Provable Non-Frameability for 5G Lawful Interception

Felipe Boeira
felipe.boeira@liu.se
Linköping University
Linköping, Sweden

Mikael Asplund
mikael.asplund@liu.se
Linköping University
Linköping, Sweden

Marinho Barcellos
marinho.barcellos@waikato.ac.nz
University of Waikato
Hamilton, New Zealand

ABSTRACT

Mobile networks have grown in size and relevance, with novel applications in areas including transportation, finance, and health. The wide use of mobile networks generates rich data about users, raising interest in using such data for law enforcement and antiterrorism through Lawful Interception (LI). Countries worldwide have established legal frameworks to conduct LI, and technical standards have been created for its implementation and deployment, but without sufficient (and rigorous) security controls. While LI originated for benign purposes, we show in this paper that malicious entities could exploit it to frame users into suspicion of criminal activity. Further, we propose a solution for non-frameability, which we formally prove uphold desired properties even in scenarios where attackers completely infiltrate the operator networks. To perform the formal verification, we extend prior work with a more complete model of the fifth generation (5G) of mobile networks in the Tamarin prover.

CCS CONCEPTS

• **Security and privacy → Security protocols**; **Mobile and wireless security**; **Formal security models**.

KEYWORDS

5G, formal verification, lawful interception, non-frameability

ACM Reference Format:
Felipe Boeira, Mikael Asplund, and Marinho Barcellos. 2023. Provable Non-Frameability for 5G Lawful Interception. In *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '23), May 29-June 1, 2023, Guildford, United Kingdom.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3558482.3581780

1 INTRODUCTION

Mobile networks are increasingly being adopted for applications beyond basic communication services and include several social and business activities where our real and digital identities become intertwined. The fifth generation (5G) of mobile networks are being deployed with applications in e-health and autonomous vehicles. The next generation (6G) may expand even further to use cases such as telepresence. The security and privacy of these networks have improved with every new generation of mobile networks. However, privacy is rarely absolute and there are cases where data capture is desired. Lawful interception (LI) allows authorities to obtain communication data from targeted end users. It is regulated by laws and standards in many countries and has been widely used to investigate serious crimes and terrorism. According to the latest report by the Investigatory Powers Commissioner[1], more than 250 thousand data communication interceptions were performed during 2020 in the UK alone.

The design of LI mechanisms in mobile networks is specified by the 3rd Generation Partnership Project (3GPP) and standardised by entities such as the European Telecommunications Standards Institute (ETSI). An obvious concern is that access to LI mechanisms must be properly secured. However, even if they are secured from adversaries, weaknesses in 5G might also affect LI operations and their purpose. In particular, consider an attacker who can impersonate other end users. Such impersonation may result in framing the (potentially honest) end user into suspicion of crimes and using LI to obtain false evidence to be used in court or for political purposes. Unfortunately, current authentication mechanisms in 5G make this possible in several cases.

In this paper, we propose a *non-frameability* solution that protects honest users, guaranteeing the authenticity of their intercepted metadata even in the presence of strong adversaries. In line with results from the research community in recent years [9], we argue that protocol design needs to be formally verified to ensure that there are no logical design flaws. We build on a number of results (e.g., [10, 17, 34]) that have laid the groundwork for rigorous and trustworthy protocol design in mobile networks. We extend prior work with a formal definition of non-frameability as well as an extension of the formal models of 5G core networks to account for a more complete representation of key derivation and communication steps.

In order to demonstrate the frameability problem, we explore scenarios where attackers may subvert assumptions about the security of 5G network operators and associated supply chains. We discuss severe limitations that arise when adversaries may compromise certain communication channels among components of the core network or the provisioning of SIM cards. These conditions are not far-fetched since examples of such compromises have been observed in the wild[2][3]. We describe a solution to this problem and present proof that the design thwarts attacks and mitigates the frameability of end users.

Recently, Arfaoui et al. [8] proposed a solution for establishing encrypted channels that authorities may open in case of demand for lawful interception (similar to the idea of key escrow systems or

WiSec '23, May 29-June 1, 2023, Guildford, United Kingdom

ACM ISBN 978-1-4503-9859-6/23/05.
https://doi.org/10.1145/3558482.3581780

[1] Annual Report of the Investigatory Powers Commissioner 2020, available: https://www.ipco.org.uk/publications/annual-reports/

[2] LightBasin: A Roaming Threat to Telecommunications Companies, available: https://www.crowdstrike.com/blog/an-analysis-of-lightbasin-telecommunications-attacks/

[3] Two Billion Owned SIM Cards is a Real-Life Nightmare, available: https://www.kaspersky.com/blog/gemalto-sim-hack/7774/

the Clipper Chip heavily discussed around the 90s). In our work, we explore a different perspective, aligned with the Dutch position [40] on the dilemma between online privacy and national security. First, we believe it is counterproductive to weaken the security and privacy properties of honest users of mobile communication in favour of LI. Criminals and terrorists have many options to circumvent LI, including virtual private networks and anonymous mobile subscriptions purchased with cryptocurrencies. Second, with the ubiquitous deployment of end-to-end encryption, it is likely that LI will mostly rely on metadata (e.g. location, timestamps, communication patterns and peers, etc.) rather than the actual communication content (as in the case of instant messaging today). See Abelson et al. [6] for a in-depth discussion on LI and its potential impact on society.

Our contributions are as follows:

- We extend the 5G formal model by Cremers and Dehnel-Wild [17] with 5G base stations, and include the initial context setup, non-access-stratum and access-stratum security mode commands. The extended model accounts for the security context establishment and more comprehensive key derivations used throughout user communication. We also update the model to the latest specifications from 3GPP and include the concealment of the permanent identity based on the work by Wang et al. [41].
- We outline how attackers may frame users into suspicion of criminal activity by exploiting an operator's infrastructure. We show that the attacker only requires read-only access to any of the core network channels used during authentication in order to be able to conduct the impersonation and frameability attack.
- We design a non-frameability solution for 5G lawful interception under adversaries that may compromise communication channels of the operator's network or SIM card supply chains. The solution thwarts impersonation even when attackers have control of channels used during authentication. In the worst case, when the attackers can also subvert the operator's private key, we still allow the end user to dispute malicious traffic injected due to a compromised operator. The solution's security is proven under the symbolic model using the Tamarin prover.

The paper is organised as follows: Section 2 provides relevant background. Section 3 presents the problem of susceptibility of mobile networks to impersonation when attackers obtain access to internal channels of the core network. Section 4 describes our non-frameability solution proposal designed for compatibility with the latest 5G specifications, and Section 5 presents the formal verification of the solution. Finally, Section 6 reviews the related work, and Section VII concludes the paper by outlining future work.

2 PRELIMINARIES

In this section, we provide the background information to make this work self-contained, including lawful interception in 5G and protocol security verification.

2.1 Lawful Interception Overview

Lawful interception provides the ability for authorised law enforcement agencies to acquire the communication content and metadata (e.g. location information and connection timestamps) from target users in mobile communication networks. In 5G, 3GPP has an ongoing specification effort of LI under the following technical specifications: TS 33.126 [2], TS 33.127 [1], and TS 33.128 [3].

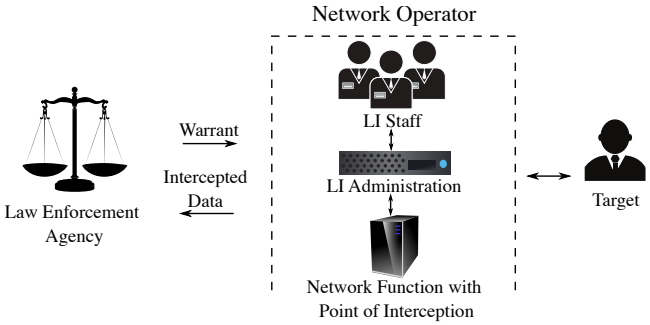

Figure 1: Overview of lawful interception in mobile networks.

Fig. 1 provides an overview of lawful interception in mobile networks. The Law Enforcement Agency (LEA) is an entity that is legally allowed to conduct an interception of mobile communications and has agreements with network operators. To start an interception, the LEA prepares a warrant containing the target's identity to be intercepted. In the operator, the staff responsible for managing LI operations uses the administrative functions to configure the request included in the warrant. The administrative functions are subsequently used to provide interception points in the operator's network components to collect data. Whenever the target communicates through the mobile network infrastructure, the collected data is forwarded to the law enforcement agency for investigation.

2.2 5G Architecture and Protocols

We focus on the components and protocols of the 5G architecture relevant to user authentication, communication with the data network (i.e. the Internet), and lawful interception. Fig. 2 depicts a simplified architecture: a User Equipment **(UE)** is managed by an end user to connect to the network and obtain services. The UE is usually a smartphone but can also represent other devices such as vehicles or drones. The base station (denoted as **gNB**) serves as a wireless access point to connect the UE to components of the core network in the operator so that the end user may use its services.

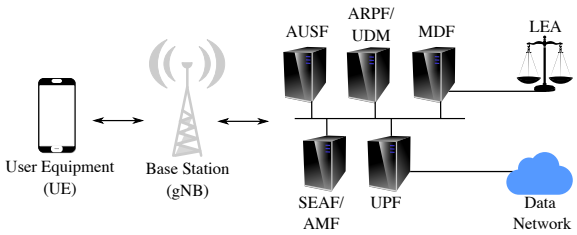

Figure 2: Simplified 5G architecture.

The Access and Mobility Function **(AMF)** has several functionalities, including authentication/authorisation and mobility management of the UE across base stations. It is co-located with the Security

Anchor Function **(SEAF)** employed by a serving network during authentication and key agreement. The Authentication Server Function **(AUSF)** generates the anchor key from the authentication material received from the Authentication credential Repository and Processing Function **(ARPF)**, co-located with the Unified Data Management **(UDM)** which holds the private key of the operator. The User Plane Function **(UPF)** performs packet routing to the data network as its main role. Most components in the 5G network are equipped with a point of interception that forwards collected traffic and metadata to the Mediation and Delivery Function **(MDF)**. The MDF relays the data to a law enforcement agency in case there is an ongoing lawful interception operation for the connecting user (i.e. a target). While we represent the functions as separate computing nodes in Fig. 2, 5G may use a service-based architecture, with functions deployed as virtual machines part of a cloud infrastructure rather than standalone bare-metal nodes.

Authentication and Key Agreement (AKA): In 5G, there are three protocol options for performing authentication and key agreement: 5G AKA, EAP-AKA', and EAP-TLS. EAP-TLS has limited use cases [43] (e.g. private networks) and 5G AKA and EAP-AKA' only differ slightly in the derivation of keys. We focus on 5G AKA due to its prevalence and existing security models. 5G AKA aims to derive communication session keys and achieve mutual authentication between the entities.

Initial Context and Security Mode Commands: While AKA is used to perform mutual authentication and derive anchor keys, the actual keys used throughout user communication are derived only in subsequent interactions. Once authentication is successful, the SEAF/AMF initiates the Non-Access-Stratum Security Mode Command (NAS SMC) procedure with the user equipment to start integrity protection and ciphering within their interface. Then, the SEAF/AMF shall derive the key for the base station and transmit it as part of the initial context setup. This will trigger the Access-Stratum Security Mode Command (AS SMC) between the base station and user equipment to derive integrity and ciphering keys to be activated in their communication henceforth.

Protocol Data Unit Session: In 3GPP specifications [4], Protocol Data Unit (PDU) session is the terminology used for a data communication session established for the user equipment. The establishment of a PDU session is initiated by the SEAF/AMF with another core network component called session management function (omitted from Fig. 2 for brevity) that communicates with the UPF. From then on, the user equipment may communicate with the data network (usually the Internet).

2.3 Protocol Security Verification

There are two main abstraction approaches to performing security verification of cryptographic protocols: computational and symbolic [13]. The computational model [24, 25] considers terms as bitstrings, cryptographic algorithms are functions over bitstrings, and the adversary is represented as a probabilistic polynomial-time Turing machine. The symbolic model abstracts the bitstrings as algebraic terms, and an equational theory captures the expected properties of cryptographic algorithms. In our work, we employ Tamarin, a state-of-the-art cryptographic protocol verification tool in the symbolic model. Tamarin has been used to verify many protocols that exhibit complex state machines that may include loops and agent memory [10, 18–20, 29].

The Tamarin solver maintains a state multiset of facts that can be consumed as premises to activate a rule. The action facts are logged into a trace of protocol execution and the set of traces is used to verify properties such as the example below. Variables starting with '#' are time points, and $a_n@i$ specifies that the action fact a_n occurred at time point i. The formula below states that for all traces where the action fact a_1 is logged with the term x, it implies that there exists a log of a_2 with the same term and it occurred before a_1.

```
lemma example:
    "All x #i. a1(x)@i ==> Ex #j. a2(x)@j & j<i"
```

To reason about attacker actions, Tamarin models network communication under a Dolev-Yao [21] threat model. This represents a strong attacker that is able to capture messages between entities, decompose and compose messages, apply cryptographic algorithms to known terms, and replay/drop messages. For a more detailed discussion and presentation of Tamarin we refer the reader to [16, 22, 23, 32, 39]. Furthermore, we also refer to Barbosa et al. [9] where they present a systematization of the computer-aided cryptography literature and discuss not only the symbolic and computational models, but also functional correctness and implementation-level security, which are out of the scope of this paper.

2.4 Elliptic Curve Cryptography

Elliptic curve cryptography has been proposed and developed since the 1980s [28, 33], and several primitives have been developed based on elliptic curves since then. We briefly present two of them: Elliptic Curve Integrated Encryption Scheme (ECIES) and Elliptic Curve Digital Signature Algorithm (ECDSA) employed in 5G and our solution.

ECIES is a hybrid encryption scheme that leverages elliptic curves to derive an ephemeral key that is used in a symmetric encryption algorithm to generate the ciphertext from the desired plaintext data. We follow Wang et al. [41] in the notation as we leverage their modeling approach in our work, and we refer the reader to their paper for more details. The agents are configured to use a set of standardised parameters, which include elements such as the curve equation, generator point, and the prime order. Public/private key pairs may be generated by a key generation function using these parameters, and encryption/decryption are intuitively performed as follows: $\mathsf{Encap}_{\mathsf{ECIES}}(\mathsf{pk}(\mathsf{x}))$ derives through the public key of the receiver ($\mathsf{pk}(\mathsf{x})$) an ephemeral key $\mathsf{k}_{\mathsf{ECIES}}$ and a cipher C_0 (used by the receiver to derive the same ephemeral key). The sender employs an authenticated symmetric algorithm to encrypt the data ($\mathsf{C} \leftarrow \mathsf{SEnc}_{\mathsf{ECIES}}(\mathsf{data}, \mathsf{k}_{\mathsf{ECIES}})$) and transmits $\{\mathsf{C}, \mathsf{C}_0\}$ to the receiver. The receiver derives the ephemeral key using its private key ($\mathsf{k}_{\mathsf{ECIES}} \leftarrow \mathsf{Decap}_{\mathsf{ECIES}}(\mathsf{x}, \mathsf{C}_0)$) and decrypts the data ($\mathsf{SDec}_{\mathsf{ECIES}}(\mathsf{C}, \mathsf{k}_{\mathsf{ECIES}})$).

ECDSA is a digital signature scheme based on elliptic curves that guarantee authenticity, integrity, and non-repudiation properties about the signed data from an agent. Like ECIES, the agents are configured to use a set of parameters and can generate public/private keys based on these parameters. To sign data, an agent executes $\mathsf{Sign}_{\mathsf{ECDSA}}(\mathsf{data}, \mathsf{x})$ with its private key x and generates

the signature σ. The receiver executes $\text{Verify}_{\text{ECDSA}}(\sigma, \text{data}, \text{pk}(\text{x}))$ to verify the validity of the signature concerning the data and the corresponding public key $\text{pk}(\text{x})$ of the sender, and may reject the message in case the verification fails. We refer to Johnson et al. [26] for a more comprehensive presentation of ECDSA.

3 SUSCEPTIBILITY OF MOBILE NETWORKS TO IMPERSONATION AND FRAMEABILITY

This section presents details about the problem and how an attacker may achieve the impersonation of honest users in mobile networks. We first provide the threat model to contextualise the attacker's capabilities, then outline how impersonation may be carried out and show the results of the security analysis.

3.1 Threat Model

Our threat models have distinct capabilities concerning communication channels and cryptographic keys compromise. In a communication channel (we use C to denote the capabilities related to channels), we consider two types of capabilities: Read-Only (C_{RO}) and Dolev-Yao (C_{DY}). A C_{RO} capability corresponds to the attacker obtaining access to the content of a channel assumed to be secured (e.g. IPSec or HTTPS) but without the possibility to drop, modify or inject messages. These are provided in C_{DY}, which also assumes access to the secured communication where applicable (i.e. in the internal channels of the operator's core network) and the possibility to manipulate its transmitted messages. The manipulation follows standard Dolev-Yao capabilities, such as intercepting and decomposing messages, applying public functions and cryptographic operations to construct new messages that may be transmitted.

We clarify that these channel threat models do not require Transport Layer Security (TLS employed in HTTPS) or IPSec to be broken by an attacker. For example, an attacker can exploit vulnerabilities in the software running in the network functions, obtaining access to the data at the application layer before it is encrypted and transmitted in the secure channel. Similarly, a disgruntled (or bribed) operator employee could be used to achieve this.

Capabilities regarding the reveal (compromise) of cryptographic keys (denoted using $\mathcal{R}$) are divided into the user's symmetric key ($\mathcal{R}_{\text{k}}$), home network private asymmetric key ($\mathcal{R}_{\text{HN}}$), and non-frameability private asymmetric key of *other end users* ($\mathcal{R}_{\text{NF}^*}$). The $\mathcal{R}_{\text{k}}$ represents a myriad of scenarios, including the compromise of the protocol or operator's servers used for remote provisioning of eSIMs, the compromise of SIM card manufacturers, and the compromise of over-the-air provisioning of SIM cards. It also captures the possible compromise of the SIM card through malware in the user's device (however, in this work, we assume that the user keeps their device secure). The $\mathcal{R}_{\text{HN}}$ represents an attacker that may have leaked the private key of the home network or has access to the device that stores it and is able to interact with it to decrypt data that has been encrypted based on the corresponding public key or sign data of its choice. Finally, $\mathcal{R}_{\text{NF}^*}$ represents the reveal of the non-frameability private key of *other end users* to the attacker, which allows the attacker to obtain any generated key for non-frameability except the one for the end user being checked.

As in other works [10, 17, 41], we consider C_{DY} on the communication between the user and the base station as this is the public wireless channel. In other channels, both C_{DY} and C_{RO} may be employed, and for each verification result, we indicate the corresponding capabilities. Likewise, we also specify the key compromised capabilities for the individual results.

Compromising core network channels and services requires a powerful attacker which could, for example, be composed of criminal organisations, malicious insiders, or governmental entities (not mutually exclusive) that have enough resources to conduct these attacks. The compromise of components in the system could be performed through technical (e.g. 0-day exploits), financial (e.g. by bribing operator's employees), or coercive means (e.g. authoritarian or corrupt members of governmental agencies). Potential capabilities related to modifying the software that runs on the network functions are outside the scope of our work.

3.2 Modeling 5G

Our work is based on formal methods using the symbolic model in the Tamarin prover. To construct our proofs, we significantly extend the models by Cremers and Dehnel-Wild [17]. First, we explicitly model the base station (gNB) so that it is possible to evaluate the impact of compromising its channel that is assumed to be secure. Second, we consider both non-access and access stratum security mode commands, and the UE context modification used in deriving several cryptographic keys.

Our resulting extended model[4] shows that read-only access to any of the channels is sufficient to compromise session keys, extending the scope adopted in [17]. In addition, we formally model our solution to prove its expected security properties. Our model is publicly available for researchers that may wish to reproduce our proofs or extend it further[5].

The left side in Figure 3 depicts the modelled interactions and the entities that participate in the communication. The model is composed of communication between five entities, and we represent them into five protocol interactions identified by distinct colours in the figure. As previously stated, we explicitly model the base station (gNB) as a separate entity and arrows that cross its vertical line represent that messages are relayed to the adjacent entity.

The right side in Figure 3 presents the key hierarchy captured by the 5G model, colour-coded and aligned with the corresponding interactions to the left. The pre-shared key K is used to derive intermediate and anchor keys, which are subsequently used to derive the keys that are used throughout the communication and mobility of the user equipment.

3.3 Impersonation Attack

An impersonation is feasible when an attacker can generate/transmit data that would be incorrectly perceived as belonging to a different user. Given our threat model, several scenarios exist in which impersonation may be carried out. Obvious scenarios stem from the description of $\mathcal{R}_k$ (cf. previous subsection), where the attacker would simply authenticate oneself with the network on behalf of the victim end user to use its services. In this subsection, we explore other scenarios where the attacker does not directly

[4] henceforth referred simply as the model.
[5] Available: https://gitlab.liu.se/ida-rtslab/public-code/2023_5g-nf

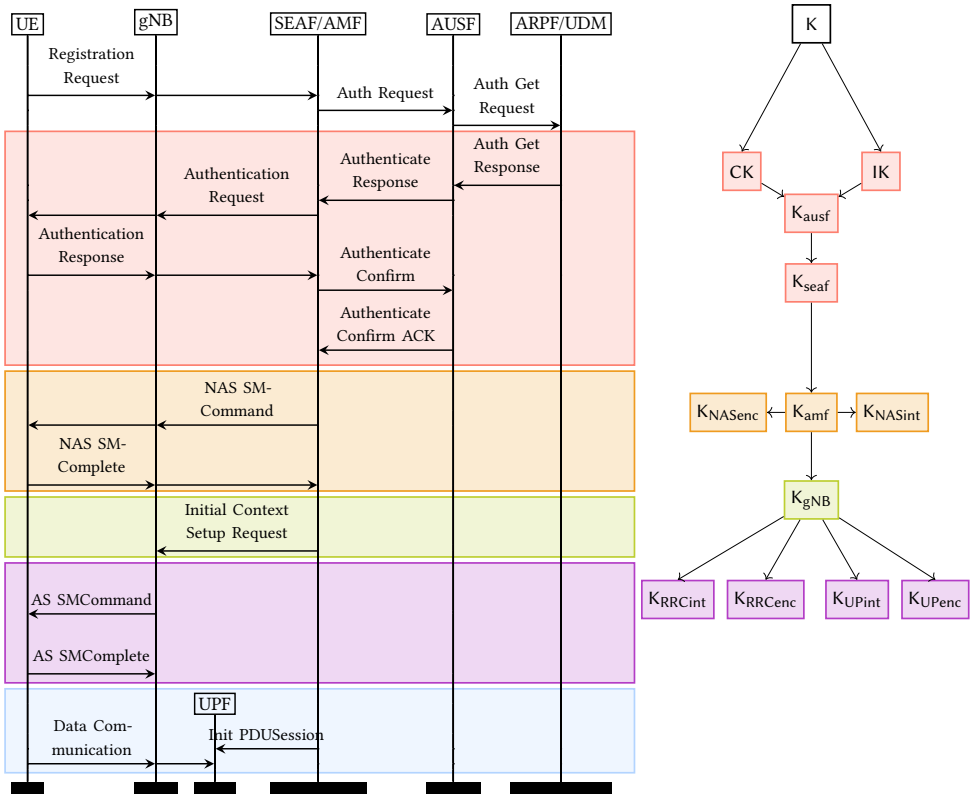

Figure 3: High-level sequence diagram of the modeled interactions in our analysis. The key hierarchy is depicted on the right side, and arrows represent Key Derivation Functions (KDFs). The top-level key K is pre-shared and stored in the SIM card and UDM.

obtain access to SUPI or K but rather compromises parts of the operator.

Table 1: Secrecy of keys when attackers may access certain communication channels.

(#) Channel read-only (C_{RO})	Cryptographic Keys Secrecy				
	K	K_{ausf}	K_{seaf}	K_{amf}	K_{gnb}
(1) UE ↔ gNB	✔	✔	✔	✔	✔
(2) gNB ↔ SEAF/AMF	✔	✔	✔	✔	✖
(3) SEAF/AMF ↔ AUSF	✔	✔	✖	✖	✖
(4) AUSF ↔ ARPF/UDM	✔	✖	✖	✖	✖

These scenarios include situations where an attacker may access (C_{RO}) individual channels in the core network of an operator. We complement the corresponding results shown in [17] with our extensions to the model and contextualise them in terms of frameability in LI. Table 1 presents the secrecy results for the cryptographic keys when the attacker may read the specified channel. A green tick means the key secrecy is maintained, whereas a red cross means the key is compromised.

As expected, intercepting the wireless communication between a user equipment and the base station (1) does not compromise the secrecy of any keys. However, reading the content of communication among certain core components allows the attacker to obtain and derive the keys necessary to impersonate the victim. This is possible for two main reasons: transmission of keying material between nodes and public parameters to key derivations. A concrete example of the former is the communication of anchor key K_{ausf} from the ARPF/UDM to the AUSF (cf. Fig. 4). Secondly, once an attacker has obtained an upper bound key of the hierarchy, deriving the lower bounds is possible due to the use of public parameters. For example, deriving K_{UPenc} requires knowledge of K_{gNB} and parameters such as a fixed identity of the algorithm and the length of the parameter (see 3GPP TS 33.501 A.8 [5]). Note that handover is outside the scope of our analysis, and we point the reader to Peltonen et al. [34] who conduct such analysis in the context of 5G handover protocols.

4 NON-FRAMEABILITY FOR 5G AND BEYOND

Our proposed solution aims to achieve non-frameability in adversarial scenarios, as presented in the previous section. The solution is divided into the following parts: initial setup, establishing the non-frameability security context, executing protocols under non-frameability, and lawful interception and dispute resolution. We remind the reader that the main objective of our solution is to make it possible for end users to have a defence mechanism against being

framed into criminal activity rather than guaranteeing that criminal activity can be traced to its practitioners. Furthermore, our solution works upon a person identification procedure at the subscription purchase time which falls outside the scope of our analysis.

4.1 Initial Setup

The initial setup is executed at the time that an end user is provisioned with a new identity and key *pair* (SUPI and K, respectively). This would correspond to the end user obtaining a SIM card or running remote provisioning of an embedded SIM (eSIM). We use a strict binding between the *pair* and the setup, i.e. the setup is only allowed to be executed once for a given *pair*. We believe this will not limit practical viability, especially when eSIM is employed and the end user may provision it over the air. This strictness prevents two scenarios where the attacker may have gained access to the *pair* of an end user either before or after the end user performs the setup. In the first case, the attacker could execute the setup before the user, but that would result in the user not being able to set up and therefore proceeding to replace its *pair*. In the second case, the attacker is not able to set up.

We now outline the steps of our setup protocol. We abstract the other core network components for simplicity, but our security analysis considers Dolev-Yao attackers in the setup communication as well. The identity SUPI and key K are provisioned to the user by its home operator according to standard mobile networks design. In 5G, the user also obtains the public key of the home network ($\mathsf{pk}(\mathsf{HN}_\mathsf{k})$), which we also leverage to use Elliptic Curve Integrated Encryption Scheme (ECIES) and the Elliptic Curve Digital Signature Algorithm (ECDSA).

The setup begins with the user generating a fresh elliptic curve key NF and calculating the associated public curve point denoted as pk(NF). The user constructs a setup request on the NF key and commits to keeping it secure to prevent frameability. The user encrypts the current timestamp t_req with the pre-shared key K (to prove its custody), and appends its identity SUPI and pk(NF) to create sReq, which is subsequently signed with NF using ECDSA. The user setup request consists of the signature and encryption of sReq using ECIES with the home operator's public key. This prevents eavesdroppers from obtaining the SUPI and mapping it to pk(NF).

The operator receives the end user's setup request and uses its private key HN_k to decrypt and obtain sReq and pk(NF). The operator obtains the SUPI and, if there is no existing setup for this identity, retrieves K from its database so that t_req can be decrypted. At this point, the operator checks t_req for freshness and performs signature verification of sReqSig.

To construct the binding, the home network signs a timestamp t_res, the SUPI, and the received pk(NF) from the user to create σ^{setup}. The operator encrypts σ^{setup} and t_res using ECIES with pk(NF) and sends the setup response. Hence, the operator commits to accepting communication if and only if the user knows the private key NF and will reject any future setups for this SUPI.

Upon reception of the setup response, the user performs appropriate checks of signature and timestamp. The binding $\beta^{\mathsf{SUPI}} := \{\mathsf{t}_\mathsf{res}, \sigma^{\mathsf{setup}}, \mathsf{pk}(\mathsf{NF})\}$ can be constructed and saved in the UE. We also note that this binding could be stored online but leave this outside the scope of the paper.

4.2 Establishing the Security Context for Non-Frameability

Every communication session becomes associated with what we term a security context for non-frameability $\mathsf{Ctx}^{\mathsf{NF}}$. $\mathsf{Ctx}^{\mathsf{NF}}$ is established during a modified authentication and key agreement AKA and used throughout other communication protocols during the lifecycle of the user communication.

Fig. 4 provides details about the authentication and key agreement and our design changes shown in green. Step ① remains unmodified and we leverage the ephemeral key $\mathsf{k}_{\mathsf{ECIES}}$ derived by $\mathsf{Encap}_{\mathsf{ECIES}}$ in the generation of the signature σ^{NF} in later steps. Step ② generates the home environment authentication vector (HE AV), which is used to challenge the user in the authentication process. The binding β^{SUPI} associated with that subscription is fetched from the UDM and, in Step ③, it is transmitted to the AUSF in addition to the ephemeral key $\mathsf{k}_{\mathsf{ECIES}}$. These terms are not relayed to the SEAF as this would allow a serving network to immediately obtain the user's SUPI. Therefore, we comply with the current 5G specifications and only allow serving networks to obtain this term when authentication is confirmed. In Step ④, the user equipment produces the challenge response RES^*. A signature is generated based on the relevant data to prove knowledge of the private key NF. The reasoning for the choice of data elements in the signature generation is as follows: RES^* is included to tie the signature to the current authentication response instance, gNB_id prevents an attacker from replaying the signature to another gNB. The ephemeral key $\mathsf{k}_{\mathsf{ECIES}}$ is included to prevent both eavesdroppers and the serving network from identifying the identity of the user (e.g. by recovering the ECDSA public key [26] and testing against a database of known public keys). Finally, a timestamp t prevents a replay of the signature (even to the same gNB) by checking freshness. In Step ⑤ the resulting signature σ^{NF} and timestamp t are transmitted to the gNB, which relays to the SEAF/AMF, and subsequently to the AUSF in Step ⑥. The signature can be verified in Step ⑦. Because the standard defines that at this point the serving network may receive the SUPI, we also provide in Step ⑧ the ephemeral key and the binding β^{SUPI} so that the serving network may verify the binding and the signature provided by the user in the authentication response. The signature can now be verified in Step ⑨ and therefore the non-frameability security context $\mathsf{Ctx}^{\mathsf{NF}} := \{\mathsf{SUPI}, \beta^{\mathsf{SUPI}}, \mathsf{RES}^*, \mathsf{k}_{\mathsf{ECIES}}, \sigma^{\mathsf{NF}}, \mathsf{t}, \mathsf{gNB}_\mathsf{id}\}$ is established and logged in the point of interception by the SEAF.

The subsequent interactions are depicted in Fig. 5 and contain minor changes in order to propagate the security context $\mathsf{Ctx}^{\mathsf{NF}}$. The NIA function represents the integrity algorithm used to generate the Message Authentication Codes (MACs) according to the specifications. Furthermore, we use $\mathsf{senc}(x, y)$ to denote the symmetric encryption of the term x with the key y. Our design changes require the transmission of $\mathsf{Ctx}^{\mathsf{NF}}$ in Steps ⑩ and ⑪. Finally, Step ⑫ represents the data communication from the user equipment that must include a digital signature with the private non-frameability key NF. We note that a practical implementation should employ an

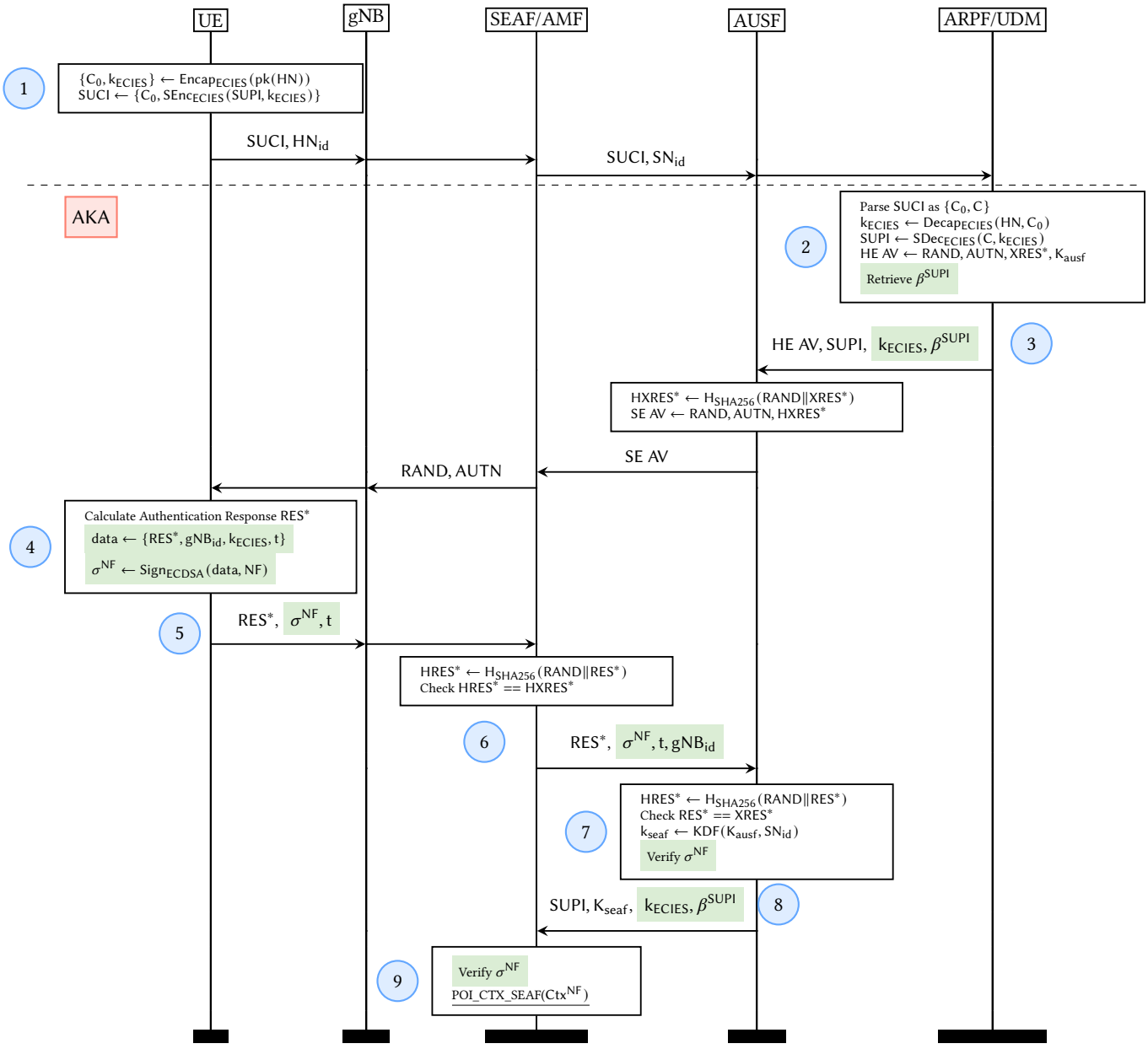

Figure 4: The establishment of the non-frameability security context Ctx^{NF} in AKA. The highlighted text denotes our modifications to the protocol. Underlined text aid in the presentation of formal verification in Section 5.

efficient signature scheme [11, 31, 35] that must leverage the established security context to provide the security guarantees under the high throughput of 5G networks (this performance aspect is out of scope).

We also recall that the implementation of ECDSA must avoid the reuse of nonces during signature generation and even leaks of one bit from the nonce [7]. Although there are several cryptographic primitives that could be leveraged for our purpose, we opted to minimise the design complexity as long as the expected properties can be guaranteed.

4.3 Non-Frameable LI and Dispute Resolution

In our solution, lawful interception is a tuple $(Ctx^{NF}, \mathcal{T}, \mathcal{D})$, where Ctx^{NF} is the non-frameability security context established during AKA, $\mathcal{T}$ is the intercepted traffic and D is the set of decryption keys that the operator may need to provide to the LEA (e.g. k_{UPenc}).

Algorithm 1 in Appendix A shows the steps for verification of the intercepted traffic. Before verifying the captured packets, the LEA checks that the provided signature verifies the binding of the target SUPI. Then, the signature present in the security context for

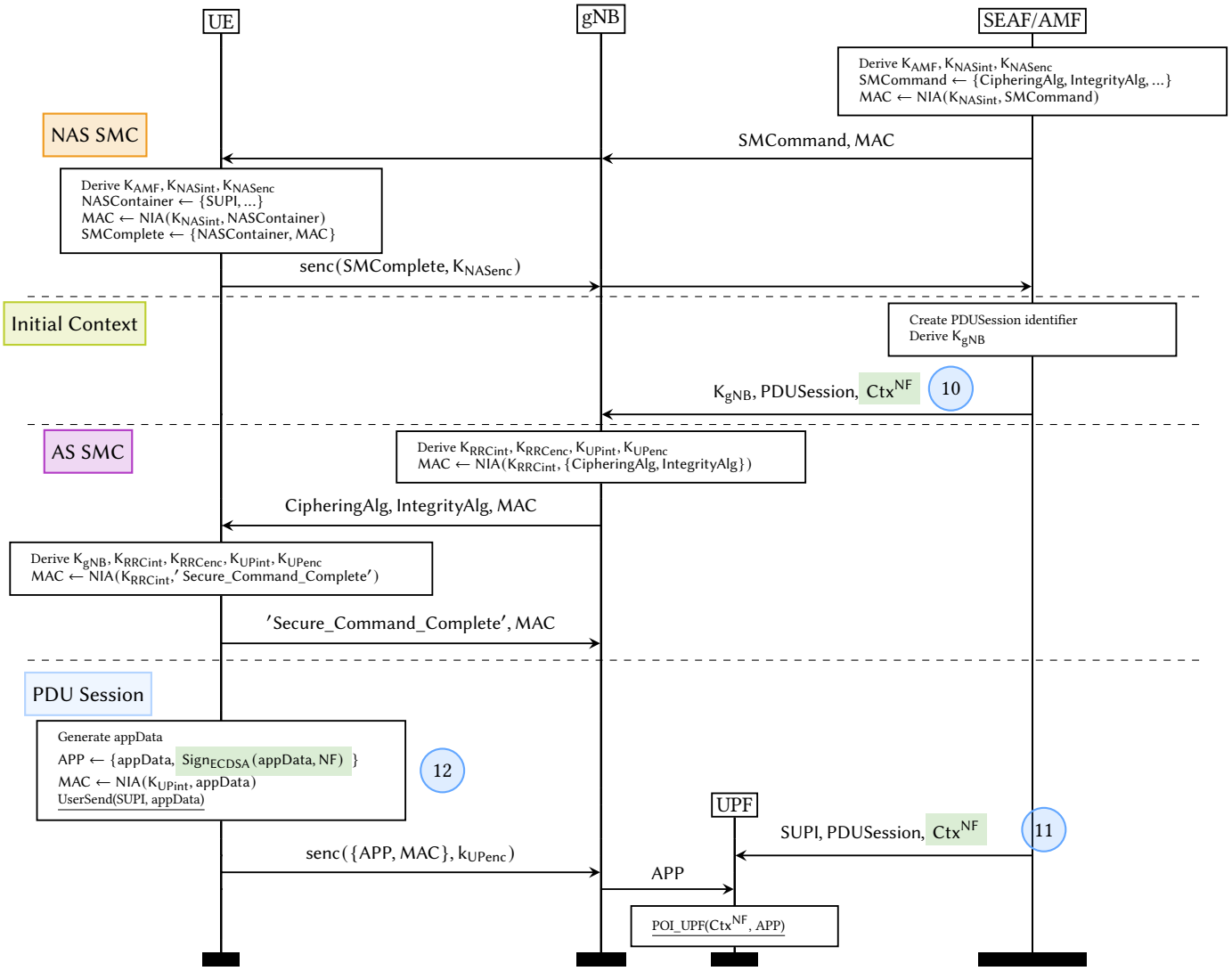

Figure 5: Sequence diagram of the interactions following 5G AKA. The highlighted text denotes our modifications to the protocol. Underlined text aid in the presentation of formal verification in Section 5.

non-frameability is also verified. If both verify, then the traffic may also be checked for authenticity.

The user may dispute the intercepted traffic in a later stage (e.g. when notified by the LEA or during the trial) by providing the binding obtained during setup. The dispute is rejected if the binding is the same as the one received within the context Ctx^{NF} because all required checks have already passed. However, the dispute is accepted if they are not equal and the user's binding verifies for the intercepted SUPI. According to the solution design, the operator must produce at most one binding per SUPI, otherwise, it is evidence that their infrastructure has been compromised.

5 FORMAL VERIFICATION OF SECURITY

The protocols are modelled in the Tamarin prover to verify the security properties under the presence of adversaries. The base model by Cremers and Dehnel-Wild [17] is substantially extended (their model focused solely on AKA) to support our setup protocol and the interactions depicted in Figs. 4 and 5. Furthermore, we have updated the model according to the latest versions of the technical specifications by 3GPP and also included the SUPI concealment using ECIES based on the work by Wang et al. [41]. This section provides details about modelling choices related to communication channels and key revealing, the properties that are checked, and the summarised results.

Table 2: Results marked with ✔ mean that frameability attempts are mitigated at run time. In other cases, ▲ indicates that the user must dispute the framed intercepted data. To ease visualisation, we highlight Dolev-Yao and Read-Only channels.

Key Reveal	Scenario	Communication Channels						Result
		UE ↔ gNB	gNB ↔ AMF	gNB ↔ UPF	AMF ↔ AUSF	AMF ↔ UPF	AUSF ↔ ARPF/UDM	
$\mathcal{R}_k$	1	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	✔
$\mathcal{R}_k \wedge \mathcal{R}_{NF^*}$	2	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	✔
$\mathcal{R}_{HN}$	3	C_{DY}	C_{RO}	C_{RO}	C_{RO}	C_{RO}	C_{RO}	✔
	4	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	▲
$\mathcal{R}_k \wedge \mathcal{R}_{HN}$	5	C_{DY}	C_{RO}	C_{RO}	C_{RO}	C_{RO}	C_{RO}	✔*
	6	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	C_{DY}	▲

5.1 Non-Frameability Verification

Formal verification of security properties commonly revolves around secrecy and authenticity variants [30], such as perfect forward secrecy or injective agreement. As discussed throughout this paper, we focus on the non-frameability of end users. It intuitively means that, with our proposed solution, communication data intercepted has indeed been generated by the end user, otherwise a dispute would reveal that there was an attempt to frame the end user. This property resembles *non-repudiation of origin*, which in the context of network protocols, is evidence to the recipient of a message that the origin is authentic and cannot falsely deny having sent the message [42]. In fact, non-repudiation of origin is achieved in our work in case a dispute is rejected according to Algorithm 1, provided that the user kept its device secure and did not share the private key.

To express the non-frameability property, we employ variants of the formula below. In each case, we adjust whether we allow certain keys to be exposed to the attacker or to check whether a dispute would be needed to assure non-frameability.

```
lemma LI_intercept_noRevs:
"All SUPI beta_SUPI RES_star k_ECIES sig_NF t gNB APP #li
     #seaf.
  POI_UPF(<SUPI, beta_SUPI, RES_star, k_ECIES, sig_NF, t,
     gNB>, APP)@li
& POI_CTX_SEAF(<SUPI, beta_SUPI, RES_star, k_ECIES,
    sig_NF, t, gNB>)@seaf
& not(Ex R #rev. Rev(R)@rev)  ==>
   (Ex #send. UserSend(SUPI, APP)@send & send<li)"
```

To conduct the analysis, several lemmas are constructed based on this example, and distinct model variants are instantiated to account for the different channel compromises considered in the evaluation. We found this approach to be more tractable by Tamarin prover rather than restriction formulas to specify which channel rules could be activated in each case, leading to more complex proving and non-termination issues. While it may create some inconvenience due to the need for handling several files, it does not weaken any results.

5.2 Verification Results

The results of the formal verification of our proposal are presented in Table 2. The notation for key reveals and channel compromises follows from Section 3.1. We classify the outcome into two categories: frameability block at run time (✔) and dispute required (▲). Note that dispute is only required in extreme cases, where the attacker has taken control over the operator infrastructure, including their private key.

We now describe some potential interpretations of the scenarios shown in Table 2. In Scenario 1, the attacker has obtained the 5G symmetric credentials of the user by, for example, exploiting the provisioning of SIM cards. Even with complete control over the communication channels, impersonation is not possible. In Scenario 2, the attacker has also managed to compromise other users' devices and obtain their non-frameability private key. However, the attacker cannot bind those keys to the victim's identity (whose device is secured), therefore mitigating impersonation attempts. In Scenario 3, the attacker leaked the operator's private key but can only read internal communications, therefore unable to induce the network to accept a forged binding and security context. This is captured in Scenario 4, where the victim may dispute the falsified data. In Scenarios 5 and 6, the attacker has obtained both the private key of the operator and the 5G symmetric credentials of the user. In this case, stopping the impersonation at runtime is only possible if the attacker has read-only access over internal channels and the compromise of the operator's private key occurred after the user performed its setup (✔*). A dispute is only required in the most severe cases, where the attacker also obtained data injection capabilities in the internal channels.

While using the symbolic model favours automation of the proving procedure, complex models often require manual help from humans to guide the prover to termination. In Tamarin, this can be accomplished through several methods, for example: writing *sources* lemmas to indicate the origin of terms, specifying *reusable* lemmas that serve as axioms in proving subsequent lemmas, and writing scripts (called *oracles*) that rank the order of goals to be chosen at each proving step. To obtain our proofs, we had to employ all of these methods and thousands of CPU core-hours at the *Anonymised* National Supercomputer Centre throughout development and proving.

5.3 Limitations

First, we remind the reader of the limitations associated with using the symbolic model for protocol security verification, and then we discuss some limitations inherent to our work.

In the symbolic model, cryptographic primitives are black-boxes (perfect cryptography); therefore, the model does not account for

weaknesses in these primitives. Contrary to the computational model, bitstrings are abstracted as algebraic terms which disregard their partial compromise (e.g. leak of some bits of a key). Furthermore, side-channel attacks are outside the scope of symbolic analysis.

With respect to our work, we simplify the messages exchanged in the protocols to focus on the security-related aspects (e.g. we omit physical and link layers in the model). We simplify the PDU session initialisation, which involves using the AMF to abstract the role of the Session Management Function (SMF) with a single initialisation message (the actual UE-requested PDU session establishment is composed of 21 steps [4]). In reality, we would only require that the security context (Ctx^{NF}) is sent to the UPF in one of those steps, which should not hinder the practicality of our proposal. Furthermore, we omit the Mediation and Delivery Function (MDF), as it serves as an intermediary between the points of interception (POIs) and the law enforcement agency. Since we allow the compromise of the components where POIs reside, this captures the potential compromise of the MDF to manipulate that data. Consequently, we believe that these limitations do not affect our results or the feasibility of the solution.

6 RELATED WORK

As the importance of mobile networks has increased over the years, so has the interest in research on mobile network security. Rupprecht et al. [36] present a comprehensive survey that covers currently active generations and discusses research gaps for future generations. In another survey by Khan and Martin [27] existing vulnerabilities in the LTE mobile networks are discussed in the context of 5G specifications, and remaining challenges for future work are outlined. The authors provide a comprehensive discussion of the vulnerabilities, which range from the physical layer to the network layer.

Examples of recent attacks discussed in the literature include eavesdropping, authentication, impersonation, and linkability (compromising users' privacy). Chlosta et al. [15] present a known technique for identity linkability in 5G along with possible threat scenarios. They implement and evaluate the attack denoted as SUCI-catcher (akin to IMSI-catcher in 4G) using Free5GC and the Amarisoft gNodeB to demonstrate the practicality of the attack. Bitsikas and Pöpper [12] discuss the exploitation of measurement reports provided by user equipment that is used in mobile networks so that an attacker is able to fool base stations to handover their users to a false base station. Even though the attacker does not possess the security context to impersonate the operator, the authors have described the possibility of performing denial of service, Machine-in-the-Middle between the user and the real operator, and identity leaking by forcing the user to send an attach request. Rupprecht et al. [37] discuss an attack on Voice over LTE (VoLTE), a packet-based telephony service. The vulnerability exploited in the attack stems from keystream reuse when two calls occur within one radio connection. Although the attack compromises the confidentiality of calls (VoLTE), it does not allow attackers to impersonate the victim. In another work, Rupprecht et al. [38] present an attack in the LTE mobile network that allows an attacker to break mutual authentication in both directions. The attack assumes a powerful malicious actor that can place Machine-in-the-Middle between the user and the base station, and between the operator and the target servers. This allows full impersonation of a user. Our attack model differs from these works since we focus on (partially) compromised operator components. Moreover, we also provide a verified solution mechanism for the attacks discussed in this work.

Formal verification of security properties in 5G was pioneered by Basin et al. [10] who first formalised the 5G authentication and key agreement protocol and verified security properties using Tamarin. The authors found that security goals and assumptions were underspecified or missing. Cremers and Dehnel-Wild [17] provide a more detailed model with four parties and several different compromise models. Our work further extends the work by Cremers and Dehnel-Wild by including the interactions with the base station (gNB) and also considers other security properties (non-frameability).

Other security solutions and mechanisms recently presented in the literature for mobile networks include the work by Zhao et al. [44] investigate vulnerabilities in SIM/e-SIM that may result in traffic eavesdropping, man-in-the-middle, and impersonation attacks. The authors validate the attacks through a 4G LTE testbed and over two US mobile operators. The authors focus on local attacks on the user equipment and (e-)SIM card, rather than on supply chain or mobile network component compromise as explored in this work. Akin to our work, An Braeken [14] proposes a modification to the 5G AKA protocol to mitigate attacks proposed in the literature. The protocol is verified under simplifying assumptions such as the unification of home and serving networks, and the unification of internal functions in the operator, such as the Unified data management, Authentication Credential Repository and Processing Function, and the Subscription Identifier De-concealing Function. The proposed protocol is verified with RUBIN logic under a Dolev-Yao threat model between the user equipment and the operator.

7 CONCLUSION AND FUTURE DIRECTIONS

We present the susceptibility of mobile networks to attacks involving core components and supply chains and discuss how these may have a severe impact in the context of lawful interception. We mainly focus on the frameability of honest users, and present the capabilities that threat agents may exploit to conduct attacks. We show, for example, that the attacker only requires read-only access to one of the channels assumed to be secure during authentication to derive keys necessary to impersonate victims. Impersonation leads to the possibility of framing the victim by engaging in criminal activities, and may be used to create reputation damage that cannot be timely repaired in some cases. We present a solution to support non-frameability for users that is designed based on 5G specifications and model it in Tamarin to prove its security under strong adversaries that may compromise cryptographic keys and infiltrate the operator networks. In future directions, the data communication in 5G could be integrated with our non-frameability security context by using efficient stream signing in order to support the higher throughput. Furthermore, other protocols must be integrated with the NF security context, such as the handover protocols to update the security context as the users move geographically.

REFERENCES

[1] 3rd Generation Partnership Project 2021. *Lawful Interception (LI) architecture and functions.* 3rd Generation Partnership Project. TS 33.127 V17.1.0.

[2] 3rd Generation Partnership Project 2021. *Lawful Interception requirements.* 3rd Generation Partnership Project. TS 33.126 V17.0.0.

[3] 3rd Generation Partnership Project 2021. *Protocol and procedures for Lawful Interception (LI).* 3rd Generation Partnership Project. TS 33.128 V17.1.0.

[4] 3rd Generation Partnership Project 2022. *Procedures for the 5G System (5GS).* 3rd Generation Partnership Project. TS 23.502 V17.5.2.

[5] 3rd Generation Partnership Project 2022. *Security architecture and procedures for 5G system.* 3rd Generation Partnership Project. TS 33.501 V17.4.2.

[6] Harold Abelson, Ross Anderson, Steven M. Bellovin, Josh Benaloh, Matt Blaze, Whitfield Diffie, John Gilmore, Matthew Green, Susan Landau, Peter G. Neumann, Ronald L. Rivest, Jeffrey I. Schiller, Bruce Schneier, Michael A. Specter, and Daniel J. Weitzner. [n. d.]. Keys under doormats: mandating insecurity by requiring government access to all data and communications. *Journal of Cybersecurity* ([n. d.]). https://doi.org/10.1093/cybsec/tyv009

[7] Diego F. Aranha, Felipe Rodrigues Novaes, Akira Takahashi, Mehdi Tibouchi, and Yuval Yarom. 2020. LadderLeak: Breaking ECDSA with Less than One Bit of Nonce Leakage. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. Association for Computing Machinery, New York, NY, USA, 225–242. https://doi.org/10.1145/3372297.3417268

[8] Ghada Arfaoui, Olivier Blazy, Xavier Bultel, Pierre-Alain Fouque, Thibaut Jacques, Adina Nedelcu, and Cristina Onete. 2021. How to (Legally) Keep Secrets from Mobile Operators. In *Computer Security – ESORICS 2021*, Elisa Bertino, Haya Shulman, and Michael Waidner (Eds.). Springer International Publishing, Cham, 23–43.

[9] Manuel Barbosa, Gilles Barthe, Karthik Bhargavan, Bruno Blanchet, Cas Cremers, Kevin Liao, and Bryan Parno. 2021. SoK: Computer-Aided Cryptography. In *2021 IEEE Symposium on Security and Privacy (SP)*. 777–795. https://doi.org/10.1109/SP40001.2021.00008

[10] David Basin, Jannik Dreier, Lucca Hirschi, Saša Radomirovic, Ralf Sasse, and Vincent Stettler. 2018. A Formal Analysis of 5G Authentication. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (Toronto, Canada) *(CCS '18)*. Association for Computing Machinery, New York, NY, USA, 1383–1396. https://doi.org/10.1145/3243734.3243846

[11] Rouzbeh Behnia and Attilla Altay Yavuz. 2021. Towards Practical Post-Quantum Signatures for Resource-Limited Internet of Things. In *Annual Computer Security Applications Conference* (Virtual Event, USA) *(ACSAC '21)*. Association for Computing Machinery, New York, NY, USA, 119–130. https://doi.org/10.1145/3485832.3488023

[12] Evangelos Bitsikas and Christina Pöpper. 2021. Don't Hand It Over: Vulnerabilities in the Handover Procedure of Cellular Telecommunications. In *Annual Computer Security Applications Conference* (Virtual Event, USA) *(ACSAC)*. Association for Computing Machinery, New York, NY, USA, 900–915. https://doi.org/10.1145/3485832.3485914

[13] Bruno Blanchet. 2012. Security Protocol Verification: Symbolic and Computational Models. In *Principles of Security and Trust*, Pierpaolo Degano and Joshua D. Guttman (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 3–29.

[14] An Braeken. 2020. Symmetric key based 5G AKA authentication protocol satisfying anonymity and unlinkability. *Computer Networks* 181 (2020), 107424. https://doi.org/10.1016/j.comnet.2020.107424

[15] Merlin Chlosta, David Rupprecht, Christina Pöpper, and Thorsten Holz. 2021. 5G SUCI-Catchers: Still Catching Them All?. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (Abu Dhabi, United Arab Emirates) *(WiSec '21)*. Association for Computing Machinery, New York, NY, USA, 359–364. https://doi.org/10.1145/3448300.3467826

[16] Véronique Cortier, Stéphanie Delaune, and Jannik Dreier. 2020. Automatic Generation of Sources Lemmas in Tamarin: Towards Automatic Proofs of Security Protocols. In *Computer Security – ESORICS 2020*, Liqun Chen, Ninghui Li, Kaitai Liang, and Steve Schneider (Eds.). Springer International Publishing, Cham, 3–22.

[17] Cas Cremers and Martin Dehnel-Wild. 2019. Component-Based Formal Analysis of 5G-AKA: Channel Assumptions and Session Confusion. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society.

[18] Cas Cremers, Martin Dehnel-Wild, and Kevin Milner. 2017. Secure Authentication in the Grid: A Formal Analysis of DNP3: SAv5. In *Computer Security – ESORICS 2017*, Simon N. Foley, Dieter Gollmann, and Einar Snekkenes (Eds.). Springer International Publishing, Cham, 389–407.

[19] C. Cremers, M. Horvat, S. Scott, and T. van der Merwe. 2016. Automated Analysis and Verification of TLS 1.3: 0-RTT, Resumption and Delayed Authentication. In *2016 IEEE Symposium on Security and Privacy (SP)*. 470–485. https://doi.org/10.1109/SP.2016.35

[20] Cas Cremers, Benjamin Kiesl, and Niklas Medinger. 2020. A Formal Analysis of {IEEE} 802.11's WPA2: Countering the Kracks Caused by Cracking the Counters. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*. 1–17.

[21] D. Dolev and A. Yao. 1983. On the security of public key protocols. *IEEE Transactions on Information Theory* 29, 2 (1983), 198–208.

[22] J. Dreier, L. Hirschi, S. Radomirovic, and R. Sasse. 2018. Automated Unbounded Verification of Stateful Cryptographic Protocols with Exclusive OR. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. 359–373.

[23] Jannik Dreier, Lucca Hirschi, Saša Radomirović, and Ralf Sasse. 2020. Verification of Stateful Cryptographic Protocols with Exclusive OR. *Journal of Computer Security* 28, 1 (Feb. 2020), 1–34. https://doi.org/10.3233/JCS-191358

[24] Shafi Goldwasser and Silvio Micali. 1984. Probabilistic encryption. *J. Comput. System Sci.* 28, 2 (1984), 270–299. https://doi.org/10.1016/0022-0000(84)90070-9

[25] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. 1988. A Digital Signature Scheme Secure against Adaptive Chosen-Message Attacks. *SIAM J. Comput.* 17, 2 (April 1988), 281–308. https://doi.org/10.1137/0217017

[26] Don Johnson, Alfred Menezes, and Scott Vanstone. 2001. The elliptic curve digital signature algorithm (ECDSA). *International journal of information security* 1, 1 (2001), 36–63.

[27] Haibat Khan and Keith M. Martin. 2020. A survey of subscription privacy on the 5G radio interface - The past, present and future. *Journal of Information Security and Applications* 53 (2020), 102537. https://doi.org/10.1016/j.jisa.2020.102537

[28] Neal Koblitz. 1987. Elliptic curve cryptosystems. *Mathematics of computation* 48, 177 (1987), 203–209.

[29] Robert Künnemann. 2015. Automated Backward Analysis of PKCS#11 v2.20. In *Principles of Security and Trust*, Riccardo Focardi and Andrew Myers (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 219–238.

[30] G. Lowe. 1997. A hierarchy of authentication specifications. In *Proceedings 10th Computer Security Foundations Workshop*. 31–43.

[31] Vadim Lyubashevsky and Daniele Micciancio. 2018. Asymptotically efficient lattice-based digital signatures. *Journal of Cryptology* 31, 3 (2018), 774–797.

[32] Simon Meier. 2013. *Advancing automated security protocol verification.* Ph. D. Dissertation. ETH Zurich, Zürich. https://doi.org/10.3929/ethz-a-009790675

[33] Victor S. Miller. 1986. Use of Elliptic Curves in Cryptography. In *Advances in Cryptology — CRYPTO '85 Proceedings*, Hugh C. Williams (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 417–426.

[34] Aleksi Peltonen, Ralf Sasse, and David Basin. 2021. A Comprehensive Formal Analysis of 5G Handover. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (Abu Dhabi, United Arab Emirates) *(WiSec '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3448300.3467823

[35] Adrian Perrig and J. D. Tygar. 2003. *TESLA Broadcast Authentication.* Springer US, Boston, MA, 29–53. https://doi.org/10.1007/978-1-4615-0229-6_3

[36] David Rupprecht, Adrian Dabrowski, Thorsten Holz, Edgar Weippl, and Christina Pöpper. 2018. On Security Research Towards Future Mobile Network Generations. *IEEE Communications Surveys Tutorials* 20, 3 (2018), 2518–2542. https://doi.org/10.1109/COMST.2018.2820728

[37] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. 2020. Call Me Maybe: Eavesdropping Encrypted LTE Calls With ReVoLTE. In *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, 73–88. https://www.usenix.org/conference/usenixsecurity20/presentation/rupprecht

[38] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. 2020. IMP4GT: IMPersonation Attacks in 4G NeTworks. In *27th Annual Network and Distributed System Security Symposium, NDSS 2020, San Diego, California, USA, February 23-26, 2020*. The Internet Society. https://www.ndss-symposium.org/ndss-paper/imp4gt-impersonation-attacks-in-4g-networks/

[39] Benedikt Schmidt. 2012. *Formal analysis of key exchange protocols and physical protocols.* Ph. D. Dissertation. ETH Zurich, Zürich. https://doi.org/10.3929/ethz-a-009898924

[40] Jeroen Veen and Sergei Boeke. 2020. Which is more important: online privacy or national security?: The Dutch position in the ongoing encryption debate. *Atlantisch Perspectief* 44, 4 (2020), 36–40. https://www.jstor.org/stable/48600570

[41] Yuchen Wang, Zhenfeng Zhang, and Yongquan Xie. 2021. Privacy-Preserving and Standard-Compatible AKA Protocol for 5G. In *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, 3595–3612. https://www.usenix.org/conference/usenixsecurity21/presentation/wang-yuchen

[42] Wei Wu, Jianying Zhou, Yang Xiang, and Li Xu. 2013. How to achieve non-repudiation of origin with privacy protection in cloud computing. *J. Comput. System Sci.* 79, 8 (2013), 1200–1213. https://doi.org/10.1016/j.jcss.2013.03.001

[43] Jingjing Zhang, Lin Yang, Weipeng Cao, and Qiang Wang. 2020. Formal Analysis of 5G EAP-TLS Authentication Protocol Using Proverif. *IEEE Access* 8 (2020), 23674–23688. https://doi.org/10.1109/ACCESS.2020.2969474

[44] Jinghao Zhao, Boyan Ding, Yunqi Guo, Zhaowei Tan, and Songwu Lu. 2021. SecureSIM: Rethinking Authentication and Access Control for SIM/ESIM. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New Orleans, Louisiana) *(MobiCom '21)*. Association for Computing Machinery, New York, NY, USA, 451–464. https://doi.org/10.1145/3447993.3483254

A VERIFICATION AND DISPUTE RESOLUTION ALGORITHM

Algorithm 1 LEA Verification and Dispute Resolution

procedure LEA VERIFICATION($\mathsf{Ctx}^{\mathsf{NF}}, \beta^{\mathsf{SUPI}}, \mathcal{T}$)
  Parse $\mathsf{Ctx}^{\mathsf{NF}}$ as $\{\mathsf{SUPI}, \beta^{\mathsf{SUPI}}, \mathsf{RES}^*, \mathsf{k}_{\mathsf{ECIES}}, \sigma^{\mathsf{NF}}, \mathsf{t}, \mathsf{gNB}_{\mathsf{id}}\}$
  Parse β^{SUPI} as $\{\mathsf{t}_{\mathsf{res}}, \sigma^{\mathsf{setup}}, \mathsf{pk(NF)}\}$
  $v_1 \leftarrow \mathsf{Verify}_{\mathsf{ECDSA}}(\sigma^{\mathsf{setup}}, \{\mathsf{t}_{\mathsf{res}}, \mathsf{SUPI}, \mathsf{pk(NF)}\}, \mathsf{pk}(\mathsf{HN}_{\mathsf{k}}))$
  $v_2 \leftarrow \mathsf{Verify}_{\mathsf{ECDSA}}(\sigma^{\mathsf{NF}}, \{\mathsf{RES}^*, \mathsf{gNB}_{\mathsf{id}}, \mathsf{k}_{\mathsf{ECIES}}, \mathsf{t}\}, \mathsf{pk(NF)})$
  Raise exception if v_1 or v_2 fails
  for packet $\in \mathcal{T}$ **do**
    Decrypt and verify packet signature
  end for
end procedure
procedure DISPUTE RESOLUTION($\beta_{\mathsf{r}}^{\mathsf{SUPI}}, \beta^{\mathsf{SUPI}}$) ▷ The received binding from the user
  if $\beta_{\mathsf{r}}^{\mathsf{SUPI}} = \beta^{\mathsf{SUPI}}$ **then**
    Reject dispute
  end if
  Parse $\beta_{\mathsf{r}}^{\mathsf{SUPI}}$ as $\{\mathsf{t}_{\mathsf{r}}, \sigma_{\mathsf{r}}^{\mathsf{setup}}, \mathsf{pk(NF)}_{\mathsf{r}}\}$
  $v_3 \leftarrow \mathsf{Verify}_{\mathsf{ECDSA}}(\sigma_{\mathsf{r}}^{\mathsf{setup}}, \{\mathsf{t}_{\mathsf{r}}, \mathsf{SUPI}, \mathsf{pk(NF)}_{\mathsf{r}}\}, \mathsf{pk}(\mathsf{HN}_{\mathsf{k}}))$
  accept dispute if v_3 = verified, reject otherwise
end procedure