF do not match with the ones the UE sent during registration. This usually means they were altered when they were being sent, as those messages were not yet protected. Cause #24 is a catch-all in case an unspecified error occurs [3GP24d].

In either case, the AMF will stop the ongoing security mode procedure and both parties will resort back to the security context they used before the start of this procedure (if any) [3GP24d].

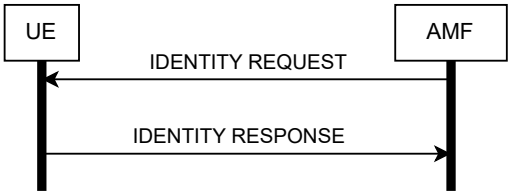

Figure 4.21: The identification procedure as per [3GP24d].

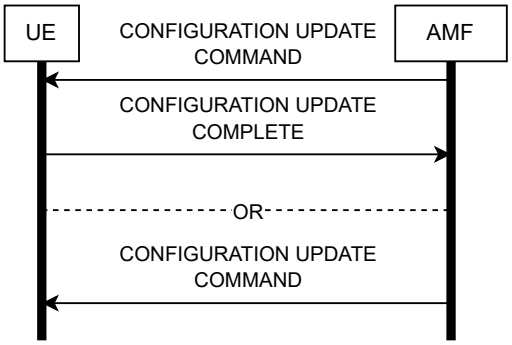

Figure 4.22: The generic UE configuration update procedure as per [3GP24d].

4.4.7 5GMM: Identification Procedure

The identification procedure can be used by the AMF to request a certain identity of a UE, such as the SUCI, IMEI, etc. [3GP24d].

This procedure's general flow of messages can be seen in Figure 4.21.

The procedure is started by the AMF by sending an `IDENTITY REQUEST` message to the UE, which describes which identity it would like to know in the `identity type` IE [3GP24d].

The UE will respond to these identity requests at any time while in the `5GMM-CONNECTED` state. In its response, the UE includes the requested identification parameters, generating a new SUCI for the response if it hasn't done so already. Responding to these messages does require that security has been established, however. If security is not yet established, the UE is only allowed to answer with its identity if the SUCI is requested [3GP24d].

4.4.8 5GMM: Generic UE Configuration Update Procedure

The UE configuration update procedure is used by the AMF to update the configuration of the UE with regard to 5GMM or 5GSM parameters by providing new values (such as a new 5G GUTI) using the `CONFIGURATION UPDATE COMMAND` message [3GP24d]. The procedure has more purposes than this, but those go beyond this thesis's scope.

The procedure is only started with a UE that already has a 5GMM context with the AMF and is currently in the `5GMM-CONNECTED` mode. In case the UE is idling, paging (see Section 4.4.9) can be used to force it back into `5GMM-CONNECTED` before initiating this procedure [3GP24d].

The general message flow of the procedure can be seen in Figure 4.22. The confirmation sent by the UE is optional and is in answer to the AMF actually requesting the UE to do so in the initiating message [3GP24d].

The AMF initiates this procedure by sending a `CONFIGURATION UPDATE COMMAND` to the UE in which it specifies the values for the parameters it wishes the UE to update [3GP24d]. Many

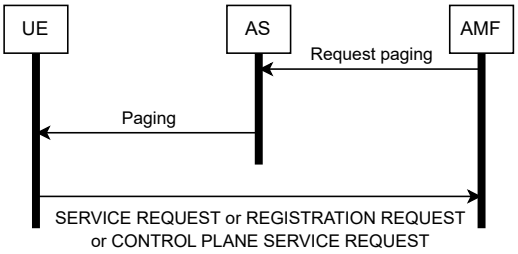

Figure 4.23: The paging for 5GS services procedure as per [3GP24d].

parameters can be altered, but with regard to access and mobility, the following are most important [3GP24d]:

- 5G GUTI.
- TAI list.
- Service area list.
- Mobility restrictions.

These parameters can be set by the AMF depending on different circumstances such as mobility changes or different network policies. Furthermore, this procedure can also be used trigger a re-registration procedure [Rom+19].

Upon receiving the `CONFIGURATION UPDATE COMMAND`, the UE will apply the updates as the AMF requests (e.g., delete its previous 5G GUTI if a new one was assigned, for example). If the AMF sets the acknowledgement bit of the command to 'acknowledgement requested', the UE will send a `CONFIGURATION UPDATE COMPLETE` message back to the AMF [3GP24d].

4.4.9 5GMM: Paging Procedure

The paging procedure is used when the UE has gone idle and no longer has an active NAS connection with the AMF (the RRC connection is also idling in this situation). The AMF uses paging to force the UE to re-establish a NAS connection and/or to re-establish the resources that are necessary for a PDU session, the UE had previously opened [3GP24d]. 5GMM is responsible for specifically requesting the gNB to page the UE. The actual paging message is sent from the gNB to the UE over RRC, without any NAS payloads.

The general message flow of the paging procedure can be seen in Figure 4.23.

The paging procedure is initiated by the network when there are NAS messages or user data messages waiting to be delivered to a currently idle UE. The AMF requests the gNB to send a paging message to the UE, forcing it to move from its idle state back to a connected state [3GP24d]. It is possible for there to be restrictions on when the AMF can initiate paging [3GP24d], but those go beyond the scope of this thesis.

When the UE receives a paging indication, it will respond to it with a service request or a registration request, depending on its internal 5GMM state and the access type, re-establishing the NAS connection [3GP24d].

4.4.10 5GSM: UE-requested PDU Session Establishment Procedure

The UE-requested PDU session establishment procedure allows a UE to set up a new PDU session with the SMF. This session can either be a new session with an external DN, or one that is carried over from an existing PDU session in the 4G EPS. There are more possibilities beyond those two, but those go beyond the scope of this thesis. If the network accepts the new PDU session, the UE and DN will receive a connection over which they can communicate

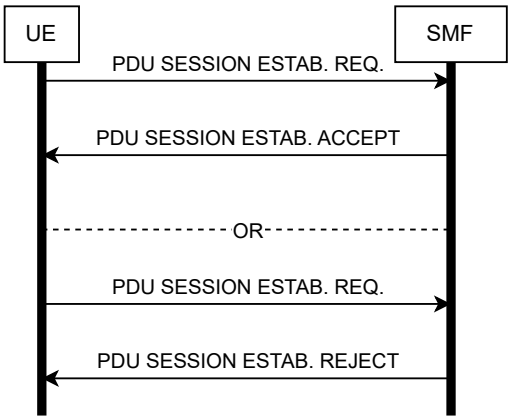

Figure 4.24: The UE-requested PDU session establishment procedure as per [3GP24d].

in both directions, with the UE receiving an IP address that it can use for this connection [3GP24d].

The general message flow of the UE-requested PDU session establishment procedure can be seen in Figure 4.24. Notice the shift in direction of the first message as opposed to the previously described procedures (UE-initiated procedure) as well as the SMF taking part in the communication, not the AMF (although the UE communicates with the SMF through the AMF, see Figure 3.3).

The UE initiates the procedure by sending a `PDU SESSION ESTABLISHMENT REQUEST` message to the SMF. To establish a new PDU session, the UE uses a PDU session ID that was previously unused [3GP24d]. In the `PDU SESSION ESTABLISHMENT REQUEST` message, the following relevant fields can be identified [3GP24d]:

- `PDU session ID`: the ID of the PDU session, should be a new value for a new PDU session.
- `Procedure Transaction Identity (PTI)`: distinguishes between different bi-directional message flows [3GP24c].
- `Integrity protection maximum data rate`: informs the network about the maximum supported data rate for the UE regarding UP integrity protection.
- `PDU session type`: describes the type of PDU session (IPv4, IPv6, IPv4v6, Ethernet, Unstructured).
- `5GSM capability`: indicates the UE capabilities with regards to various 5GSM features such as QoS.

If the requested PDU session can be accepted by the network, the SMF responds with a `PDU SESSION ESTABLISHMENT ACCEPT` message. The UE is granted an IP address if the type of PDU session requires it. Whether or not the PDU session establishment is acceptable to the network depends on factors such as service area restrictions or subscription restrictions, which are ignored in case of an emergency PDU session [3GP24d].

It is also possible, however, that the network rejects the connection with the requested DN. In that case, the SMF creates a `PDU SESSION ESTABLISHMENT REJECT` message. This message contains an IE describing the cause for the rejection [3GP24d]. The most important ones are [3GP24d]:

- #8: operator determined barring.
- #27: missing or unknown DNN.

- #28: unknown PDU session type.
- #29: user authentication or authorization failed.
- #50: PDU session type IPv4 only allowed (and #51, 57, 58, 61 for IPv6, IPv4v6, Unstructured and Ethernet respectively).
- #82: maximum data rate per UE for UP integrity protection is too low.

If the PDU session can not be established, the UE releases the PTI it used for this session and considers the PDU establishment as failed.

4.4.11 NAS Security

NAS security protection refers to both the ciphering and integrity protection of 5GMM and 5GSM messages. In essence, NAS security actually only protects 5GMM messages, but as 5GSM messages are sent through piggybacking on 5GMM messages, those are protected by extension. Protection of these messages is provided using the keys generated during the AKA procedure and the algorithms agreed upon during the security mode control procedure. In NAS security, integrity protection of NAS messages is a requirement. Ciphering, however, is an operator decision. It is perfectly possible and standard-conform behavior to have the AMF be configured to use the null ciphering algorithm, implying no ciphering protection for the signaling with the network using that AMF [3GP24d].

Handling of 5G NAS Security Contexts

To keep track of the set of keys, algorithms, etc. used to protect a connection, NAS security contexts provide a way to refer to all parameters for authentication, integrity protection, and ciphering at once using an ngKSI. Establishing the security context happens during the AKA procedure. Actually taking the security context into use happens during the security mode control procedure. When this procedure has been finished, the NAS signaling between the UE and network will be protected with both ciphering (possibly using the null scheme) and integrity protection, except for a few specific messages [3GP24d]. Before security activation, only the following messages are allowed to be sent/handled [3GP24d]:

- `REGISTRATION REQUEST`
- `REGISTRATION REJECT` if the cause is not #76, 78, 81 or 82.
- `IDENTITY REQUEST` in case of SUCI being requested.
- `IDENTITY RESPONSE` in case of SUCI being requested.
- `AUTHENTICATION REQUEST`.
- `AUTHENTICATION RESPONSE`.
- `AUTHENTICATION FAILURE`.
- `SECURITY MODE REJECT`.
- `DEREGISTRATION REQUEST`
- `DEREGISTRATION ACCEPT`.
- `SERVICE REQUEST`.
- `SERVICE REJECT` if the cause is not #76 or 78.

NAS COUNT and Sequence Number

A NAS security context will also maintain two `COUNT` fields: one for uplink NAS messages and one for downlink. These NAS `COUNT` counters use 24-bit internal representations, and both the

UE and AMF maintain a pair. The `COUNT` is made up of a NAS sequence number of 8 least significant bits and a NAS overflow counter of 16 most significant bits [3GP24d].

The NAS sequence number is used by both the UE and AMF in their signaling to one another. For each message that gets sent with protection, the sender should increase their sequence number (including for retransmissions). The overflow counter is incremented by one when the sequence number counter overflows and resets [3GP24d]. The NAS `COUNT` has a different meaning in relation to where it is stored [3GP24d]:

- The uplink `COUNT` on the UE is the value that will be used in the next uplink NAS message from the UE to the AMF.
- The uplink `COUNT` on the AMF is the `COUNT` that was in the last integrity checked message from the UE to the AMF.
- The downlink `COUNT` on the UE is the `COUNT` that was in the last integrity checked message from the AMF to the UE.
- The downlink `COUNT` on the AMF is the value that will be used in the next downlink message from the AMF to the UE.

The `COUNT` is used by NAS security to provide replay protection by adding the sequence number subpart of the `COUNT` to the NAS message. This ensures that no NAS message is handled twice by the receiver, as each sequence number inside a successfully integrity-checked message is handled only once [3GP24d].

Ciphering and Integrity Protection

The NAS `COUNT` value is also used as input into the protection algorithm in the form of the NAS sequence number and the NAS overflow counter as previously mentioned [3GP24d]. Besides the `COUNT` the following is part of the NAS integrity protection input [3GP24d]:

- The data to be protected.
- `BEARER ID`.
- `DIRECTION` bit.
- K_{NASint}.

Ciphering takes a very similar approach in which the only differing parameters are the K_{NASenc} instead of K_{NASint} and an extra input being the length of the keystream used for encryption [3GP24d].

4.4.12 Known Attacks on NAS: SUCI-catchers

As described in Section 3.3.3, 5G allows for the UE to send its permanent identity in an encrypted form, which allows the UE to identify itself within the network, without leaking its identity to others in the network. The SUCI prevents linkability by encrypting the IMSI, and doing so with every use to get a different result each time. The necessity for the SUCI-provided anonymity was shown by the well-known IMSI-catchers attack on 4G networks, and below which we will now discuss.

IMSI-Catchers

An IMSI-catcher or 'stingray' is a user-privacy focused attack, which allows an attacker to force the victim to leak its IMSI (permanent and unique identifier, see Section 3.3.1). As described in Section 3.3, the temporary identifiers are used to minimize the number of times a user exposes their IMSI, but there are still moments (such as the very first registration of the device) in which the UE has to use its IMSI. An IMSI-catcher exploits this by forcing such a scenario in which the user unknowingly leaks their permanent identity, allowing the attacker to track and

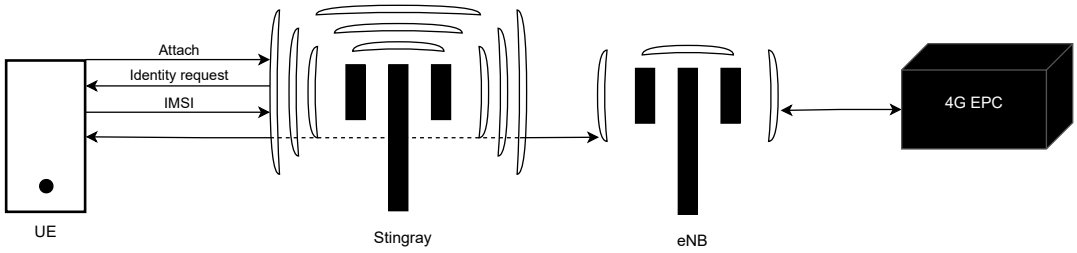

Figure 4.25: A stringray operating in camping mode.

pinpoint the user in the network [Pal+21]. Besides malicious actors, stingrays are also known to be employed by public authorities like law enforcement to identify and track down wanted criminals [Bat17]. IMSI catchers started as an expensive and large piece of equipment only accessible to governmental agencies, but have become both smaller and cheaper as time went on, making them a real privacy threat [Dab+14].

IMSI-catchers started being deployed as early as in 2G, exploiting the many security flaws of 2G. Particularly vulnerable for user data protection was the lack of mutual authentication, allowing the IMSI-catcher to tell the UE to disable encryption. Furthermore, IMSI-catchers exploit that a UE likes to connect to the cell with the strongest signal strength, fooling the UE into establishing a connection with the IMSI-catcher just by transmitting the information of the real network with a better signal strength than the real base station (hence the alternative name Cell Site Simulator (CSS)). Subsequent mobile network generations attempted to introduce fixes for these issues, such as mutual authentication between the network and UE, but IMSI catchers evolved with them, remaining relevant throughout the years [YCC19].

Figure 4.25 portrays a possible IMSI-catcher setup in which the attacker acts as a MITM while identifying the user. This is one possible setup, as there exist other versions that, for example, deny the UE access to the network through the CSS after fetching their identity, causing the UE to connect to the legitimate network after the rejection. An IMSI-catcher that holds the UE in its own cell by forwarding the user's data after identifying them acts under the 'camping mode' operation mode. A stringray that purely identifies the user and then releases them to the legitimate network is a stringray acting under the 'identification' mode [Dab+14].

Once the UE is connected to the fake base station, it will start the attachment procedure. Since the UE only just joined the network, no security has been established yet. As such, the first messages sent by the UE will not be encrypted. For its identifier during the attachment, the UE can use a temporary identifier or its permanent identifier (in 5G, using the IMSI during registration has been replaced with using the SUCI). When the UE uses its IMSI to connect to the network, it immediately leaks this to the attacker, completing the attack. Should the UE provide a temporary identifier, the attacker can circumvent this by sending an identity request for the IMSI (in 5G, this request has been replaced with requesting the SUCI), also completing the attack [Pal+21].

SUCI Catchers Description

One could then reasonably assume that 5G solved the issue of IMSI-catching by having the SUPI encrypted and never transmitting it in plain text and changing it with each transmission. However, [Chl+21] showed that this is still vulnerable to privacy attacks to a certain extent: where IMSI-catchers were able to actively request the identity of a user and verify their presence, SUCI-catchers can only answer the question if a specific, known subscriber is currently present in the proximity of the attacker. It should be noted that the more technical term for this type of attack is IMSI-probing: asking if a known subscriber is present. IMSI-catching refers to actually retrieving/storing the permanent identifier of a subscriber [Chl+21].

Besides the attack shown in [Chl+21], it should also be noted that SUCI encryption is not

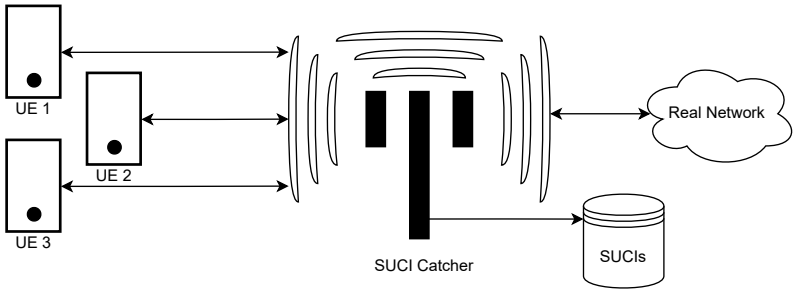

Figure 4.26: The SUCI-catcher discovery phase as described in [Chl+21].

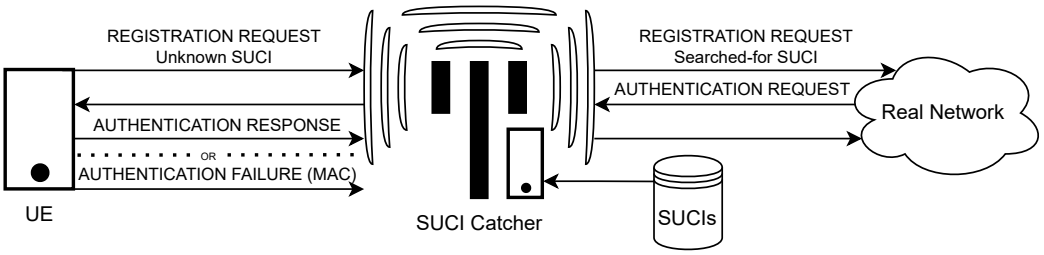

Figure 4.27: The probing step of the attack phase of the SUCI-catcher as described in [Chl+21].

mandatory and can be enabled or disabled depending on the operator's configurations and device/USIM capabilities [3GP24d]. In case the encryption is disabled, 5G becomes just as vulnerable to IMSI-catching as 4G since an unencrypted SUCI is equivalent to the SUPI.

The SUCI-catcher attack exploits the semantics of the `AUTHENTICATION REJECT` by sending the UE AV's generated by the AKA procedure that belong to different SUCI's that the attacker gathered beforehand. Since the generation of the AV happens using the user's permanent identifier behind the scenes (see Section 4.4.5), only the UE whose SUPI was used to create the SUCI is able to answer the AV positively. Other UE's will respond with an `AUTHENTICATION REJECT: cause #20: MAC failure`, thereby inadvertently leaking that they are not that specific user [Chl+21].

SUCI Catchers Implementation

The attack is implemented in two phases: discovery and attack. The discovery phase is the first phase of the attack in which the attacker stores any SUCI of interest that it later would like to query for its presence. The attacker can do this in two main ways: either extracting the SUCI out of a `REGISTRATION REQUEST` sent by the UE when connecting to the network (requires that the UE does not use a temporary identifier, which it should prioritize, see Section 4.4.4), or by sending an `IDENTITY REQUEST` for the SUCI, which is allowed before security activation (see Section 4.4.7 and 4.4.11). After fetching the SUCI's they need, the attacker can move on to the attack phase [Chl+21].

The attack phase consists of two substeps, shown in Figures 4.27 and 4.28. The reset step performs a simple 5G AKA between the UE and the network to prevent the UE from disconnecting after multiple `AUTHENTICATION FAILURE`s. This reset step is executed before each SUCI probe to maintain connectivity, as it was discovered that the tested UE stopped answering authentication requests after two consecutive authentication failures. [Chl+21].

The probe step alters the content of the transmitted `REGISTRATION REQUEST` by the UE by inserting the searched-for SUCI into the identity field. The network will generate an AV for this SUCI and put this in its `AUTHENTICATION REQUEST`. As seen in Section 4.4.5, the UE

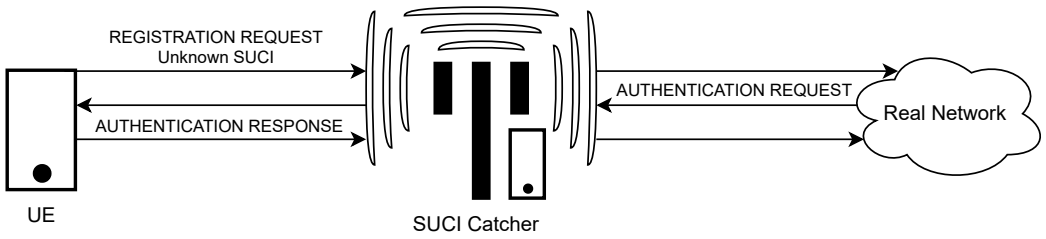

Figure 4.28: The reset & sync step of the attack phase of the SUCI-catcher as described in [Chl+21].

will reject the `AUTHENTICATION REQUEST` with cause `#20: MAC failure` if the AV was not as expected. Only the UE to whom the used SUCI belongs can answer the AV generated for that SUCI. In this way, the UE accidentally leaks its identity, because if it replies with an `AUTHENTICATION RESPONSE`, we know the UE is the subscriber whose sniffed SUCI we inserted, compromising their identity [Chl+21].

4.4.13 Known Attacks on NAS: Key Reinstallation

As described in Section 4.4.11, 5G NAS uses encryption based on keystreams in which multiple parameters generate the keystream block. These parameters are the 128-bit user key `K`, 32-bit `COUNT`, 5-bit `BEARER`, 1-bit `DIRECTION` and the `LENGTH` [FSP24]. This method of encryption should be secure as long as no cryptographic material is leaked. Research done in [FSP24] shows that despite this security, there exist vulnerabilities that allow for specific signaling to leak the keystream used by the UE: the NReplay attack.

NReplay is a key reinstallation attack, which is an attack that attempts to reinstall an already in use key to weaken a One-time Pad (OTP) encryption scheme. The attack was introduced in [VP17], where the four-way handshake of 802.11i was attacked by forcing the session key to be installed multiple times over, resetting the nonce and replay counter, opening up a data-confidentiality attack.

The NReplay attack discovered two vulnerabilities in NAS security that allow for a similar type of attack. These two vulnerabilities are:

1. Keystream reuse: When the UE has to retransmit the `SECURITY MODE COMPLETE` message, its keystream does not change. Furthermore, it resets the NAS `COUNT`, causing it to reuse the keystream for the next message that has to be security protected [FSP24].
2. Unclear specifications: Due to a lack of clarity in the specifications and operator interpretations, it is possible for the AMF to not immediately reject messages that failed the integrity check. For example, when the AMF receives a `DEREGISTRATION REQUEST` that fails the integrity check, it will inspect the cause that was given with the message. If the cause of the `DEREGISTRATION REQUEST` was a switch off, the AMF ignores the request. If, however, the deregistration request was not caused by a switch off, the AMF authenticates the subscriber after which it processes the request [FSP24].

NReplay Implementation

To perform this attack, a MITM is required that can forward messages in both directions as well as store these messages and stop forwarding them on-demand [FSP24]. If these requirements are met, the following steps (shown in Figure 4.29) should be followed [FSP24]:

1. Forward all signaling between the UE and network until the first `REGISTRATION REQUEST`, which is not yet ciphered.
2. Store this `REGISTRATION REQUEST` and forward this and all subsequent traffic.

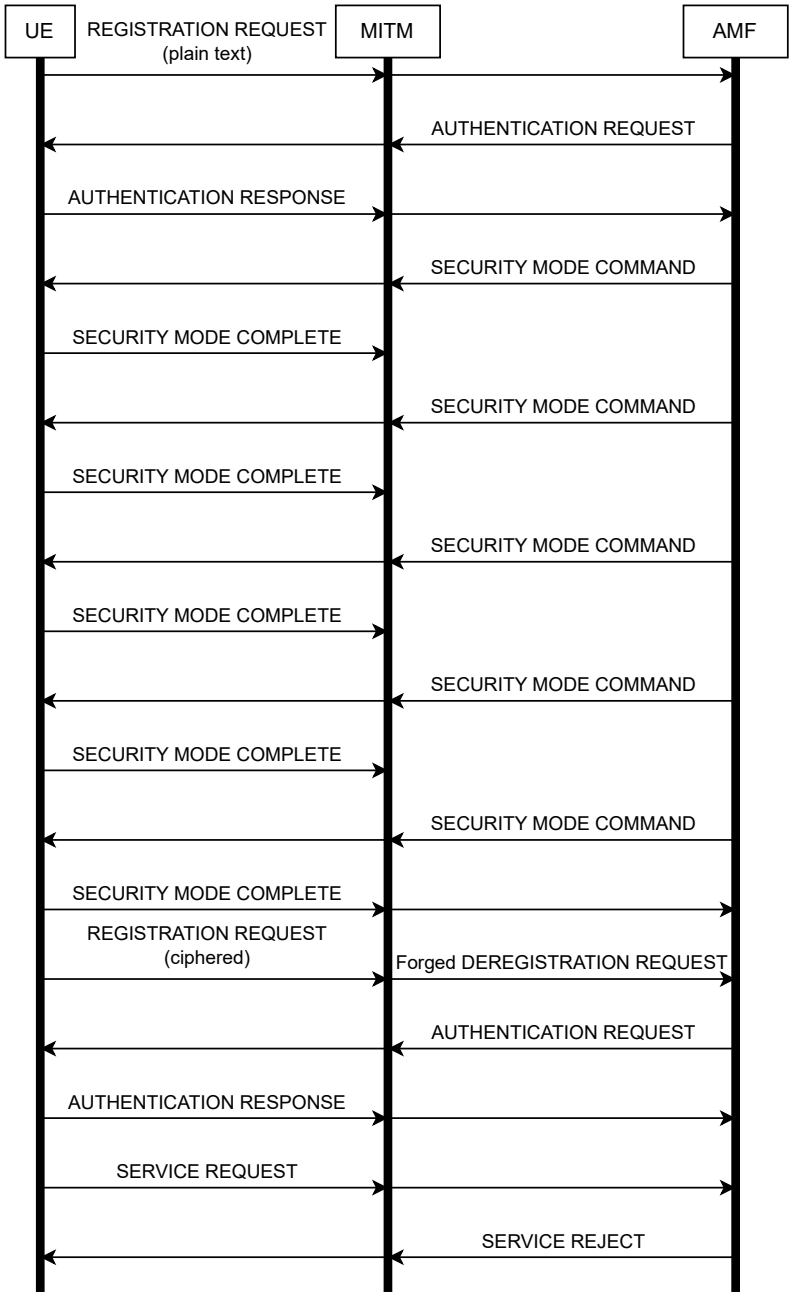

Figure 4.29: The NReplay message flow, from [FSP24].

3. When the message to forward is the `SECURITY MODE COMPLETE`, stop forwarding traffic until the message has been retransmitted four times by the UE. In this scenario, the UE will, as previously described, not change its keystream and keep resetting `COUNT`, such that the next message it sends will reuse the same keystream.
4. Allow the fifth `SECURITY MODE COMPLETE` to pass through.
5. Due to the failing `SECURITY MODE COMPLETE` transmissions, the UE will now retransmit the `REGISTRATION REQUEST` using the keying material that was continuously reset. This `REGISTRATION REQUEST` should be stored by the attacker as well, but not forwarded.
6. If the attacker now XOR's the originally sent plain text `REGISTRATION REQUEST` with the retransmitted, ciphered `REGISTRATION REQUEST`, they obtain the keystream used by the UE.
7. Using this keystream, the attacker sends a ciphered `DEREGISTRATION REQUEST`, which will fail integrity verification on the AMF. As mentioned, however, the AMF might not drop this message but initiate an authentication procedure which the MITM should forward. On successful authentication, the AMF will accept the deregistration, causing a Denial of Service (DoS) on the UE with service rejects.

Chapter 5

5G Man-In-The-Middle Implementation

5.1 Introduction

The overview of possible attacks on the different layers in the 5G protocol stack highlights the power a MITM setup gives an attacker. Often, an attacker requires the ability to inject or alter the content of messages while they are being transmitted, without immediate loss of connectivity as was the case in all of the discussed attacks. A MITM setup allows an attacker to achieve this by ingesting traffic, possibly manipulating it, and then forwarding it to its original destination.

High level, a MITM attack can be seen as a setup in which a malicious party (the attacker) puts themselves in between the communication of two unknowing parties (the victims). This access allows the attacker to listen in on the communication, manipulate the messages, or disturb the communication [CDL16]. An abstract visualisation of a MITM setup can be seen in Figure 5.1. As the MITM in our setup interacts both with the legitimate UE and the legitimate core network, the MITM is made up of two parts: a rogue gNB and a rogue UE to handle traffic as seemingly legitimate parties for the victims to connect with.

To the best of our ability, we haven't found a generic 5G MITM implementation that can be used for general experimentation and security research. The only implementations we could find seemed to be tailored for a specific attack that was the subject of that particular research and did not seem publicly accessible. The work done in this thesis provides an implementation of a generic MITM setup that can be configured to act as both a passive and active MITM, allowing the user to specify multiple operations on incoming messages as well as craft messages to inject them into the communication.

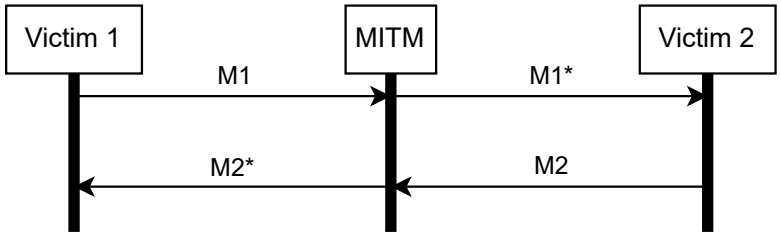

Figure 5.1: High level view of a MITM forwarding messages, while possibly altering them [CDL16].

5.2 GNB Implementation Details

5.2.1 Software Stack

The rogue gNB was implemented using the open-source srsRAN Project 5G gNB software stack (henceforth called 'srsRAN'). SrsRAN provides a gNB implementation that at the time of writing follows the 3GPP Release 17 specifications as well as the Open Radio Access Network (O-RAN) Alliance specifications. The full gNB stack is implemented, with most relevant procedures and functionalities present [Sys25n]. The source code for srsRAN can be found at [Sys25m].

The most important features supported by the gNB implementation are [Sys25e]:

- 3GPP release 17 alignment.
- All RRC procedures.
- All MAC procedures.
- Software Defined Radio (SDR) support.

There were alternatives to srsRAN that could have been chosen as well. OpenAirInterface also provides a comprehensive open-source implementation of a 5G RAN, which can be found at [Ope25b] and [Ope25c]. Besides OpenAirInterface, a third option was UERANSIM [GÜN25]. This implementation, however, does not fully provide a physical layer implementation. The 5G NR radio interface is only partially implemented and simulated over the UDP [GÜN25]. Based on the well-rounded features provided by srsRAN as well as the well-written documentation and deployment experience [Mam+23], we chose this framework over OpenAirInterface.

For network simulation purposes (detailed in Section 5.4.2), a vanilla install of srsRAN was used to simulate a legitimate gNB in the network.

5.2.2 SrsRAN Project Architecture

SrsRAN has an intricate architecture that can be seen in its entirety in Figure 5.2. It integrates a Central Unit (CU)/Distributed Unit (DU) split, with a further division into Central Unit - Control Plane (CU-CP), Central Unit - User Plane (CU-UP), Distributed Unit - High (DU-High), and Distributed Unit - Low (DU-Low). The CU/DU split was originally part of Release 15 and is further specified by the O-RAN Alliance specifications. Figure 5.3 shows the changes made in Release 15 compared to LTE. The specific split shown in Figure 5.2 is the O-RAN Alliance defined option 7.2x, which is a "low-level split for URLLC and near-edge deployments" [Sys25g].

At a basic level, the 5G protocol stack is divided across the two units as shown in Figure 5.4.

DU-low

Starting at the bottom, the DU-Low handles uplink and downlink traffic. In case of DU-Low, 'handling' means executing the processing that should happen in the upper part of the PHY layer. DU-Low contains only the upper PHY layer and has two interfaces for communicating with the outside world. DU-Low communicates directly with DU-High through the Functional Application Platform Interface (FAPI) interface [Sys25d].

DU-high

The DU-High handles both uplink and downlink traffic, just like DU-Low. DU-High handles the MAC and RLC processing of this traffic [Sys25c]. DU-High has three components that together implement this portion of the protocol stack [Sys25c]:

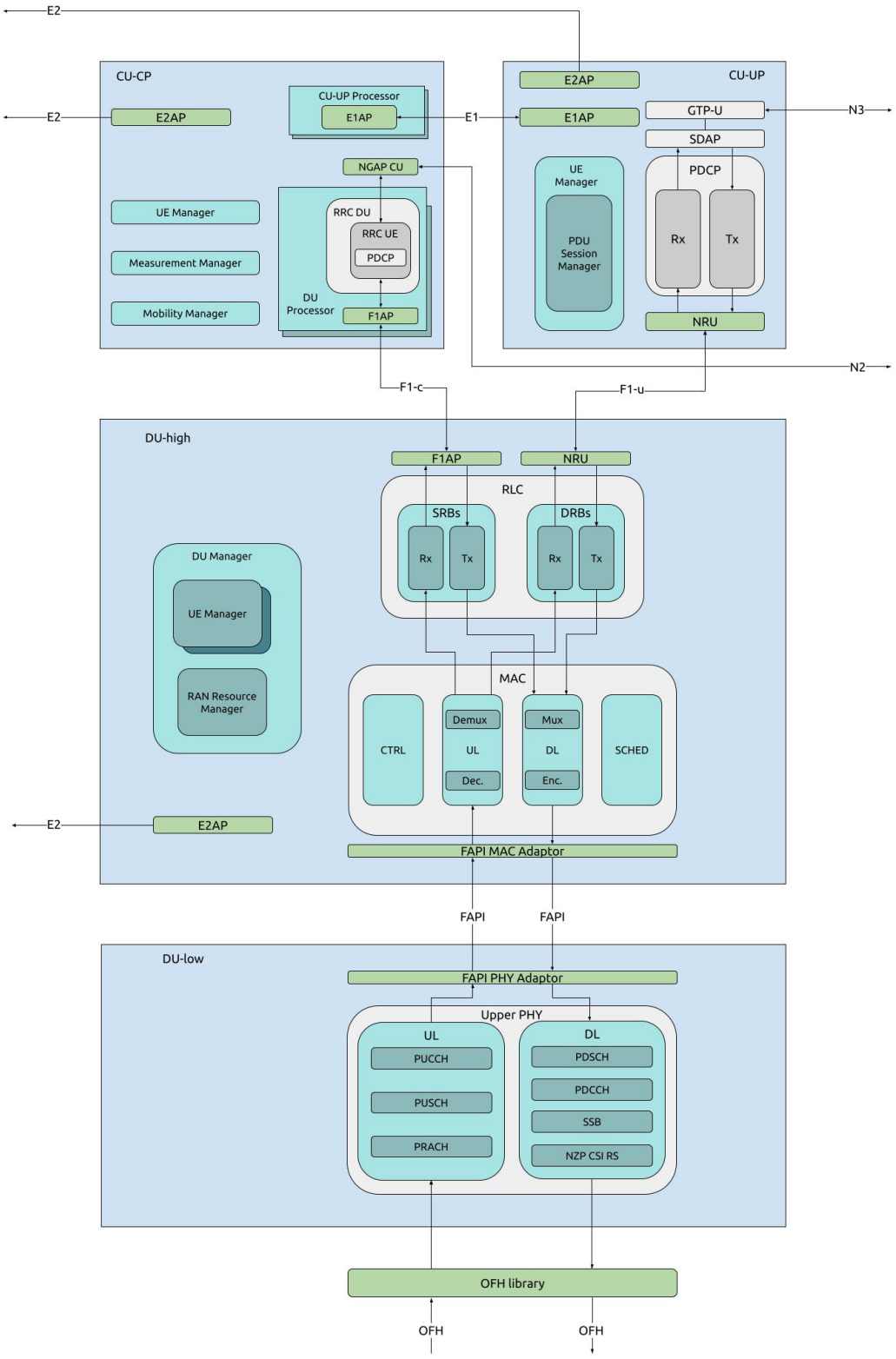

Figure 5.2: The srsRAN architecture, from [Sys25h].

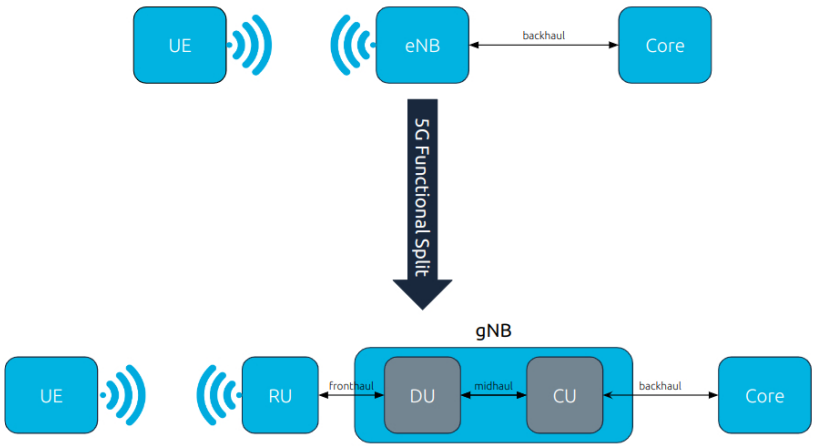

Figure 5.3: The CU/DU split introduced by Release 15, from [Sys25g].

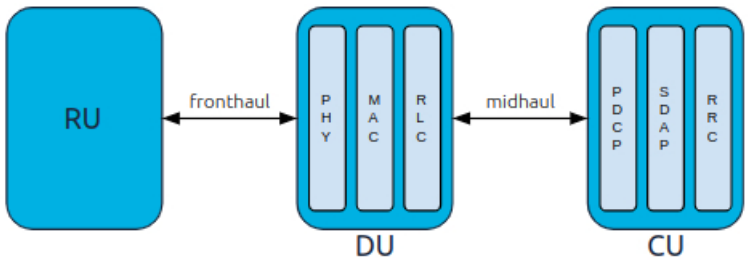

Figure 5.4: The protocol stack divided across the CU/DU split, from [Sys25g].

- DU Manager: manages the DU itself, meaning the connected UE's and the RAN resources.
- RLC: The RLC component implements the RLC functionalities of the protocol stack.
- MAC: The MAC component implements the MAC functionalities of the protocol stack, as described in Section 4.1.

The MAC component is further divided into four sub-components [Sys25f]:

- MAC controller: the MAC controller translates configuration requests (like adding new DU cells, for example) into commands that can be given to the MAC sub-components. The MAC controller also ensures optimal operation of its components, minimizing latency and avoiding race conditions.
- RACH handler: manages the allocation of RNTI's when UE's execute the RACH procedure.
- MAC uplink processor: decodes the incoming MAC PDU's and forwards them to their local channels. Also forwards the uplink buffer status reports.
- MAC downlink processor: manages the MAC scheduler for transmitting downlink messages.

CU-UP

The CU-UP handles all UP traffic, focusing on PDCP and SDAP [Sys25b]. This part of the protocol stack is implemented using four main components [Sys25b]:

- UE manager: manages connected UEs. Its responsibilities include: adding and removing UEs, providing UE information to other processes, and communicating with the CU-CP, DU, and core.
- GPRS Tunneling Protocol - User Plane (GTP-U): transports UP traffic to and from the UPF in the core.
- SDAP: manages QoS requirements on UP traffic.
- PDCP: handles UP data before and after it enters DU-High as described in Section 4.2

The PDCP component is further divided into two sub-components: 'TX' and 'RX'. These components are split according to whether they handle transmission data (TX) or received data (RX) [Sys25b].

CU-CP

The CU-CP is responsible for handling CP messaging, specifically PDCP, RRC, and NGAP [Sys25a]. This part of the protocol stack is implemented using five main components [Sys25a]:

- CU-CP processor: handles each CU-UP connected to the CU-CP as multiple CU-UPs could be connected to one CU-CP (one control plane connection per UE, possibly multiple UP connections).
- DU processor: handles DUs connected to the CU-CP, with each DU having its own DU processor with included F1 Application Protocol (F1AP), PDCP, and RRC procedures. Each UE connected to the DU also has its own RRC instance.
- UE manager: manages connected UEs. Its responsibilities include: adding and removing UEs, providing UE information to other processors, and communicating with the CU-UP, DU, and core.
- Measurement manager: manages cell measurements.
- Mobility manager: manages UE mobility.

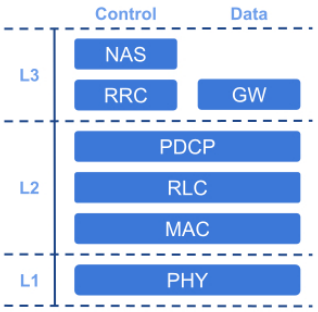

Figure 5.5: The srsUE architecture, from [Sys25o].

The DU processor contains the functionalities needed to provide a working PDCP, RRC, and NAS implementation. The PDCP implementation packs/unpacks outgoing and incoming messages, respectively. RRC handles the semantics of these messages, starting necessary procedures, etc. If the RRC message was a regular uplink information transfer, the message gets forwarded to the AMF using the NGAP implementation. As further explained in Section 5.4.2, the DU processor is where the bulk of the MITM implementation is active.

5.3 UE Implementation Details

5.3.1 Software Stack

The rogue UE was implemented using srsRAN 4G. SrsRAN 4G is an open-source 4G radio suite that is developed by Software Radio Systems, like the srsRAN Project. This suite includes a full-stack UE that is both 4G and 5G (prototyping) capable (henceforth called srsUE), a full-stack eNB, and an EPC [Sys25j]. As this thesis focuses on 5G, only the UE implementation was used.

The most important features supported by the UE implementation are the following [Sys25k]:

- Prototype 5G NSA and SA support.
- Soft USIM support
- Hard USIM support via PC/SC.
- SDR support.

As with the gNB, some alternatives could have been used to implement the UE. OpenAirInterface, for example, provides a UE in its previously mentioned RAN implementation. UERANSIM also provides a UE with the previously mentioned physical layer limitations. We maintained a uniform platform for a smoother development experience and went with srsUE.

For network simulation purposes (detailed in Section 5.4.2), a vanilla install of srsUE was used to simulate a legitimate UE in the network.

5.3.2 SrsRAN 4G - UE Architecture

Compared to the srsRAN architecture, the srsUE architecture is much simpler: a basic implementation of each protocol layer according to its responsibilities. The visual representation given in the documentation can be seen in Figure 5.5, which exactly matches the 5G protocol stack as described in the introduction of Chapter 4. As the implementation closely adheres to the protocol stack, the description of each component will overlap with each layer's previous descriptions.

L1

Starting at the bottom, the PHY layer handles moving the MAC messages over the air interface. Responsibilities include cell search, and cell measurement [Sys25o]

L2

The MAC layer translates data coming from the logical channels into transport blocks that get moved to the PHY layer on transport channels. The reverse is also true: handling messages coming from the PHY layer, to move them to the correct logical channel. Besides this, the MAC layer handles control and scheduling information as well as retransmissions, error correction, and priority management of logical channels [Sys25o].

The RLC layer has three possible modes of operation: transparent, unacknowledged, and acknowledged. The UE in our setup runs the conventional acknowledged mode. RLC manages the logical channels or bearers, each operating in one of the aforementioned modes. Running in acknowledged mode means that RLC will perform concatenation, segmentation, reassembly, reordering, and duplicate detection on PDUs as well as retransmission of missing data [Sys25o].

The PDCP layer handles the security protection (ciphering and integrity protection) and transfer of CP and UP traffic. Besides that, the PDCP layer also implements duplicate discarding and in-sequence delivery to/from the RRC and Gateway (GW) layers [Sys25o].

L3

The L3 architecture is split into two subparts: CP and UP. The CP is implemented in two layers: RRC and NAS. The UP is implemented in the GW layer [Sys25o].

The RRC layer implements the CP signaling between the UE and its eNB or gNB. This layer also parses the SI sent out by the network and also handles the RRC connection in general, including the security keys for RRC security protection [Sys25o].

The NAS layer implements the communication between the UE and the core network (4G EPC or 5GC). This layer controls PLMN selection, network attach procedures, identification, authentication, etc [Sys25o].

Finally, the GW layer allows srsUE to provide a data plane by creating and managing a TUN virtual network kernel interface [Sys25o].

5G Prototyping

The documentation describes the UE provided by srsRAN 4G as having "prototype 5G NSA and SA support" [Sys25k]. In practice, this means the implemented layers all have an LTE and NR version that gets used depending on the network generation the UE is connected to. In case of NSA, the UE will utilize the LTE implementation for CP signaling and the NR implementation for UP signaling, and in case of SA, the UE will utilize all NR versions of the implemented layers.

As this implementation is merely meant for prototyping, not all features are supported as in LTE. An example of this is paging, which is only supported in LTE and is not yet present in the NR RRC implementation as of writing this thesis.

5.4 MITM Implementation

The MITM tool provides two operating modes: passive and active. In a passive setup, the MITM transparently forwards messages with the option to store messages that were part of the communication. A transparent MITM can be useful for reading messages before security activation, as these messages are unencrypted, without altering the messages by either party,

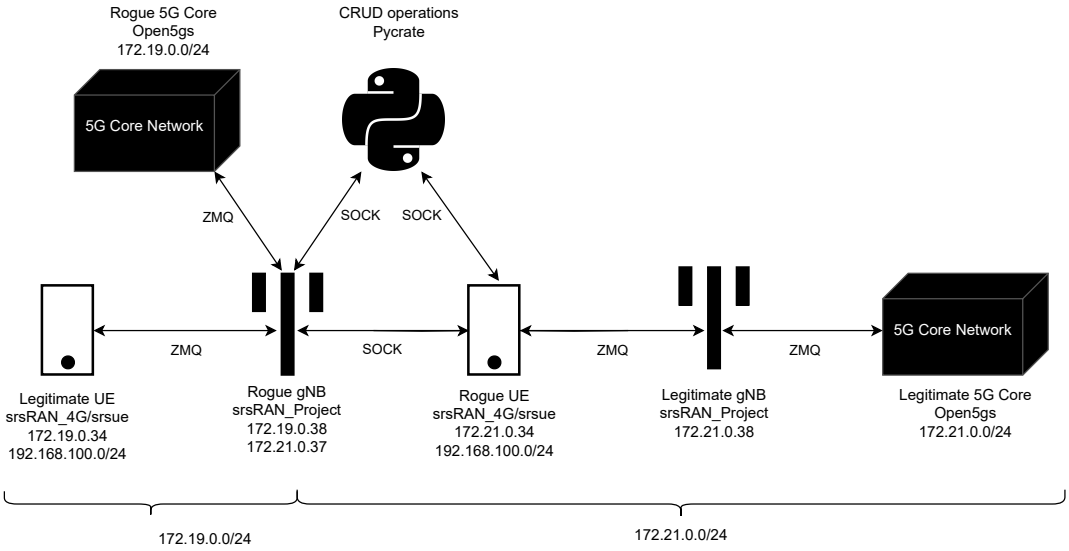

Figure 5.6: The ZMQ MITM setup architecture.

which could disrupt communication. An active MITM can influence the course of the communication in the network by altering what was sent by either party or by injecting messages as if they were sent as part of the communication, possibly triggering different procedures that would otherwise not occur.

5.4.1 Malicious Attachment

For the MITM to be active in the network, it first needs to force UE's to connect to it instead of the legitimate gNB's in the network. Appearing as a gNB of the legitimate network is done through spoofing the MIB and SIB's broadcast by the legitimate network, making the rogue gNB seem as a gNB part of that network.

Broadcasting the correct SI is not enough to force a UE to connect to the rogue gNB, however. The MITM can achieve this by surpassing signal strength threshold requirements on a UE, offering a better quality of service, which entices the UE to connect [BP22].

More specifically, the UE will connect with the rogue gNB in two different ways depending on the RRC state it currently finds itself in [BP22]:

- `RRC_IDLE`: in `RRC_IDLE`, cell (re)selection happens, in which case the UE can be lured in on its own by transmitting a stronger signal than the UE's current connection.
- `RRC_CONNECTED`: in `RRC_CONNECTED`, the UE will perform measurements of signal strength and report them to the current gNB through measurement reports. If the measured malicious signal strength exceeds the threshold for handover, the handover procedure will be triggered, causing the legitimate gNB to pass the subscriber on to the rogue gNB.

To remain within scope, malicious attachment was simulated by configuring srsUE to immediately connect to the rogue gNB. Over-The-Air (OTA) malicious attachment was simulated by manually forcing the Commercial-Off-The-Shelf (COTS) UE to connect to the test network instead of its regular carrier as described in [Sys25l].

5.4.2 MITM Architecture

The created setup consists of two UEs, two gNBs and two core networks, each running in a separate Docker container. The socket between the rogue gNB and rogue UE acts as a bridge

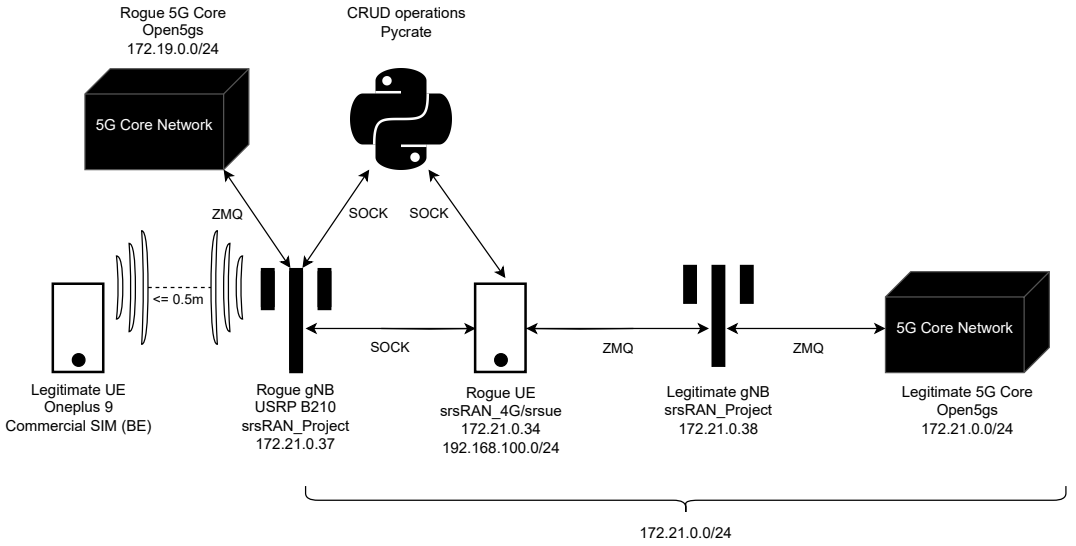

Figure 5.7: The OTA MITM setup architecture. A distance of less than or equal to 0.5m was used between the UE and SDR when testing.

between the two Docker networks, making it necessary for the UE to go through the MITM to connect with the legitimate core network, proving its forwarding functionality.

We provide a 5G SA setup through the open-source 5GC implementation provided by Open5GS [Ope25a]. The legitimate core network contains the subscriber data the legitimate UE uses on a softsim. The rogue core network was added as a tool that the rogue gNB could use in case it needed functionalities already implemented by the AMF, for example, but remained unused in our tests.

The connection between the legitimate UE and rogue gNB can be made in two ways: we can use ZeroMQ (ZMQ) RF simulation [Zer25] as well as true OTA using SDR devices. Our setup was tested with both options, using ZMQ and srsUE for some tests and using both USRP B210 and USRP X300 devices for OTA tests. These tests are possible using a COTS smartphone, as described in [Sys25l].

The connection between the legitimate UE and rogue gNB was made using ZMQ as shown in Figure 5.6 as well as OTA using a Oneplus 9 smartphone as shown in Figure 5.7. All other connections were made using ZMQ.

Commercial Networks

When using a COTS UE with a hard USIM, it is also possible to connect this device to a commercial network using a second SDR device for the rogue UE. The connection between the COTS UE and rogue gNB will happen OTA as previously described, and the rogue UE will attach OTA using the data sent by the legitimate UE. This setup does require that 5G SA is offered at the testing location, however. As previously described in Section 2.1 this is not always the case. During our testing we also found this to be an issue, as only 5G NSA option 3 (see Section 2.3.2) was available in our region.

SrsRAN is focused on NR, and as such we only offer NR as a supported RAT. SrsUE was made with 4G in mind, though, and as such also has (albeit limited) support for 5G NSA. In that regard, we considered translating 5G signaling provided by the legitimate UE to 4G signaling which the rogue UE could use to attach to the network. A basic translation between a `REGISTRATION REQUEST` and `ATTACH REQUEST` has been provided and will be used by the rogue UE in its attempt to connect to the NSA network as a proof of concept. The steps necessary

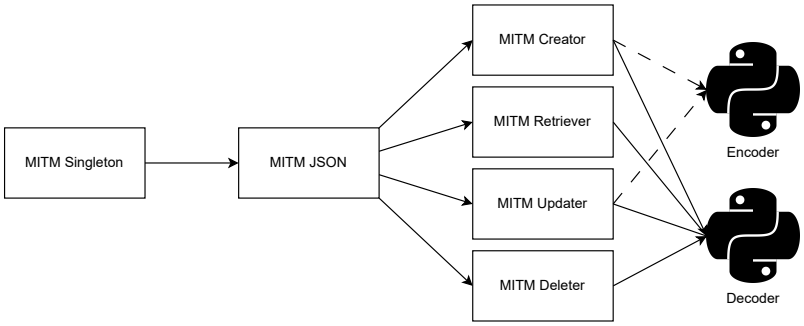

Figure 5.8: The design of the MITM code that was added to the existing implementations.

to facilitate translation were implemented on both the UE and gNB, but no translation was provided for messages other than the `REGISTRATION REQUEST`.

CRUD architecture

Two Python instances run in the background of the rogue gNB and rogue UE in order to provide support for the Create, Retrieve, Update, Delete (CRUD) operations the MITM provides on messages passing through. These Python instances communicate with the gNB and UE through different sockets on the same IP address, with a different port for each supported operation. A more detailed look at the CRUD implementation can be seen in Figure 5.8.

A central singleton provides the general MITM functionalities such as sending a message from the rogue gNB to the rogue UE and gets instantiated in different parts of the srsRAN and srsUE source code. The MITM singleton provides a JSON interface, which in turn links to an implementation of each of the four CRUD operations and provides some general JSON functionalities.

The Python encoder application executes functionalities such as altering the fields inside a given message and encoding it again afterwards. All CRUD operations requiring a message conversion from JSON format to bytes are handled by the encoder. The Python decoder application executes functionalities such as determining the message type given a hexadecimal bytestring. All CRUD operations requiring a message conversion from bytes to JSON are handled by the decoder. Both applications utilize Pycrate [Pyc25] for these operations.

5.4.3 Forwarding

Forwarding messages as a MITM between the rogue UE and gNB requires us to break away from the regular protocol stack control flow at some point, as otherwise these messages would be handled like a regular gNB or UE implementation would do. Deciding where to break away from the regular control flow is a decision that is governed by striking a balance between the potency of the MITM (i.e., what it is able to do with the messages it intercepts) and general connectivity (i.e., maintaining the connection with the victim UE). The MITM implemented in this thesis has been designed to be active on the PDCP layer after carefully considering the NAS and RRC layers as alternatives. We also provide the NAS version as a separate implementation, but this was not used in any of our experiments.

To motivate our decision for the PDCP layer, consider the following: the higher in the protocol stack we decide to maintain the regular implementation, the more the rogue gNB and rogue UE are themselves responsible for executing parts of the protocol stack. For example, if we decide to forward on the NAS layer, the rogue gNB and legitimate UE will maintain an RRC connection between the two of them, as will the rogue UE and legitimate gNB, as shown in Figure 5.9. This causes an issue in our setup, however, as the keying material necessary to

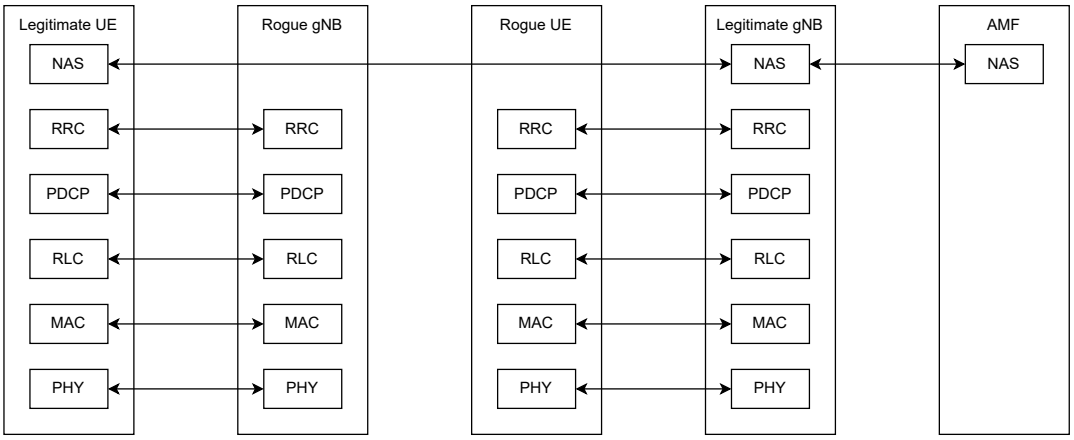

Figure 5.9: If we forward only the NAS messages, a separate RRC, PDCP, etc. connection exists between the rogue components and legitimate components, with their own sequence numbers, etc.

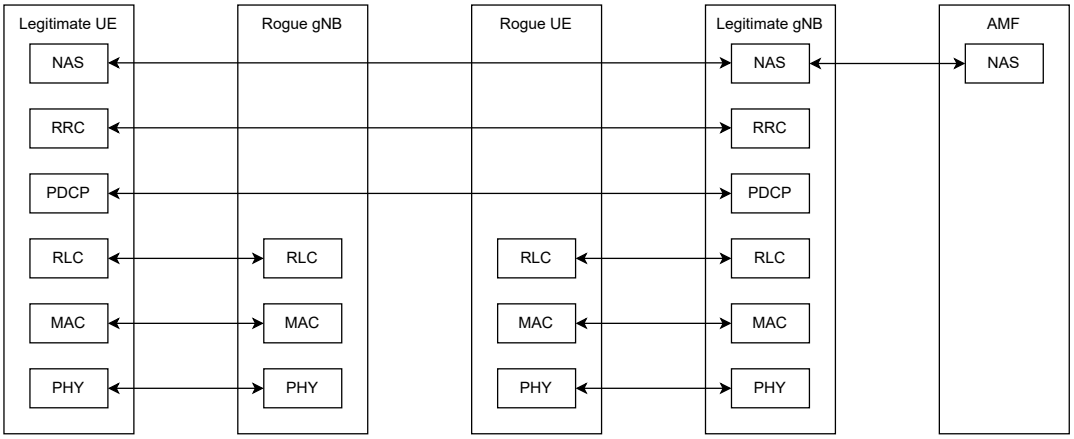

Figure 5.10: If we forward from the PDCP layer, the legitimate parties are responsible for maintaining the sequence numbers, etc. of that connection, including calculating the PDCP MAC field, which requires keying material agreed upon with NAS procedures.

execute security protection is delivered through NAS messages, which need to be interpreted by the component that should execute the protection. Since the rogue components forward these messages instead of interpreting them, they do not possess the keying material necessary to calculate the MAC used to integrity protect PDCP messages as described in Section 4.2.3. In this case, the legitimate UE will be expecting a MAC that can only be calculated by the legitimate gNB, but since the PDCP connection is maintained between the legitimate UE and rogue gNB, which does not possess the required keying material, the MAC will not match its expected value, and the connection will fail.

The same goes for forwarding on the RRC layer, as still the required keying material is only present on the legitimate UE and gNB that interpreted the NAS signaling in which the security configuration was agreed upon.

To deal with this issue, the MITM has been designed to be active on the PDCP layer. Forwarding PDCP messages means that the legitimate parties are the ones responsible for maintaining the NAS, RRC, and PDCP connection (see Figure 5.10), making them the ones responsible for calculating the PDCP MAC, which, since both parties possess the right keying material, will be correct, meaning signaling will no longer fail after security activation. As such, we decided to

develop the MITM functionalities on this layer. PDCP forwarding does not prohibit the MITM from being active in the network (for example, altering NAS message contents, etc.), provided we package the NAS messages inside RRC messages before re-packaging them with PDCP as further detailed in Section 5.6. Forwarding on the PDCP layer does have two downsides as well:

1. RRC and NAS are not interpreted on the rogue gNB and UE, meaning they will not configure any DRB's during the RRC reconfiguration procedure (see Section 4.3.8), making the MITM only functional for CP signaling.
2. Injecting or removing messages means the PDCP sequence numbers will no longer match on the legitimate parties, causing a connection failure when security is activated. This happens because the party that receives the injected message will increment its receive window, not expecting the lower sequence number that will be used by the spoofed party. Removed messages alter the window in the opposite direction, increasing the sequence number the original sender uses, while not incrementing the receive window on the would-be receiver.

Solving the problem of not supporting the UP is possible by moving all the way down the protocol stack and having the MITM forward over MAC. Since MAC messages no longer distinguish between DRB's or SRB's, but only which transport channel should be used (see Section 4.1.2), these messages will not suffer the problems PDCP forwarding has in that regard. Moving to the MAC layer does come with its own problem though: this makes it relatively difficult for the MITM to be anything more than a forwarding tool, prohibiting it from being active in the network. RLC sequence numbers become a problem, and the MITM would have to unpack MAC, RLC, PDCP and RRC to reach the NAS messages transmitted by the legitimate parties.

As forwarding over PDCP means full CP support without loss of connectivity, with the option to manipulate the signaling, we opted for this layer.

5.4.4 Supporting SIB 6

In the implementation, support for broadcasting SIB 6 was added, as this is not present in the latest version of srsRAN as of writing this thesis. This support was implemented by updating multiple internal srsRAN classes to support a new SIB as most classes handling SIBs only supported SIB 1, 2, and 19. The creation of an earthquake and tsunami warning message can be seen in the snippet below. The creation of this snippet was inspired by a GitHub discussion found at [Wil25]. Supporting this message is a feature only provided by our tool and is not present in vanilla srsRAN. SrsUE also does not support receiving these messages, meaning a COTS UE is required to verify the transmission of the messages as we have done.

```
static sib6_info create_sib6_info() {
  sib6_info sib6;
  sib6.msg_id = "0001000100000010";
  sib6.serial_num = "0011000000000001";
  sib6.warning_type = {0x05, 0x80};

  return sib6;
}
```

5.5 Passive Man-In-The-Middle

When the MITM has been configured to only act passively (i.e., no CRUD operations were specified except for `RETRIEVE`), all messages will be forwarded transparently without alterations.

In the uplink, messages will arrive on the PHY layer, and will be handled as srsRAN would

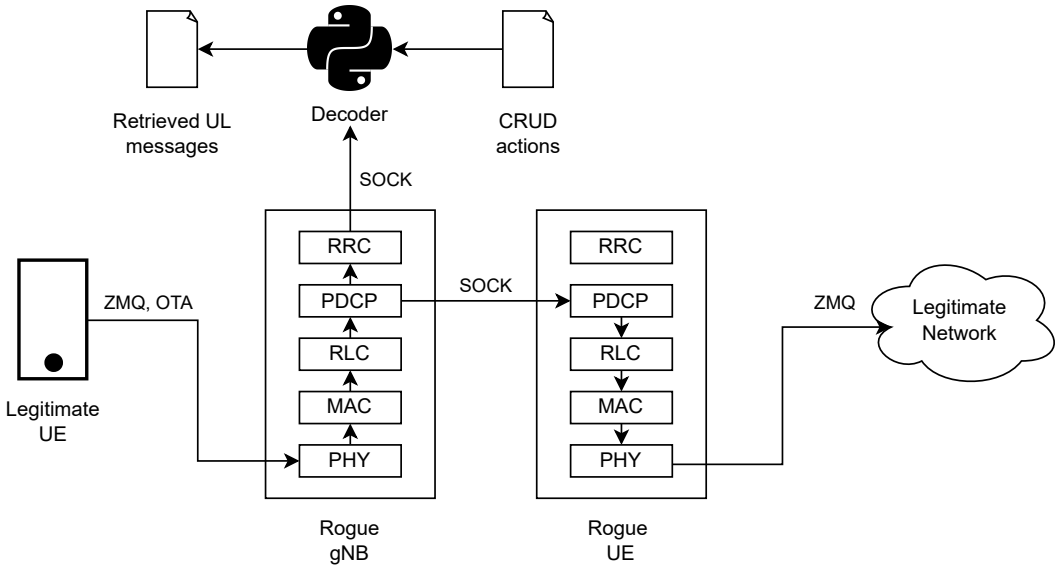

Figure 5.11: The path an uplink message follows after being transmitted by the legitimate UE in case of a passive MITM configuration.

normally do so. This happens up until the PDCP implementation, where normally the message would be unpacked and the typical RRC handling would happen. In the rogue gNB, we halt the regular execution here and forward the message in its PDCP form over the socket to the rogue UE. Before doing so however, we utilize PDCP's unpacking functionality on a copy of the PDU such that this can be forwarded to the decoder to check if the message should be stored according to the given actions. Figure 5.11 shows the path an uplink message takes in case of a passive configuration.

The downlink path follows the same steps, allowing the srsUE implementation to execute its regular steps until the PDCP unpacking happens. We halt the execution, forward the message over the socket, and check if the message should be stored.

5.5.1 RETRIEVE Action

The MITM can be configured to retrieve messages as long as they are not ciphered. As soon as ciphering has activated, identifying the message type is no longer possible, and therefore the MITM cannot identify whether this message should be stored. By default, ciphering is not activated on an srsRAN and srsUE setup, however, allowing this setup to execute `RETRIEVE` actions on all signaling.

One can configure the MITM to retrieve messages on the RRC and NAS layers as follows:

```
{
    "0": {
        "retrieve": {
            "msg": "ulInformationTransfer",
            "layer": "RRC"
        }
    },
    "1": {
        "retrieve": {
            "msg": "5GMMAuthenticationResponse",
            "layer": "NAS"
        }
```

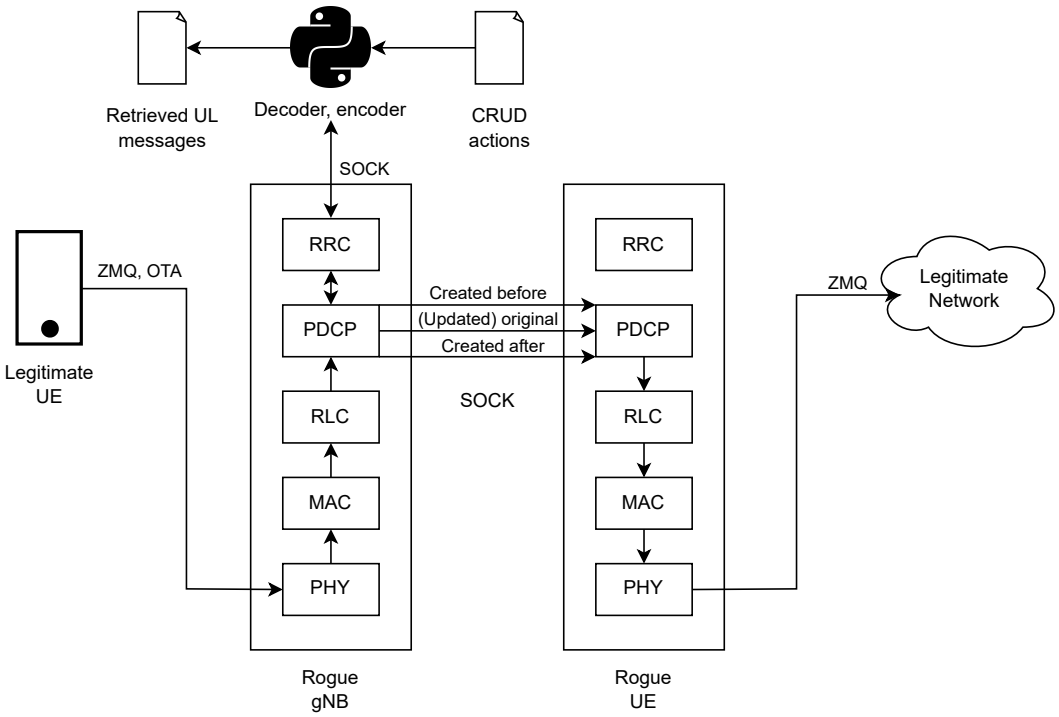

Figure 5.12: The path an uplink message follows after being transmitted by the legitimate UE in case of an active MITM configuration.

```
    }
}
```

When the decoder determines that a message should be saved, it gets exported in JSON format to a file containing all stored messages in the specific direction.

5.6 Active Man-In-The-Middle

The MITM also supports taking a more active role in the network in the form of different actions it can execute upon the messages it has to forward. Figure 5.12 shows what a message's path looks like in the case of an active configuration, if the user did not specify to delete the message. If the user determined the message should be deleted, the flow stops at the PDCP layer.

The messages follow the same path as in the passive setup, going up the protocol stack until the PDCP layer. PDCP unpacks the incoming message, which is then decoded using Pycrate. After decoding, the message will be checked for its type to determine if messages need to be inserted before or after it, with the given JSON content. If so, the messages that need to be inserted before the original one get encoded using Pycrate and are given the PDCP sequence numbers as expected by the legitimate network. That is, the messages that should be inserted before the original message receive the PDCP sequence numbers starting at the PDCP sequence number of the original message.

After inserting messages before the original one, the original message can also be updated. That is, the user can specify fields of the original message that should be altered with the contents they provide. This can range from simple fields such as replacing some UE capabilities to replacing the entire NAS PDU inside an RRC uplink information transfer. If at least one `CREATE` action happened before handling the original message, the PDCP sequence number

of the original message is updated to reflect the new amount of messages that came before it.

After handling the original message, new insertions can be made that should be transmitted after it. These insertions will once again have a sequence number that is as expected by the legitimate network or UE and their creation will affect the subsequent sequence numbers as well. In fact, if ever one `CREATE` operation was done, all subsequent PDCP sequence numbers will be updated throughout all following signaling to maintain connectivity.

It should be noted, however, that exercising active MITM actions (especially `CREATE`) will mean an inevitable loss of connectivity if either of the legitimate UE and network checks for integrity, or they enable ciphering. In case of integrity checking, it will be noticed that the PDCP MAC is no longer correct due to alteration of the PDCP sequence number, meaning that the message should be dropped. Furthermore, after enabling integrity protection, the MAC should not be empty for regular messages. Since neither the rogue gNB nor UE has possession of the integrity keys, it is not possible to compute a (valid) MAC when repackaging the messages after updating their sequence number, causing the messages to be dropped once again.

`CREATE` Action

The MITM can be configured to inject messages into the network as long as messages should not be ciphered or integrity protected (the rogue gNB and UE do not possess the correct keys to cover this case). The user can decide when the message is transmitted by specifying before or after which message type the message should be created. An amount can be specified to determine how many times this message should be crafted on separate occasions. Creating a message can be done through both a NAS-bytestring and a JSON representation of the NAS or RRC message. As mentioned, the PDCP sequence number of the message will be taken care of internally in the MITM, reflecting the fact that an injection occurred in later PDCP sequence numbers by incrementing them.

An example of creating an encoded NAS message can be seen below:

```
{
    "0": {
        "create": {
            "after": "5GMMAuthenticationResponse",
            "layer": "RRC",
            "amount": 1,
            "msg": {
                "c1": {
                    "ulInformationTransfer": {
                        "criticalExtensions": {
                            "ulInformationTransfer": {
                                "dedicatedNAS-Message": "7e00572d1..."
                            }
                        }
                    }
                }
            }
        }
    }
}
```

`UPDATE` Action

Besides injection, the MITM is also capable of message alteration, provided the messages are not ciphered or integrity protected as previously explained. The user can configure the MITM

to update the contents of a message by specifying which type of message should be altered, which fields of said message should be altered, and how many times an instance of such a message should be changed.

The user is able to granularly decide which fields to update, but must define all subfields starting at the point of alteration (see example below). There is no limit to how drastic a change can be, making it possible for a user to even alter the entire NAS PDU of an RRC information transfer without issue.

An example of updating a `REGISTRATION REQUEST`'s SUCI field can be seen below. Notice the definition of all subfields starting at the node we wish to alter (`SUCI_IMSI`):

```
{
    "0": {
        "update": {
            "layer": "NAS",
            "replace": "5GMMRegistrationRequest",
            "amount": 1,
            "fields": {
                "SUCI_IMSI": [
                    {
                        "PLMN": "00f110"
                    },
                    {
                        "RoutingInd": "0000"
                    },
                    {
                        "spare": 0
                    },
                    {
                        "ProtSchemeID": 0
                    },
                    {
                        "HNPKID": 0
                    },
                    {
                        "Output": {
                            "MSIN": "2143658769"
                        }
                    }
                ]
            }
        }
    }
}
```

`DELETE` Action

Finally, the active MITM configuration also allows for specifying types of messages that should not be forwarded after they are received on the rogue gNB or rogue UE. The user specifies how often these messages should be deleted and on which logical channel the message can be found.

An example of how one can delete messages can be seen below:

```
{
    "0": {
        "delete": {
```

```
            "layer": "NAS",
            "msg": "5GMMAuthenticationResponse",
            "lcid": "1"
        }
    }
}
```

5.7 Implementing Known Attacks and Experimentation

5.7.1 SUCI Catcher

5G increased subscriber privacy protection by implementing a concealed identifier (SUCI) that can be used by the UE to identify themselves without revealing their permanent identifier (SUPI). Only the operator can decrypt the identifier; therefore, attackers cannot derive the permanent identity of a user. Utilizing the semantics of legitimate messages in the network, however, an attacker can still answer the question of 'is user X currently present in the network?' even if they don't know the SUPI of the user [Chl+21].

By storing a SUCI once used by a user of interest, an attacker can utilize this SUCI to let the AMF generate an AV contained within an `AUTHENTICATION REQUEST`, which the UE will answer with either an `AUTHENTICATION RESPONSE` in case the utilized SUCI belonged to the UE, or with an `AUTHENTICATION FAILURE` with cause `#20: MAC failure` in case the SUCI never belonged to the UE. The attacker does this multiple times using all SUCIs that interest them, possibly finding that the UE is one of the searched for subscribers, meaning that the person of interest is present in the network. [Chl+21].

Section 4.4.12 provides a more detailed description of this attack.

Normal Behavior

Under normal circumstances, the legitimate UE would receive only one `AUTHENTICATION REQUEST`, containing an AV that was generated by the AMF using the SUCI provided by the UE in the `REGISTRATION REQUEST` (or a different identifier if applicable, as described in Section 4.4.4). As this AV was generated based on the deconcealed SUPI of the UE (detailed in 4.4.5), the UE will accept the AV and respond with an `AUTHENTICATION RESPONSE`.

Vulnerability

Suppose an attacker manages to force the AMF to send an `AUTHENTICATION REQUEST` containing an AV for a different UE to the legitimate UE. In that case, the UE will respond with an `AUTHENTICATION FAILURE`, with cause `#20: MAC failure` as this doesn't match with what the UE expected. In doing so, however, the UE exposes that they are not the UE that would respond positively, hereby answering the question 'are you X?'. By storing the SUCI of a person of interest and abusing this system, an attacker is able to probe whether that person of interest is present in the network, violating their privacy [Chl+21].

MITM Tool Implementation

To test this attack, we used both the ZMQ setup shown in Figure 5.6 and the partial OTA setup shown in Figure 5.7.

The attack is done in two phases, with three steps. The first phase is the discovery phase, in which the MITM registers the different SUCIs of UEs that connect to it. The tool supports this step by using the `RETRIEVE` functionality to store any `REGISTRATION REQUEST` or `IDENTITY RESPONSE`, which should contain the SUCI of a UE. Soliciting an `IDENTITY RESPONSE`

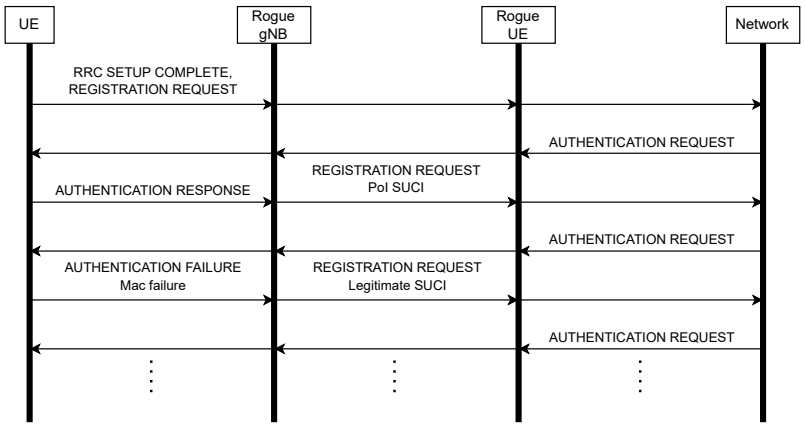

Figure 5.13: The observed signaling of a test of the SUCI catcher attack using the setup shown in Figure 5.6.

can be done through the `CREATE` functionality, to create an `IDENTITY REQUEST` for the UE's SUCI.

Phase two will see the MITM use the SUCIs it sniffed in order to answer the question if the owner of a particular SUCI is present in the network. The tool also supports this phase of the attack as follows: the probe step is supported by using the `UPDATE` functionality on an incoming `AUTHENTICATION FAILURE` or `AUTHENTICATION RESPONSE`, to convert it into a `REGISTRATION REQUEST` containing the SUCI that must be tested. This will trigger an AV in the network, to which the UE will respond with either an `AUTHENTICATION RESPONSE` in case the SUCI belonged to the UE, or an `AUTHENTICATION FAILURE` in case it didn't. The reset can be done through using the `UPDATE` functionality again on an `AUTHENTICATION FAILURE`, but using the SUCI of the UE that is connecting instead of a SUCI that must be tested. This will generate a correct `AUTHENTICATION REQUEST` for the UE, to which it responds with a `AUTHENTICATION RESPONSE`, resetting the amount of failed authentications.

Signaling

Figure 5.13 shows the signaling representing the steps of the SUCI catcher attack using the MITM tool. The rogue gNB updates the incoming messages according to the sequence in which they were defined in the JSON configuration.

Results

The SUCI catcher attack is fully implementable using our tool, with Figure 5.13 showing signaling we observed during one of our ZMQ tests. In the tests we executed, the MITM is able to successfully update the `REGISTRATION REQUEST`s sent by the UE in order to evoke `AUTHENTICATION REQUEST`s from the core, prompting the UE to reveal its identity through rejections of the authentication challenges. These results are exactly as described in the paper, showing that no specialized implementation of a MITM is necessary to achieve these results and that the vulnerability still exists.

Contrary to the results of the paper, we observed that srsUE did not require a reset stage to keep responding to the `AUTHENTICATION REQUEST`s sent by the core, with no obvious limit to the amount of authentication failures the UE tolerates. This could indicate that the reset stage may be unnecessary depending on the victim device, although our OTA tests with a COTS UE did also confirm a reset stage was necessary for that particular device as it stopped responding after two consecutive authentication failures, as described in the paper. Running the reset stage on srsUE even though it does not require it, did not impede the overall signaling of the attack,

so always prepending this stage does not seem to be an issue for achieving its goals, but not executing this stage if it was unnecessary would increase the number of identities that can be tested in a given timeframe substantially.

We also observed the MAC failures as a response from the COTS UE to `AUTHENTICATION REQUEST`s sent for a SUCI that did not belong to the UE, further confirming that this attack is still viable on commercial devices.

A final observation was made in that the COTS UE does not actually use a SUCI to register with the network. This is further described in Section 5.7.4, which shows that the UE uses a plaintext SUCI (e.g. its SUPI) as its identity. In this scenario, the SUCI catcher attack is in fact obsolete, as the regular IMSI catcher attack (see Section 4.4.12) suffices to learn the user's actual identity instead of answering the question "are you X?". Even so, the use of a null scheme SUCI by the UE does not change the implementation or results of the SUCI catcher attack, as this simply means the SUCI does not need to be decrypted first to retrieve the SUPI necessary to generate the AV of the `AUTHENTICATION REQUEST`.

5.7.2 ETWS Spoofing Attack

The ETWS system is a warning system designed to notify all UEs in a specific area of an emergency like earthquakes or tsunamis as described in Section 4.3.12. If an attacker manages to abuse this mechanism, they could spread mass panic among the affected users. Besides that, the suppression of warning messages could endanger users who are now uninformed about imminent disasters [BP22].

the paging system is used by the PWS to re-establish the connection between the UE and the network, putting it back in the `RRC_CONNECTED` state [BP22].

The PWS relies on paging to re-establish the connection between the UE and the network, putting it back in the `RRC_CONNECTED` state [BP22]. "Once active, the UE can receive warning messages transmitted through SIB messages" [BP22], specifically SIB6, 7, and 8. SIB6 carries small amounts of data that should immediately be picked up on by a user, to warn them of the emergency [BP22].

A more detailed description of this attack can be found in Section 4.3.12.

Implementing this attack in our MITM was done through adding support for SIB 6 to srsRAN, to allow it to transmit these messages. Furthermore, the suppression of paging messages meant to wake the legitimate UE will inherently be achieved by the rogue UE, as the NR implementation of RRC in srsUE does not support paging yet. If it did, the paging messages could easily be halted using the `DELETE` operator.

Normal Behavior

Under normal circumstances, the legitimate UE would receive the paging messages required for it to leave the `RRC_IDLE` and `RRC_INACTIVE` state, meaning it would be listening actively for messages including the SIB 6 transmission by the legitimate gNB in case of an emergency. Furthermore, under no circumstances should a legitimate UE receive disingenuous warning messages from any other party than legitimate authorities.

Vulnerability

Due to the unprotected nature of the SIB 6 transmissions, an attacker could transmit these and create disingenuous warning messages that could spread panic among regular users. Furthermore, by blocking the paging messages, the UE will not be able to receive legitimate warning messages sent out by the network in case of an emergency [BP22].

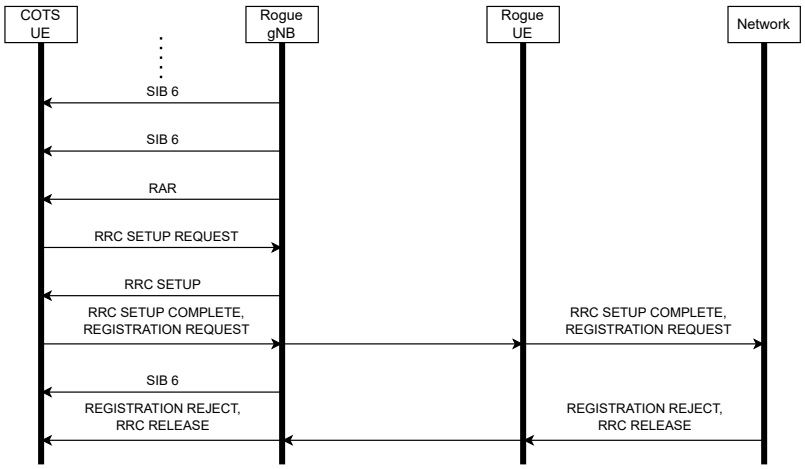

Figure 5.14: The observed signaling of an OTA test of the ETWS spoofing attack using a COTS UE with the partial OTA setup shown in Figure 5.7.

MITM Tool Implementation

To test this attack, we used the partial OTA setup shown in Figure 5.7. Other UE's were present in the testing environment, which were connected to their regular providers.

The MITM tool supports broadcasting the warning messages by adding the following configuration parameters to the gNB configuration:

```
cell_cfg:
  # ...
  # general configuration parameters
  # ...
  sib:
    si_window_length: 20
    si_sched_info:
      - si_period: 8
        sib_mapping: 6
```

Signaling

Figure 5.14 shows the signaling that caused the COTS UE in Figure 5.15 to show the warning message. The rogue gNB will broadcast the SIB 6 message according to the `si_period` parameter in the configuration above, even if no UEs are connected to it. When the UE attaches to the network, it enters the `RRC_CONNECTED` state and picks up the broadcasted SIB 6 messages as well, causing it to show a warning message as seen in Figure 5.15. The warning message was only displayed on the UE after it connected to the network. Other devices in the direct neighborhood of the gNB that were not connected to it did not show the warning.

Results

As can be seen in Figure 5.15, the MITM successfully broadcasts crafted ETWS warnings if configured to do so. As previously mentioned, the srsUE implementation used to create the rogue UE does not support paging for NR and, as such, will not forward these to the legitimate UE, thereby successfully implementing this attack.

We observed that the MITM can broadcast warning messages and that the COTS UE only handles these when it was forced to connect to the rogue gNB. We also observed that other devices in the room that were not connected to the rogue gNB but to the legitimate network

Figure 5.15: Receiving a crafted earthquake and tsunami warning on a COTS UE, originating from the rogue gNB.

would not display these warnings to their users. Our testing therefore shows that if the MITM wishes to spoof these messages it needs to be able to force UEs to connect to it instead of the legitimate network and also maintain the connection with those UEs if it wishes to spoof those messages at a later time after registration. As our tool is only capable of CP signaling, this means that spoofing of SIB 6 should happen during the registration of the UE, as the connection will eventually fail, releasing the UE.

5.7.3 Registration Reject: 5GS Services not Allowed

Bidding-down attacks are meant to reduce the security of a mobile network connection by forcing the UE to connect with an older, less protected (compared to 5G) generation of mobile network such as 4G or by downgrading parameters of the connection such as supported security algorithms to reduce the security of the chosen mobile network generation. Bidding-down can mean many different things in execution, such as downgrading the used encryption schemes or downgrading the type of identifier used in the communication with regards to privacy [Kar+23].

Bidding-down attacks that make the UE choose a lower generation network often set up a rogue base station of that generation as well, forcing the UE to connect to it, allowing the attacker to make use of the weaker security of those older generations [Kar+23].

There are many different ways this can be achieved, as described in [Kar+23]. One of which is through a NAS `REGISTRATION REJECT`. When the UE connects to the network with a NAS `REGISTRATION REQUEST`, it can be rejected by the network as described in Section 4.4.4. Depending on the cause of this rejection, the UE could be downgraded according to [Kar+23]. The definition of a downgrade, in that case, is attempting to "make the UE ignore all 5G networks after the reject and having it re-select a 4G cell" [Kar+23]. The specific `REGISTRATION REJECT` that was tested in this test was the `REGISTRATION REJECT` with cause `#7: 5GS Services not Allowed`.

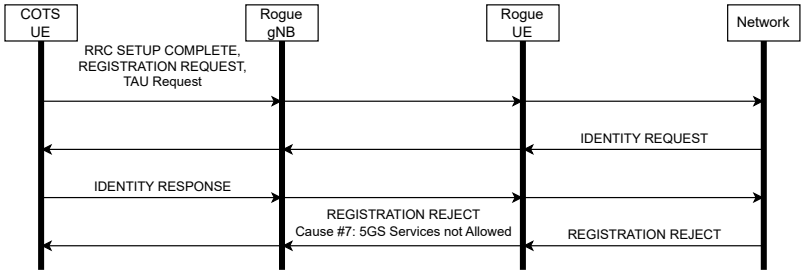

Figure 5.16: The observed signaling of an OTA test of the registration reject downgrade on a COTS UE using the `#7: 5GS Services not Allowed` cause. The setup shown in Figure 5.7 was used.

Normal Behavior

Under normal circumstances, the UE should be able to connect to the network without being rejected. There are situations, however, in which a rejection is possible, and as such, being rejected can also be considered normal behavior.

Specifically for the tested cause, the AMF responds to the UE with cause `#7: 5GS Services not Allowed` if the UE has been determined not to be allowed to operate on 5GS services due to subscription restrictions, for example. Upon reception of this message, the UE will no longer use the USIM for 5GS services until turning off to reset it [3GP24d].

Vulnerability

Due to the UE considering its USIM as invalid for 5GS services until switching off, some UEs will switch to 4G or below to provide connectivity to their users, achieving a downgrade. Other UEs will disable connectivity altogether, effectively creating a DoS until the device or USIM is restarted [Kar+23].

MITM Tool Implementation

To test this attack, we used the partial OTA setup shown in Figure 5.7.

The MITM tool supports sending this `REGISTRATION REJECT` with cause `#7: 5GS Services not Allowed` by simply executing a `CREATE` of the required message with the correct cause before the `AUTHENTICATION REQUEST` or `REGISTRATION REJECT` that would be the reply to the `REGISTRATION REQUEST`, or by executing an `UPDATE` on the NAS PDU contained within the first RRC downlink information transfer, also replacing the `AUTHENTICATION REQUEST` or `REGISTRATION REJECT` with a `REGISTRATION REJECT` with the `#7: 5GS Services not Allowed` cause.

Signaling

Figure 5.16 shows the observed test signaling representing the steps of the downgrade attempt using the `#7: 5GS Services not Allowed` cause. The rogue UE updates the contents of the RRC downlink information transfer as described earlier. The JSON configuration that delivered the signaling above specified precisely which message should be altered, which meant that the UE and network were able to execute some signaling before the `REGISTRATION REJECT` with cause `#7: 5GS Services not Allowed` replaced a `REGISTRATION REJECT`. Simply replacing the first downlink RRC message is also an option, which would be the `IDENTITY REQUEST` in this case.

Results

The downgrade attack using a `REGISTRATION REQUEST` with cause `#7: 5GS Services not Allowed` is fully implementable using our tool, as can be seen in the observed signaling of Figure 5.16. In the test we executed, the MITM is able to successfully replace the contents of a message with the required reject, causing the UE to disconnect from the network. These results are exactly as described in [Kar+23], which shows that no specialized implementation of a MITM is required to achieve these results.

As for the impact on our tested UE, we observed a DoS of the UE as was observed on some of the tested devices in the paper. The DoS on our UE was not as expected in the standard, which requires the UE disable the 5GS capabilities of the USIM, but not other network generations. Our UE was no longer able to connect to any network, 4G or below, until the USIM was restarted or the device was restarted. It is unclear if this is a DoS as was observed by [Kar+23], as they describe a DoS to be "a refusal of the service completely without switching to an older generation" [Kar+23] without specifying if this means a UE will also refuse to connect to other networks if the user commands it to do so, as was observed in our test.

5.7.4 Identity request: SUCI and IMEI

Besides downgrading a network generation, bidding-down attacks also take interest in downgrading the security of the generation that the UE decided to use. These kinds of downgrades usually relate to supported security algorithms or user identifiers for example. In 5G, one of those seemingly secure identifiers is the SUCI, providing an encrypted version of the SUPI to protect the user's permanent identifier. As seen in Section 4.4.12 the SUCI itself is also vulnerable to privacy attacks, although it can be argued that answering the question of whether a user is present in the network is less of a privacy concern than the user telling the attacker who they are outright, which the SUCI still manages to prevent (if it is encrypted).

In that regard, the SUPI of the UE is a unique identifier for the subscriber as described in Section 3.3.2 and therefore privacy sensitive information which, in 5G networks, should never be exposed due to the addition of the SUCI. The same goes for the IMEI, which, as described in Section 3.3.8, identifies the hardware used by the user to connect to the network. As the IMEI is also permanent, this identifier can also be used as a way for an attacker to identify a user and should therefore not be exposed [3GP24d].

Using the NAS identification procedure (see Section 4.4.7), the network is able to request different identities of a user such as a SUCI and the IMEI. The identities that can be requested by the network before security activation are different from the ones that can be requested after though: only the SUCI should be given in an `IDENTITY RESPONSE` before security activation. If an attacker were to request the IMEI of a subscriber before security activation, the subscriber would leak their identity. If the SUCI is not supported on the UE or USIM, responding to the `IDENTITY REQUEST` for the SUCI with a SUPI (i.e., not using encryption) would also leak the identity of the user, precisely like was the case in 4G with IMSI catchers (see Section 4.4.12) [Kar+23].

Normal Behavior

Under normal circumstances, the UE would only respond to an `IDENTITY REQUEST` for the SUCI before security activation and do so with a SUCI and not an unencrypted SUPI. If a different identifier such as the IMEI is requested before security activation, the UE should ignore this request [3GP24d].

Vulnerability

If an attacker requests the SUCI of the UE and it responds with its SUPI due to a lack of support for the SUCI, the subscriber's privacy is compromised and we effectively have the same

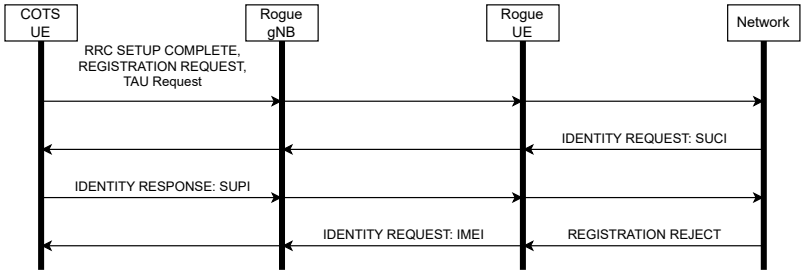

Figure 5.17: The observed signaling of an OTA test of requesting the SUCI and IMEI identities of a COTS UE before security activation using the setup shown in Figure 5.7.

identity vulnerabilities as in 4G and below, making IMSI-catching possible in 5G. If an attacker requests the IMEI of the UE before security activation, and the UE responds to this request, the UE compromises the privacy of the user as well.

MITM Tool Implementation

To test this attack, we used both the ZMQ setup shown in Figure 5.6 and the partial OTA setup shown in Figure 5.7.

The MITM can execute this attack by utilizing the `CREATE` functionality as well as the `UPDATE` functionality to either inject an `IDENTITY REQUEST` into the communication or replace the contents of a downlink message with an `IDENTITY REQUEST`. By using the `RETRIEVE` functionality, the response of the UE can be stored, achieving the goal of extracting the user's SUPI or IMEI if they respond to the `IDENTITY REQUEST` in an unsafe way.

Signaling

Figure 5.17 shows the signaling we observed when we requested the IMEI of a COTS UE as a replacement of the `REGISTRATION REJECT` sent by the network. As part of the signaling, the network already requested the SUCI of the UE, so it was not necessary to include this message as part of the test. As previously mentioned, the `CREATE` and `UPDATE` functionalities would be usable to add an `IDENTITY REQUEST` for the SUCI as we did for the IMEI.

Results

As can be seen in Figure 5.17, the UE responds to the `IDENTITY REQUEST` for the SUCI using its SUPI (actually: a null scheme SUCI), exposing the subscriber's permanent identifier. This happens because the commercial Belgian USIM that was used in the UE does not support the SUCI-functionality, even though this is a brand new commercially available USIM that was purchased specifically for these tests in 2025. This result was also observed in [Kar+23]. We observed this result in both the COTS UE and srsUE.

The request for the IMEI of the COTS UE went unanswered, meaning the UE did not leak any sensitive information in that regard. SrsUE did respond to the IMEI `IDENTITY REQUEST` before security activation, however.

These observed results highlight once again that our tool would be usable for these kinds of tests through the general CRUD functionalities.

5.7.5 Registration Reject: N1 Mode not Allowed

As described in Section 5.7.3, bidding down attacks often try to force a UE to a lower generation network to reduce the security of the connection. As seen in that section, it is possible to do this using a reject cause in the `REGISTRATION REJECT` message. `REGISTRATION`

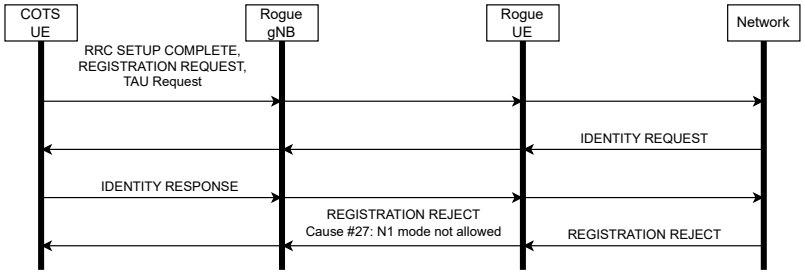

Figure 5.18: The observed signaling of an OTA test of the registration reject downgrade on a COTS UE using the `#27: N1 mode not allowed` cause. The setup that was used is shown in Figure 5.7.

`REJECT`s have a plethora of causes to choose from, and one of those is cause `#27: N1 mode not allowed`.

N1 mode is defined as the mode of the UE that allows it to connect to the 5GC via a 5G AN [3GP24d]. In other words: 5G SA [Ryu25b]. When the UE receives this cause, it should disable its capabilities for 5G SA completely [Kar+23], opening it up to bidding down attacks as it attempts to attach to a lower generation network.

Normal Behavior

Under normal circumstances, the UE should be able to connect to the network without being rejected. There are situations, however, in which a rejection is possible, and as such, being rejected can also be considered normal behavior.

The AMF responds to the UE with cause `#27: N1 mode not allowed` if the subscription data on the USIM of the UE does not permit it to access 5G services, in both a normal and a roaming scenario. Upon reception of this message, the UE will disable N1 mode [3GP24d].

Vulnerability

If the UE disables its capabilities for 5G SA, it might attempt to register itself to a 4G network or older, achieving the downgrade.

MITM Tool Implementation

To test this attack, we used the partial OTA setup shown in Figure 5.7.

The MITM tool supports sending this `REGISTRATION REJECT` with cause `#27: N1 mode not allowed` by simply executing a `CREATE` of the required message before the `AUTHENTICATION REQUEST` or `REGISTRATION REJECT` that would be the reply to the `REGISTRATION REQUEST`, or by executing an UPDATE on the NAS PDU contained within the first RRC downlink information transfer, replacing the `AUTHENTICATION REQUEST` or `REGISTRATION REJECT` with a `REGISTRATION REJECT` with cause `#27: N1 mode not allowed`.

Signaling

Figure 5.18 shows the signaling representing the steps of the downgrade attempt using the `#27: N1 mode not allowed` cause. The rogue UE updates the contents of the RRC downlink information transfer as described earlier. The signaling between the legitimate UE and network was unaltered for the first set of messages due to the JSON configuration that was used to achieve the active MITM setup specifying precisely which message (e.g., the `REGISTRATION REJECT`) should be replaced. Simply replacing the first downlink message was also an option (the `IDENTITY REQUEST` in this case).

Results

The downgrade attack using a `REGISTRATION REJECT` with cause `#27: N1 mode not allowed` is fully implementable using our tool, as can be seen from the observed signaling of Figure 5.18. In the test we executed, the MITM is able to successfully replace the contents of a message with the required reject, causing the UE to disconnect from the network. The observed signaling is precisely that of [Kar+23], which shows that no specialized implementation of a MITM is necessary to achieve these results.

The tested UE did ignore the semantics of the message we sent, however. After receiving the rejection, the UE disconnected from the network, but was still able to reconnect to it in subsequent attempts without any resets. This should not be possible if the UE follows the procedure of disabling SA as specified in [3GP24d]. The downgrade was therefore unsuccessful. This is not as described in [Kar+23], in which this test was able to downgrade all tested UE's.

5.7.6 RRC Release with Redirection

As described in Section 4.3.10, the RRC release procedure is meant to release the RRC connection with a UE. This procedure gets triggered by the network through an `RRCRelease` message. Besides simply releasing the UE, the release message is able to tell the UE to move to a different frequency or a different network generation [Kar+23].

Since an `RRCRelease` can be sent at any moment, sending one before security activation is possible. This allows us to attempt to downgrade the user by sending such a release with redirection to a lower generation network. In 5G, however, the UE should ignore the redirection field before security activation. Besides that, the UE should only accept a redirection to 4G [Kar+23]. If a UE does accept the redirection, however, we successfully achieve a downgrade.

Normal Behavior

Under normal circumstances, the UE should be able to connect to the network without being released. There are situations, however, in which a release is possible, and as such, being released can also be considered normal behavior.

If a UE receives an `RRCRelease` with redirection, it should follow this redirection if the message was authenticated. If it was not, the UE should ignore the message. Furthermore, if the message attempts to redirect to any generation older than 4G, the redirection should be ignored as well.

Vulnerability

If a UE does not ignore the redirection of an unauthenticated `RRCRelease` with redirection to 4G, the UE will connect to the 4G network, achieving the downgrade.

MITM Tool Implementation

To test this attack, we used the partial OTA setup shown in Figure 5.7.

The MITM tool supports sending the `RRCRelease` by simply executing a `CREATE` of the required message at any point before security activation. Alternatively, an `UPDATE` can be used to replace the contents of downlink RRC message to become an `RRCRelease` with redirection. The message must be sent before security activation, as the MAC of the PDCP layer cannot be empty after security activation.

Signaling

Figure 5.19 shows the signaling representing the steps of the downgrade attempt using the redirection feature of the RRC release procedure. The rogue UE inserts the release before the

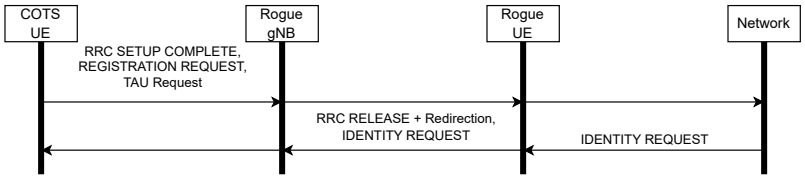

Figure 5.19: The observed signaling of an OTA test with a COTS UE, releasing it with a redirection to a 4G network before authentication. The setup that was used is shown in Figure 5.7.

first downlink information transfer by the network, ensuring that the release gets parsed before the `IDENTITY REQUEST`. Using the `DELETE` or `UPDATE` functionalities, the `IDENTITY REQUEST` could have been removed entirely, but leaving it in shows that the UE did at least partially parse the release if it does not answer the `IDENTITY REQUEST`.

Results

The downgrade attack using the `RRCRelease` message with a redirection is fully implementable using our MITM tool, as can be seen in Figure 5.19. In the test we executed, the MITM successfully injects an `RRCRelease` that is parsed by the UE, as it no longer responds to the `IDENTITY REQUEST` sent by the network. The observed signaling is exactly as expected in [Kar+23], showing that no specialized implementation of a MITM is required to achieve these results.

The UE did not redirect itself to a lower generation network, however. It remained without a connection until we manually made it connect to a different network. Both 5G and 4G networks were valid choices, and as such, no positive result was achieved. This result is as described in [Kar+23], which also reported not finding any issues with regards to UE's obeying unauthenticated RRC releases with redirection.

5.7.7 NReplay

As described in Section 4.4.6, the security mode control procedure is meant to activate security between the UE and network by having the network send the chosen algorithms and key set to the UE. Upon receiving this message, the UE takes the security context into use as was described in the `SECURITY MODE COMMAND`. [FSP24] discovered that the handling of this message contains a vulnerability, which, if combined with unclear specifications, opens the UE up to a key reinstallation and DoS attack.

Specifically, if the UE has to retransmit the `SECURITY MODE COMPLETE` multiple times, the UE will reset the NAS `COUNT`, without changing the keystream. Due to the multiple failures to transmit the message, the UE will eventually re-send its `REGISTRATION REQUEST` ciphered with the keystream provided by the security context. Since the MITM has the original unencrypted `REGISTRATION REQUEST`, and now its ciphered equivalent, the MITM is able to extract the keystream that was used. Using this keystream, the MITM is able to encrypt a `DEREGISTRATION REQUEST`, which the AMF might handle even though it fails the integrity verification (due to unclear specifications). If the AMF eventually accepts the `DEREGISTRATION REQUEST`, the UE will be DoS'ed [FSP24]. A more detailed description of this attack can be found in Section 4.4.13 .

This attack requires a MITM setup that is able to inject messages and stop forwarding them dynamically, which our setup provides. The attack also requires the MITM to be able to perform an XOR operation on an incoming message and a cached keystream, which our setup does not provide. This attack serves as an example that, in some cases, a setup requires specialized features to achieve the exploit.

Normal Behavior

Under normal circumstances, the UE is able to immediately deliver its `SECURITY MODE COMPLETE` without retransmissions that eventually cause it to reuse its keystream. Furthermore, normally the AMF would not receive a message that fails the integrity check and it would be clear what it should do with the message: either handle it or not.

Vulnerability

If the UE fails to transmit the `SECURITY MODE COMPLETE`, it resets the NAS `COUNT`, but not the keystream, causing keystream reuse. Also, when the AMF receives a message that fails the integrity check, the specification is unclear about how it should handle this message. A possible approach is to authenticate the user, and if this succeeds, handle the message despite the integrity failure. These two vulnerabilities combined allow a MITM to DoS a UE by obtaining its keystream using a combination of blocking, forwarding, and injecting messages [FSP24].

MITM Tool Implementation

As previously mentioned, the MITM tool does not provide all the features required to successfully complete this attack. Concretely, the attack requires the following:

1. Forwarding, injecting, and halting traffic in the connection. Our setup does provide this through the CRUD operations.
2. Applying an active operation, such as XOR, on messages passing through. Our setup does not support this.

As such, we will only describe how our setup could be used to emulate the signaling as described in [FSP24]:

1. `RETRIEVE` on the `REGISTRATION REQUEST`: stores the `REGISTRATION REQUEST` for later use.
2. `DELETE` on the `SECURITY MODE COMPLETE`, 4x. Forces the UE to retransmit the `SECURITY MODE COMPLETE` four times.
3. `UPDATE` on the `REGISTRATION REQUEST`, replacing it with a `DEREGISTRATION REQUEST`.

The steps above would achieve the signaling as it was shown in Figure 4.29, but since we lack the functionalities to implement the rest of the attack, we have no conclusive way to show this.

5.7.8 Overview

As an overview of the tested attacks, Table 5.1 shows how well the MITM tool could implement each attack and the results of the tested attacks.

Attack	MITM Support	Test Result	Note
SUCI catcher	Full	Positive	
ETWS spoofing	Full	Positive	
Registration reject: 5GS	Full	Positive	
Identity request: SUCI	Full	Positive	
Identity request: IMEI	Full	Positive	Tested on srsUE.
		Negative	Tested on COTS UE.
Registration reject: N1	Full	Negative	
RRC release w/ redirect	Full	Negative	
NReplay	Signaling only	No result	XOR operation not supported.

Table 5.1: An overview of all of the tested known attacks, how well they are implementable using our tool, and the test result, including a note on how the test was done or why it could not be done, if relevant.

Chapter 6

Conclusion

6.1 Research Questions

In this thesis, we attempted to answer four research questions, introduced in Section 1. Those questions were: "can we use existing open-source solutions to implement a configurable 5G MITM tool?", "would such a tool be usable to implement known vulnerabilities?", "are there vulnerabilities that are still a problem after their discovery?", and "how deployable is such a MITM setup in the real world?".

The delivered work showed that it is possible to use freely available open-source software to implement a MITM setup that can be used for testing and research purposes. We were able to use the srsRAN and srsUE software to provide a rogue base station and UE that are able to cooperate together to forward traffic from the victim UE and gNB back and forth for all CP signaling. Besides simple forwarding, we also demonstrated that such a MITM is also able to take a more active role in the network, by providing all four CRUD operations on the data passing through the MITM. Through configuration files, we allowed the user to customize the operations executed by the MITM, achieving our goal of creating a configurable 5G MITM research tool, answering our first research question.

To evaluate the tool we created, we used it to execute different attacks that were already discovered for 5G that require a MITM setup (or at least a rogue base station). We evaluated the usability of our tool for these different attacks, showing that most of the attacks we picked are fully executable using our tool. One attack, namely NReplay [FSP24], showed that sometimes support for specific operations is required, and generic tools like ours aren't enough. Nevertheless, these tests showed that our tool is can be used to implement existing 5G attacks, answering our second research question.

By testing the implementability of existing vulnerabilities, we were also able to evaluate whether or not they are still exploitable some time after their discovery. The tests we did showed some positive cases (attacks that succeeded) and some negative cases (attacks that did not succeed). We cannot draw any conclusions about the viability of an attack in case the test did not succeed: the attacks we tested often focus on (incorrect) interpretations of a standard, which might differ across different devices. Our contribution in case a test failed, therefore, is that the device we tested was not vulnerable to the attack, but other devices might still be. Combined with the positive results, this answers our third research question that at least some attacks, for sure, are still a vulnerability after their discovery.

Finally, we experienced difficulties actually deploying our setup in a real-world scenario. The setup we created only has support for NR on the gNB side (the UE does support NR, NSA, and LTE), which requires the victim UE to use NR in order to connect to the MITM. Deploying a 5G SA setup and connecting it with an external commercial network, however, proved difficult

as those deployments were not present in our immediate area. Since 5G NSA is available, we attempted to provide a translation between the 5G signaling from the victim UE and the 4G signaling the rogue UE should send. The messages we sent to the network were rejected, and we were unable to solve this problem due to timing constraints. These experienced difficulties still answer the final research question, namely that deploying a MITM setup in a real-world scenario could be difficult depending on the choices that were made during implementation (more below).

6.2 Future Work

Since the full deployment of the setup was not achieved within the provided time window, we point to the support of 4G / 5G NSA as an opportune path for future work. Implementing support for NSA would mean moving to a different gNB implementation than srsRAN, such as OpenAirInterface or srsLTE (using the provided eNB implementation). Doing so would enable the MITM to be connected to a commercial network, demonstrating its real-world capabilities.

Expanding the set of operations supported by the MITM is also a relevant path for future work. Currently, the provided CRUD operations enable the MITM to execute attacks that work by injecting, altering, removing, or storing messages, but support for features such as operations like XOR'ing a message with a given byte sequence would enhance the capabilities of the MITM such that it can support more attacks, like NReplay [FSP24], for example.

Finally, expanding the usability of our tool could also be interesting. As it stands, the interaction with the tool is done through JSON configuration files, and the final output of the messaging has to be figured out by looking at the Wireshark traces of both rogue components. If, for example, a downlink message type was configured to be deleted, that message will still be shown in the Wireshark trace of the legitimate gNB, the rogue UE, but no longer on the rogue gNB or legitimate UE. Piecing this information together can be confusing and hard to use, and we suggest an approach such as a unified overview of the actual signaling based on the configuration files the user created.

6.3 Reflection

Looking back on the work that was delivered, I feel like I was able to achieve what I set out to do when I decided to go for this thesis. From the start, I knew this would be a great undertaking, as is the case with any thesis, but I felt like mine would be a bit more challenging due to the lack of background knowledge I had on the subject. Throughout my education, I had the opportunity to look at networks in general, how they worked, their protocols, etc., but wireless networks were never really focused on. Especially mobile networks such as 5G were an untouched topic for me, which is what attracted me to this topic in the first place. I felt like studying the security of a type of network I was unfamiliar with, but use every day would be very rewarding. Those feelings, it turned out, were valid. I learned so much studying how these networks are built, how their protocols cooperate, what vulnerabilities they have, and so much more. In the end, it was a lot of information to take in, as can be seen in the background sections provided in this thesis. This meant that I also had to adapt to learning more as I went, instead of first deeply studying a topic and then applying it. This is certainly experience I will use in my later life as well.

Learning as I went also had its downsides, however, manifesting in work that had to be redone due to later insights or incorrect interpretations of certain things that cost me a great amount of time. This lost time eventually led to an implementation that, even though it does answer the research questions I identified, could have been more comprehensive. As described in the previous section about future work, I had clear plans on what I still wanted to achieve with my implementation, but due to the time-consuming learning-as-I-went nature of this thesis, I was

unable to achieve. Regardless, I am proud of the implementation that I made, and especially of its originality. Like previously mentioned, we were unable to find any existing implementation of what we made, which makes it quite special for me to have been able to achieve this. I'm also very proud of the thesis I wrote. I feel like I provide a good and thorough overview of all the research that I did and implemented, and I think that the effort I put into it really shows.

There are things that I would have done differently in hindsight, though. Looking back on the issues that I faced, especially with regards to deployment, I think choosing OpenAirInterface's RAN instead of srsRAN would have been the better choice. OpenAirInterface provides both 5G NSA and SA support, which means I would have been able to support this with my tool as well. I also would have spent less time fixing problems that were not essential and cost a great amount of time to fix. These two things learned me the valuable lesson that sometimes I should look at the bigger picture and pivot quicker to an alternative strategy instead of doubling down on a specific approach.

All in all, I am satisfied with this educational journey, and I feel like I learned a lot about 5G networks, security in general, but also about myself. These lessons will guide me towards performing at my best later in life, and I am grateful for the opportunity that was given to me with this work.

Acronyms

K_{AMF} AMF Security Key. 56, 57, 60

K_{AUSF} Home Network's Security Anchor Key. 56, 60

K_{NASenc} NAS Encryption Key. 56, 60, 66

K_{NASint} NAS Integrity Key. 56, 60, 66

K_{RRCenc} RRC Encryption Key. 38, 43, 44, 60

K_{RRCint} RRC Integrity Key. 38, 43, 44, 60

K_{SEAF} Visited Network's Security Anchor Key. 29, 55, 56, 60

K_{UPenc} UP Encryption Key. 38, 60

K_{UPint} UP Integrity Key. 38, 60

K_{gNB} gNB Security Key. 43, 60

4GS 4G System. 28

5G HE AV 5G Home Environment Authentication Vector. 29

5G-AKA 5G Authentication and Key Agreement. 27, 55–57

5GC 5G Core. 5, 17–24, 27–29, 78, 80, 96

5GMM 5GS Mobility Management. 49–54, 62, 63, 65

5GS 5G System. 7, 14, 17, 18, 21, 25, 28, 30, 34, 50, 53–55, 63, 93, 94, 100

5GSA 5G Security Architecture. 29

5GSM 5GS Session Management. 49, 51–53, 62, 64, 65

AKA Authentication and Key Agreement. 29, 38, 51, 55, 57, 59, 60, 65, 68

AMF Authentication Management Field. 59

AMF Access and Mobility Management Function. 21–29, 31, 40, 47, 49, 51–57, 60–66, 69, 71, 77, 80, 88, 93, 96, 98, 99

AN Access Network. 17, 23, 25, 96

ARPF Authentication credential Repository and Processing Function. 29, 56, 60

AS Access Stratum. 34, 41, 43, 54

AUSF Authentication Server Function. 21, 24, 27–29, 31, 56, 57

AV Authentication Vector. 56, 59, 68, 69, 88–90

BCCH Broadcast Control Channel. 34, 36

BCH Broadcast Channel. 36

C-RNTI Cell Radio Network Temporary Identifier. 32, 41, 43

CCCH Common Control Channel. 36, 40

CD Check Digit. 33

CM Connection Management. 23

CMAS Commercial Mobile Alert Service. 40, 41, 47

COTS Commercial-Off-The-Shelf. 79, 80, 83, 89–93, 95, 96, 98, 100

CP Control-plane. 15, 21, 25, 34, 35, 37–40, 49, 76, 78, 83, 92, 101

CRUD Create, Retrieve, Update, Delete. 6, 7, 81, 83, 95, 99, 101, 102

CSS Cell Site Simulator. 67

CU Central Unit. 73, 75

CU-CP Central Unit - Control Plane. 73, 76

CU-UP Central Unit - User Plane. 73, 76

DCCH Dedicated Control Channel. 36, 40

DL-SCH Downlink Shared Channel. 36

DN Data Network. 25, 26, 52, 63, 64

DNN Data Network Name. 26, 64

DNS Domain Name Service. 27

DoS Denial of Service. 71, 93, 94, 98, 99

DRB Data Radio Bearer. 40, 41, 43, 44, 46, 83

DTCH Dedicated Traffic Channel. 36

DU Distributed Unit. 73, 75–77

DU-High Distributed Unit - High. 73, 76

DU-Low Distributed Unit - Low. 73

EAP Extensible Authentication Protocol. 55

eMBB Enhanced Mobile Broadband. 12, 14, 16, 19

eNB Evolved Node B. 19, 20, 39, 77, 78, 102

EPC Evolved Packet Core. 18–21, 77, 78

EPS Evolved Packet System. 59, 63

ETWS Earthquake and Tsunami Warning System. 7, 40, 41, 47, 90, 91, 100

EU European Union. 14, 15

F1AP F1 Application Protocol. 76

FAPI Functional Application Platform Interface. 73

gNB 5G Node B. 5, 17–20, 22–24, 32, 37, 39, 40, 42, 43, 46, 49, 57, 60, 63, 72, 73, 77–84, 86, 87, 89–92, 101, 102

GTP-U GPRS Tunneling Protocol - User Plane. 76

GUAMI Globally Unique AMF Identifier. 31

GUTI Globally Unique Temporary Identifier. 24, 31, 32, 52, 54, 57, 62, 63

GW Gateway. 78

HN Home Network. 29–31, 55–57

HNI Home Network Identifier. 30, 31

IE Information Element. 43, 51, 53, 54, 60–62, 64

IMEI International Mobile station Equipment Identity. 7, 33, 62, 94, 95, 100

IMEISV International Mobile station Equipment Identity and Software Version number. 33, 61

IMSI International Mobile Subscriber Identity. 28, 30, 31, 66–68, 90, 94, 95

IoT Internet of Things. 4, 16, 17

LTE Long Term Evolution. 15, 18–21, 46, 47, 73, 78, 101

MAC Message Authentication Code. 38, 59, 82, 86, 90, 97

MAC Medium Access Control. 34, 36, 40, 73, 76, 78, 83

MCC Mobile Country Code. 30, 31, 54, 56

ME Mobile Equipment. 33

MIB Master Information Block. 36, 41, 42, 48, 79

MITM Man-In-The-Middle. 3–7, 12, 13, 28, 46, 48, 49, 67, 69, 71, 72, 77–102

MM Mobility Management. 22, 23, 50

mMTC massive Machine-Type Communications. 12, 14, 16

mmWave millimeter Wave. 15, 16

MNC Mobile Network Code. 30, 31, 54, 56

MS Mobile Station. 17, 28, 33

MSIN Mobile Subscriber Identification Number. 30, 31

MTU Maximum Transmission Unit. 27

NAS Non-Access Stratum. 22, 23, 25, 34, 38, 40, 43, 44, 46, 49–54, 56, 60, 61, 63, 65, 66, 69, 77, 78, 81–87, 92–94, 98, 99

NEA New Radio Encryption Algorithm. 44

NF Network Function. 21, 22, 24, 25, 27, 28

NG Next Generation. 18

ng-eNB Next Generation Evolved NodeB. 18

NGAP Next Generation Application Protocol. 24, 76, 77

ngKSI 5G Key Set Identifier. 54, 59–61, 65

NIA New Radio Integrity Algorithm. 44

NR New Radio. 15, 16, 18–20, 45, 47, 73, 78, 80, 90, 91, 101

NRF Network Repository Function. 21, 24, 26, 28

NSA Non-Standalone. 8, 18–20, 54, 77, 78, 80, 101–103

O-RAN Open Radio Access Network. 73

OTA Over-The-Air. 79, 80, 88, 89, 91, 93, 95–98

OTP One-time Pad. 69

PCCH Paging Control Channel. 34, 36

PCF Policy Control Function. 24, 26

PCH Paging Channel. 36

PDCP Packet Data Convergence Protocol. 6, 37–40, 44, 45, 76–78, 81–86, 97

PDU Packet Data Unit. 15, 23, 25–27, 37, 38, 49, 50, 52, 53, 63–65, 76, 78, 84, 85, 87, 93

PEI Permanent Equipment Identifier. 33, 54

PHY Physical. 40, 73, 78, 83

PLMN Public Land Mobile Network. 23, 30, 43, 54, 78

PTI Procedure Transaction Identity. 64, 65

PWS Public Warning System. 40, 42, 47, 48, 90

QFI Quality of Service Flow ID. 27

QoS Quality of Service. 18, 25–28, 53, 64, 76

RA Registration Area. 24

RACH Random Access Channel. 36, 76

RAN Radio Access Network. 16, 18, 20, 21, 47, 73, 76, 77, 103

RAT Radio Access Technology. 18, 19, 46, 47, 80

RB Radio Bearer. 38, 40, 43–46

REST Representational State Transfer. 21, 28

RI Routing Indicator. 31

RLC Radio Link Control. 40, 44, 45, 73, 76, 78, 83

RNTI Radio Network Temporary Identifier. 32, 76

RRC Radio Resource Control. 7, 23, 32, 36, 38–47, 49, 51, 60, 63, 73, 76–79, 81–87, 90, 93, 96–98, 100

S-TMSI Shortened Temporary Mobile Subscriber Identity. 32, 43

SA Standalone. 8, 19, 20, 77, 78, 80, 96, 97, 101, 103

SBA Service-based Architecture. 17, 21, 28

SD Spare Digit. 33

SDAP Service Data Adaption Protocol. 38, 76

SDR Software Defined Radio. 73, 77, 80

SEAF Security Anchor Function. 29

SI System Information. 34, 36, 40–42, 78, 79

SIB System Information Block. 40–43, 47, 48, 79, 83, 90–92

SIDF Subscription Identifier De-concealing Function. 29, 56

SM Session Management. 25

SMF Session Management Function. 21, 23–28, 51–53, 63, 64

SMS Short Message Service. 24, 27, 49, 53

SN Serving Network. 28, 55–57

SNR Serial Number. 33

SRB Signaling Radio Bearer. 40–44, 46, 83

SUCI Subscription Concealed Identifier. 7, 27, 29–31, 33, 51, 52, 54, 56, 57, 59, 62, 65–69, 87–90, 94, 95, 100

SUPI Subscription Permanent Identifier. 27–31, 51, 52, 56, 57, 67, 68, 88, 90, 94, 95

SVN Software Version Number. 33

TA Tracking Area. 24, 43

TAC Type Allocation Code. 33

TAI Tracking Area Identity. 24, 54, 63

TMSI Temporary Mobile Subscriber Identity. 31, 32

UDM Unified Data Management Function. 21, 24, 26–29, 31, 56, 57

UDR Unified Data Repository. 21, 28

UE User Equipment. 5–7, 17, 18, 21–34, 36–69, 71, 72, 76–102

UL-SCH Uplink Shared Channel. 36

UP User-plane. 15, 21, 23–26, 34, 35, 37, 38, 40, 60, 64, 65, 76, 78, 83

UPF User Plane Function. 21, 24–27, 53, 76

URLLC Ultra Reliable Low Latency Communication. 12, 14–16, 73

USIM Universal Subscriber Identity Module. 17, 28, 30, 60, 68, 77, 80, 93–96

V2V Vehicle-to-Vehicle. 15

XR Extended Reality. 17

ZMQ ZeroMQ. 80, 88, 89, 95

Bibliography

[3GP19] 3GPP. *3GPP TR 21.915 version 15.0.0 Release 15*. en. ETSI, Oct. 2019. URL: https://www.etsi.org/deliver/etsi_tr/121900_121999/121915/15.00.00_60/tr_121915v150000p.pdf (visited on 06/12/2025).

[3GP24a] 3GPP. *3GPP TS 23.003 version 18.7.0 Release 18*. en. ETSI, Sept. 2024. URL: https://www.etsi.org/deliver/etsi_ts/123000_123099/123003/18.07.00_60/ts_123003v180700p.pdf (visited on 06/12/2025).

[3GP24b] 3GPP. *3GPP TS 23.501 version 18.7.0 Release 18*. en. ETSI, Oct. 2024. URL: https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/18.07.00_60/ts_123501v180700p.pdf (visited on 06/12/2025).

[3GP24c] 3GPP. *3GPP TS 24.007 version 18.3.0 Release 18*. en. ETSI, Oct. 2024. URL: https://www.etsi.org/deliver/etsi_ts/124000_124099/124007/18.03.00_60/ts_124007v180300p.pdf (visited on 06/12/2025).

[3GP24d] 3GPP. *3GPP TS 24.501 version 18.8.0 Release 18*. en. ETSI, Oct. 2024. URL: https://www.etsi.org/deliver/etsi_ts/124500_124599/124501/18.08.00_60/ts_124501v180800p.pdf (visited on 06/12/2025).

[3GP24e] 3GPP. *3GPP TS 33.102 version 18.0.0 Release 18*. en. ETSI, Apr. 2024. URL: https://www.etsi.org/deliver/etsi_ts/133100_133199/133102/18.00.00_60/ts_133102v180000p.pdf (visited on 06/12/2025).

[3GP24f] 3GPP. *3GPP TS 33.501 version 18.7.0 Release 18*. en. ETSI, Oct. 2024. URL: https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/18.07.00_60/ts_133501v180700p.pdf (visited on 06/12/2025).

[3GP24g] 3GPP. *3GPP TS 38.331 version 18.3.0 Release 18*. en. ETSI, Oct. 2024. URL: https://www.etsi.org/deliver/etsi_ts/138300_138399/138331/18.03.00_60/ts_138331v180300p.pdf (visited on 06/12/2025).

[3GP25a] 3GPP. *3GPP TS 23.501 version 18.9.0 Release 18*. en. ETSI, Apr. 2025. URL: https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/18.09.00_60/ts_123501v180900p.pdf (visited on 06/12/2025).

[3GP25b] 3GPP. *3GPP TS 38.300 version 18.4.0 Release 18*. en. ETSI, Jan. 2025. URL: https://www.etsi.org/deliver/etsi_ts/138300_138399/138300/18.04.00_60/ts_138300v180400p.pdf (visited on 06/12/2025).

[3GP25c] 3GPP. *3GPP TS 38.306 version 18.4.0 Release 18*. en. ETSI, Jan. 2025. URL: https://www.etsi.org/deliver/etsi_ts/138300_138399/138306/18.04.00_60/ts_138306v180400p.pdf (visited on 06/12/2025).

[3GP25d] 3GPP. *3GPP TS 38.323 version 18.4.0 Release 18*. en. ETSI, Jan. 2025. URL: https://www.etsi.org/deliver/etsi_ts/138300_138399/138323/18.04.00_60/ts_138323v180400p.pdf (visited on 06/12/2025).

[3GP25e] 3GPP. *3GPP TS 38.401 version 18.4.0 Release 18*. en. ETSI, Jan. 2025. URL: https://www.etsi.org/deliver/etsi_ts/138400_138499/138401/18.04.00_60/ts_138401v180400p.pdf (visited on 06/12/2025).

[AB24] Francisco Amaya and Kalvin Bahia. *The State of 5G 2024: Introducing the GSMA Intelligence 5G Connectivity Index*. en. GSMA Intelligence, Feb. 2024. URL: https://www.gsmaintelligence.com/research/research-file-download?id=79791087&file=210224-The-State-of-5G-2024.pdf (visited on 06/12/2025).

[Ame18] 5G Americas. *Public Warning System in the Americas*. en. 5G Americas, July 2018. URL: https://www.5gamericas.org/wp-content/uploads/2019/07/Public_Warning_Systems_Americas_WhitePaper___Final_for_distribution.pdf (visited on 06/12/2025).

[Ame23] 5G Americas. *Evolving Devices for 5G Adoption*. en. 5G Americas, Oct. 2023. URL: https://www.5gamericas.org/wp-content/uploads/2023/10/Evolving-Devices-for-5G-Adoption-FINAL-Id.pdf (visited on 06/12/2025).

[Att+22] Giulia Attanasio et al. "In-depth study of RNTI management in mobile networks: Allocation strategies and implications on data trace analysis". In: *Computer Networks* 219 (2022), p. 109428. URL: https://www.sciencedirect.com/science/article/pii/S1389128622004625 (visited on 06/12/2025).

[Bat17] Adam Bates. *Stingray: A New Frontier in Police Surveillance*. en. Cato Institute, Jan. 2017. URL: https://www.cato.org/sites/cato.org/files/pubs/pdf/pa-809-revised.pdf (visited on 06/12/2025).

[BIP24] BIPT. *Phasing out of 3G networks*. en. Sept. 2024. URL: https://www.bipt.be/consumers/phasing-out-of-3g-networks (visited on 06/12/2025).

[BP22] Evangelos Bitsikas and Christina Pöpper. "You have been warned: Abusing 5G's Warning and Emergency Systems". In: *Proceedings of the 38th Annual Computer Security Applications Conference*. ACSAC. ACM, Dec. 2022, pp. 561–575. DOI: 10.1145/3564625.3568000. URL: http://dx.doi.org/10.1145/3564625.3568000 (visited on 06/12/2025).

[CDL16] Mauro Conti, Nicola Dragoni, and Viktor Lesyk. "A Survey of Man In The Middle Attacks". In: *IEEE Communications Surveys & Tutorials* 18.3 (2016), pp. 2027–2051. DOI: 10.1109/COMST.2016.2548426. (Visited on 06/12/2025).

[Chl+21] Merlin Chlosta et al. "5G SUCI-catchers: still catching them all?" In: *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. WiSec '21. Abu Dhabi, United Arab Emirates: Association for Computing Machinery, 2021, pp. 359–364. ISBN: 9781450383493. DOI: 10.1145/3448300.3467826. URL: https://doi.org/10.1145/3448300.3467826 (visited on 06/12/2025).

[Cho24] Ting-Yuan Chou. *Introduction to 5G Quality of Service (QoS)*. en. June 2024. URL: https://free5gc.org/blog/20240628/20240628/ (visited on 06/12/2025).

[Com25] European Commission. en. June 2025. URL: https://digital-strategy.ec.europa.eu/en/policies/5g-digital-decade (visited on 06/12/2025).

[Cox21] Christopher Cox. *An Introduction to 5G: The New Radio, 5G Network and Beyond*. John Wiley and Sons, Ltd, Dec. 2021. ISBN: 9781119602682. URL: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119602682 (visited on 06/12/2025).

[Dab+14] Adrian Dabrowski et al. "IMSI-catch me if you can: IMSI-catcher-catchers". In: *Proceedings of the 30th Annual Computer Security Applications Conference*. ACSAC '14. New Orleans, Louisiana, USA: Association for Computing Machinery, 2014, pp. 246–255. ISBN: 9781450330053. DOI: 10.1145/2664243.2664272. URL: https://doi.org/10.1145/2664243.2664272 (visited on 06/12/2025).

[DPS23] E. Dahlman, S. Parkvall, and J. Skold. *5G/5G-Advanced: The New Generation Wireless Access Technology*. Elsevier Science, Oct. 2023. ISBN: 9780443131738. URL: https://books.google.be/books?id=QwuvzwEACAAJ (visited on 06/12/2025).

[Emb22] Emblasoft. *Exploring the 3GPP AMF – Access & Mobility Management Function*. en. Sept. 2022. URL: https://emblasoft.com/blog/exploring-the-3gpp-amf-access-mobility-management-function (visited on 06/12/2025).

[Eve24] The Parliament Events. *A lack of action on 5G rollout risks Europe being left behind*. en. Jan. 2024. URL: https://www.theparliamentmagazine.eu/partner/article/a-lack-of-action-on-5g-rollout-risks-europe-being-left-behind (visited on 06/12/2025).

[FSP24] Wei Fan, Bingnan Shi, and Cheng Peng. "NReplay: 5G Key Reinstallation Attack Based on NAS Layer Vulnerabilities". In: *MILCOM 2024 - 2024 IEEE Military Communications Conference (MILCOM)*. 2024, pp. 1088–1093. DOI: 10.1109/MILCOM61039.2024.10773741. (Visited on 06/12/2025).

[GL22] GSMA and Coleago Consulting Ltd. *Vision 2030: Low-Band Spectrum for 5G*. en. GSMA, June 2022. URL: https://www.gsma.com/connectivity-for-good/spectrum/wp-content/uploads/2022/07/5G-Low-Band-Spectrum-1.pdf (visited on 06/12/2025).

[GSM19] GSMA. *Operator Requirements for 5G Core Connectivity Options*. en. GSMA, May 2019. URL: https://www.gsma.com/solutions-and-impact/technologies/networks/wp-content/uploads/2019/05/20190515-GSMA-Operator-Requirements-for-5G-Core-Connectivity-Options.pdf (visited on 06/12/2025).

[GSM20] GSMA. *5G Implementation Guidelines: NSA Option 3*. en. GSMA, Feb. 2020. URL: https://www.gsma.com/solutions-and-impact/technologies/networks/wp-content/uploads/2019/03/5G-Implementation-Guidelines-NSA-Option-3-v2.1.pdf (visited on 06/12/2025).

[GSM25] GSMA. *Securing the 5G Era*. en. June 2025. URL: https://www.gsma.com/solutions-and-impact/technologies/security/securing-the-5g-era/ (visited on 06/12/2025).

[GÜN25] ALİ GÜNGÖR. *UERANSIM: Open Source 5G UE and ran (gNodeB) implementation.* June 2025. URL: https://github.com/aligungr/UERANSIM (visited on 06/12/2025).

[Int19] GSMA Intelligence. *The 5G Guide: A Reference for Operators*. en. GSMA Intelligence, Apr. 2019. URL: https://www.gsma.com/wp-content/uploads/2019/04/The-5G-Guide_GSMA_2019_04_29_compressed.pdf (visited on 06/12/2025).

[IOO23] Joseph Isabona, Emughedi Oghu, and Okiemute Omasheye. "Path Loss and Models: A Survey and Future Perspective for Wireless Communication Networks". In: *International Journal of Advanced Networking and Applications* 15 (Sept. 2023), pp. 5892–5907. DOI: 10.35444/IJANA.2023.15209. (Visited on 06/12/2025).

[Kar+23] Bedran Karakoc et al. "Never Let Me Down Again: Bidding-Down Attacks and Mitigations in 5G and 4G". In: *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. WiSec '23. Guildford, United Kingdom: Association for Computing Machinery, 2023, pp. 97–108. ISBN: 9781450398596. DOI: 10.1145/3558482.3581774. URL: https://doi.org/10.1145/3558482.3581774 (visited on 06/12/2025).

[Kli+23] Daniel Klischies et al. "Instructions Unclear: Undefined Behaviour in Cellular Network Specifications". In: *32nd USENIX Security Symposium (USENIX Security 23)*. Anaheim, CA: USENIX Association, Aug. 2023, pp. 3475–3492. ISBN: 978-1-939133-37-3. URL: https://www.usenix.org/conference/usenixsecurity23/presentation/klischies (visited on 06/12/2025).

[Kor23] Juha Korhonen. *Scheduling*. en. May 2023. URL: https://www.3gpp.org/technologies/scheduling (visited on 06/12/2025).

[LG21] Coleago Consulting Ltd and GSMA. *Estimating the Mid-Band Spectrum Needs in the 2025-2030 Time Frame*. en. GSMA, July 2021. URL: https://www.gsma.com/connectivity-for-good/spectrum/wp-content/uploads/2021/07/Estimating-Mid-Band-Spectrum-Needs.pdf (visited on 06/12/2025).

[Mal19] Avijit Mallik. "MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE WORDS". In: *Cyberspace: Jurnal Pendidikan Teknologi Informasi* 2 (Jan. 2019), p. 109. DOI: 10.22373/cj.v2i2.3453. (Visited on 06/12/2025).

[Mam+23] Lusani Mamushiane et al. "Deploying a Stable 5G SA Testbed Using srsRAN and Open5GS: UE Integration and Troubleshooting Towards Network Slicing". In: *2023 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD)*. 2023, pp. 1–10. DOI: 10.1109/icABCD59051.2023.10220512. (Visited on 06/12/2025).

[NKP22] Suresh Nair, Saurabh Khare, and Jing Ping. *Authentication and Key Management for Applications (AKMA) in 5G*. en. Dec. 2022. URL: https://www.3gpp.org/technologies/akma (visited on 06/12/2025).

[NYB23] NYBSYS. *Low to high 5G bands explained*. en. May 2023. URL: https://nybsys.com/5g-bands/ (visited on 06/12/2025).

[Obs23] European 5G Observatory. *Poland completes 3.6 GHz auction*. en. Oct. 2023. URL: https://5gobservatory.eu/poland-completes-3-6-ghz-auction/ (visited on 06/12/2025).

[Ook23] Ookla. *European 5G Performance Trails its International Peers*. en. Feb. 2023. URL: https://www.gsma.com/get-involved/gsma-membership/gsma_resources/european-5g-performance-trails-its-international-peers/ (visited on 06/12/2025).

[Ope25a] Open5GS. *Open5GS*. en. June 2025. URL: https://github.com/open5gs/open5gs (visited on 06/12/2025).

[Ope25b] OpenAirInterface. *OpenAirInterface 5G Radio Access Network Project*. en. June 2025. URL: https://openairinterface.org/oai-5g-ran-project/ (visited on 06/12/2025).

[Ope25c] OpenAirInterface. *OpenAirInterface5G*. en. June 2025. URL: https://gitlab.eurecom.fr/oai/openairinterface5g.

[Pal+21] Ivan Palamà et al. "IMSI Catchers in the wild: A real world 4G/5G assessment". In: *Computer Networks* 194 (2021), p. 108137. ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2021.108137. URL: https://www.sciencedirect.com/science/article/pii/S1389128621002061 (visited on 06/12/2025).

[Pra+18] Anand Prasad et al. *3GPP 5G Security*. en. Aug. 2018. URL: https://www.3gpp.org/news-events/3gpp-news/sec-5g (visited on 06/12/2025).

[Pyc25] Pycrate-Org. *Pycrate*. en. June 2025. URL: https://github.com/pycrate-org/pycrate (visited on 06/12/2025).

[Rom+19] S. Rommer et al. *5G Core Networks: Powering Digitalization*. Academic Press, Nov. 2019. ISBN: 9780081030103. URL: https://books.google.be/books?id=82C-DwAAQBAJ (visited on 06/12/2025).

[Ryu25a] Jaeku Ryu. *5G/NR - MAC*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_MAC.html (visited on 06/12/2025).

[Ryu25b] Jaeku Ryu. *5G/NR - N1 and S1*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_N1vsS1.html (visited on 06/12/2025).

[Ryu25c] Jaeku Ryu. *5G/NR - NAS*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_Registration.html#Registration_Request (visited on 06/12/2025).

[Ryu25d] Jaeku Ryu. *5G/NR - Network Architecture - AMF*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_Core_AMF.html (visited on 06/12/2025).

[Ryu25e] Jaeku Ryu. *5G/NR - Network Architecture - N26*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_NetworkArchitecture_N26.html#N26_Architectures (visited on 06/12/2025).

[Ryu25f] Jaeku Ryu. *5G/NR - Network Architecture - SMF*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_Core_SMF.html (visited on 06/12/2025).

[Ryu25g] Jaeku Ryu. *5G/NR - PDU Session Establishment in Detail*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_PDUSessionEstablishment.html (visited on 06/12/2025).

[Ryu25h] Jaeku Ryu. *5G/NR - RRC Overview*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_RRC_Overview.html (visited on 06/12/2025).

[Ryu25i] Jaeku Ryu. *5G/NR - RRC Reconfiguration*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_RRC_Reconfiguration.html (visited on 06/12/2025).

[Ryu25j] Jaeku Ryu. *5G/NR - UE Capability*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_UE_Capability.html (visited on 06/12/2025).

[Ryu25k] Jaeku Ryu. *5G/NR - UE IDs*. en. June 2025. URL: https://www.sharetechnote.com/html/5G/5G_UEID.html (visited on 06/12/2025).

[SB24] Matthew Shanahan and Kalvin Bahia. *The State of Mobile Internet Connectivity 2024*. en. GSMA, Oct. 2024. URL: https://www.gsma.com/r/wp-content/uploads/2024/10/The-State-of-Mobile-Internet-Connectivity-Report-2024.pdf (visited on 06/12/2025).

[SC22] Stefano Suardi and Pau Castells. *The Socio-Economic Benefits of Mid-Band 5G Services*. en. GSMA, Feb. 2022. URL: https://www.gsma.com/connectivity-for-good/spectrum/wp-content/uploads/2024/12/Mid-band-5G-Spectrum-Benefits.pdf (visited on 06/12/2025).

[SLY17] Peter Schmitt, Bruno Landais, and Frank Yong Yang. *Control and User Plane Separation of EPC nodes (CUPS)*. en. July 2017. URL: https://www.3gpp.org/news-events/3gpp-news/cups (visited on 06/12/2025).

[Sul22] Alain Sultan. *5G System Overview*. en. Aug. 2022. URL: https://www.3gpp.org/technologies/5g-system-overview (visited on 06/12/2025).

[Sys25a] Software Radio Systems. *CU-CP*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/CU_cp/source/index.html (visited on 06/12/2025).

[Sys25b] Software Radio Systems. *CU-UP*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/CU_up/source/index.html (visited on 06/12/2025).

[Sys25c] Software Radio Systems. *DU-high*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/DU_high/source/index.html (visited on 06/12/2025).

[Sys25d] Software Radio Systems. *DU-low*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/DU_low/source/index.html (visited on 06/12/2025).

[Sys25e] Software Radio Systems. *Features and Roadmap*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/general/source/2_features_and_roadmap.html (visited on 06/12/2025).

[Sys25f] Software Radio Systems. *MAC*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/DU_high/source/mac.html (visited on 06/12/2025).

[Sys25g] Software Radio Systems. *O-RAN gNB Overview*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/knowledge_base/source/oran_gnb/source/index.html#oran-gnb-overview (visited on 06/12/2025).

[Sys25h] Software Radio Systems. *Software Architecture*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/dev_guide/source/software_arch/source/index.html (visited on 06/12/2025).

[Sys25i] Software Radio Systems. *srsRAN*. June 2025. URL: https://github.com/srsran/srsran_4g (visited on 06/12/2025).

[Sys25j] Software Radio Systems. *srsRAN 4G 23.11 Documentation*. June 2025. URL: https://docs.srsran.com/projects/4g/en/latest/ (visited on 06/12/2025).

[Sys25k] Software Radio Systems. *srsRAN 4G Features*. June 2025. URL: https://docs.srsran.com/projects/4g/en/latest/feature_list.html (visited on 06/12/2025).

[Sys25l] Software Radio Systems. *Srsran GNB with COTS UES*. en. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/tutorials/source/cotsUE/source/index.html (visited on 06/12/2025).

[Sys25m] Software Radio Systems. *srsRAN Project*. en. June 2025. URL: https://github.com/srsran/srsRAN_Project (visited on 06/12/2025).

[Sys25n] Software Radio Systems. *srsRAN Project Documentation*. June 2025. URL: https://docs.srsran.com/projects/project/en/latest/# (visited on 06/12/2025).

[Sys25o] Software Radio Systems. *UE Architecture*. June 2025. URL: https://docs.srsran.com/projects/4g/en/latest/usermanuals/source/srsue/source/1_ue_intro.html#ue-architecture (visited on 06/12/2025).

[VP17] Mathy Vanhoef and Frank Piessens. "Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2". In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. CCS '17. Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 1313–1328. ISBN: 9781450349468. DOI: 10.1145/3133956.3134027. URL: https://doi.org/10.1145/3133956.3134027 (visited on 06/12/2025).

[Wik24] Wikipedia. en. Sept. 2024. URL: https://en.wikipedia.org/wiki/Type_Allocation_Code (visited on 06/12/2025).

[Wil25] John Wilson. *Seeking guidance on ETWS testing with SIB6 implementation - srsRAN_Project discussion #1003*. Jan. 2025. URL: https://github.com/srsran/srsRAN_Project/discussions/1003 (visited on 06/12/2025).

[Yai23] Karim Yaici. *Mobile gaming in the Gulf region: 5G improves the experience, but latency remains an issue*. en. Nov. 2023. URL: https://www.gsma.com/get-involved/gsma-membership/gsma_resources/mobile-gaming-in-the-gulf-region-5g-improves-the-experience-but-latency-remains-an-issue/ (visited on 06/12/2025).

[YCC19] Chuan Yu, Shuhui Chen, and Zhiping Cai. "LTE Phone Number Catcher: A Practical Attack against Mobile Privacy". In: *Security and Communication Networks* 2019.1 (2019), p. 7425235. DOI: https://doi.org/10.1155/2019/7425235. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2019/7425235. URL: https://onlinelibrary.wiley.com/doi/abs/10.1155/2019/7425235 (visited on 06/12/2025).

[Zer25] ZeroMQ. *ZeroMQ*. en. June 2025. URL: https://zeromq.org/ (visited on 06/12/2025).