# Component-Based Formal Analysis of 5G-AKA: Channel Assumptions and Session Confusion

Cas Cremers
CISPA Helmholtz Center for Information Security, Germany
cremers@cispa.saarland

Martin Dehnel-Wild
Department of Computer Science, University of Oxford
martin@dehnelwild.co.uk

***Abstract*—The 5G mobile telephony standards are nearing completion; upon adoption these will be used by billions across the globe. Ensuring the security of 5G communication is of the utmost importance, building trust in a critical component of everyday life and national infrastructure.**

**We perform fine-grained formal analysis of 5G's main authentication and key agreement protocol (AKA), and provide the first models to explicitly consider all parties defined by the protocol specification. Our analysis reveals that the security of 5G-AKA critically relies on unstated assumptions on the inner workings of the underlying channels. In practice this means that following the 5G-AKA specification, a provider can easily and 'correctly' implement the standard insecurely, leaving the protocol vulnerable to a security-critical race condition. We provide the first models and analysis considering component and channel compromise in 5G, whose results further demonstrate the fragility and subtle trust assumptions of the 5G-AKA protocol.**

**We propose formally verified fixes to the encountered issues, and have worked with 3GPP to ensure these fixes are adopted.**

## I. Introduction

The 5th Generation (5G) mobile networks and telecommunications standard is currently under development, and are nearly finalised. A crucial building block in this standard is the "5G Authentication and Key Agreement" (5G-AKA) protocol. This protocol is developed by 3GPP, and is an evolution of the AKA variants used in 3G and 4G, and is used to authenticate and establish keys between the involved parties. These parties include the subscribers, the networks within close range (referred to as Serving Networks), and the subscribers' carriers (referred to as Home Networks). The security of all 5G communication therefore crucially relies on 5G-AKA.

Traditionally, security protocols standards were not developed in tandem with rigorous security analysis, leading to many vulnerabilities being found after deployment. More recently, there has been a positive trend in which rigorous scientific analysis has been part of the development process; most notably IETF's TLS 1.3 protocol [25], which has benefited from being developed in tandem with a range of analysis methods [24]. Given the extremely wide deployment of 5G in the near future, it seems prudent to perform state-of-the-art analysis for this standard as well.

**Methodology** Our work aims to provide rigorous formal analysis and to improve the security of the 5G-AKA standard. Our approach uses formal symbolic modeling with the Tamarin prover [21], which has been successfully used during the development of major protocols such as TLS 1.3 [16].

Several aspects of the 5G-AKA protocol complicate formal analysis. The first is the sheer complexity of the specification documentation, which spans hundreds of pages. The second is the complexity of the protocol, which involves four parties, depends on sequence counters for its security, and complex channel assumptions. The third is the informal nature of the security requirements, which mean the modeler has to make complex assumptions on the basis of the possible use cases.

We closely model the 5G-AKA specification: in particular, we explicitly model all four main parties in the specification, in which Home Networks include a separate component for credential storage that may be implemented in e.g., a hardware security module. We explicitly model possible assumptions on the channels connecting these four parties. We then analyse the resulting system model with respect to a range of threat models, including compromised components and channels.

**Related work** Previous versions of AKA have been analysed after deployment, and typically used simplified models.

The original 3G AKA protocol was manually analysed in 1999 [1], using TLA and BAN Logic. Both these methods consider abstract models that are much coarser than modern techniques, only considering very weak threat models.

In 2012, *Tsay and Mjølsnes* presented a vulnerability [26] on the older UMTS-AKA and LTE-AKA protocols. This attack allows for a violation of authentication properties based upon session confusion. The attack in [26] was found indirectly through use of CryptoVerif [13]. We will return to the relation between this work and ours in VIII.

*Køien* [19] proposes improvements to 4G's AKA, achieving full mutual online authentication. Previous AKA protocols delegate authority from the home network to a serving network by forwarding up to 5 irrevocable authentication vectors, allowing the home network to be offline during the challenge-response phase of the protocol: this requires too high a level of trust between network operators, hence they propose a modified protocol which is fully online for all parties. 5G-AKA now only forwards one authentication vector at a time.

*Arapinis et al.* [9] analyse 3G's authentication protocols, discovering attacks against the privacy and linkability of subscriber

Network and Distributed Systems Security (NDSS) Symposium 2019
24-27 February 2019, San Diego, CA, USA
ISBN 1-891562-55-X
https://dx.doi.org/10.14722/ndss.2019.23394
www.ndss-symposium.org

identities. This modelling and analysis uses ProVerif, formally verifying the proposed solutions achieving unlinkability and anonymity. *O'Hanlon et al.* [23] consider the interaction between 4G's authentication protocols and operator-backed WiFi services; they detail how the interaction between these can enable serious privacy violations, as well as their experiences reporting the discovered issues to the relevant stakeholders. *Hussain et al.* [18] combine symbolic model checking with cryptographic protocol verification for 4G's attach, detach, and paging procedures, discovering 10 new attacks, including an authentication relay attack, allowing an adversary to spoof the location of a legitimate user.

In recent concurrent work [11], *Basin et al.* use a similar approach to ours to analyse 5G-AKA, but focus on different aspects. Basin et al. model and analyse a 3-party interpretation of the 5G-AKA protocol and its security properties, merging two major components (the AUSF and ARPF) to form a single 'Home Network' entity, similar to previous AKA versions. They discovered the authentication issues created by lack of integrity protection on the serving network's ID. In contrast, we model all four parties as defined in the protocol's specification, and consider a range of compromise models. In this sense, our results are orthogonal: we focus on a more fine-grained model with a large range of compromise models, while Basin et al.'s models put more focus on detailed analysis of the counter re-synchronisation method and the privacy guarantees of 5G-AKA. They additionally model and analyse the 'Elliptic Curve Integrated Encryption Scheme' which 5G-AKA uses for SUPI concealment to ensure subscriber privacy. Beyond simple confirmation of these results, we do not consider privacy-specific properties within 5G. While in theory one would like to model all aspects at once in a single model, the analysis of the individual models is already very complex using current technology: analysing a combined model does not yet seem feasible. In practice, this means that the individual models cover partially overlapping, but different classes of attacks. For example, our model does consider attacks on privacy, whereas the attack we detail later is not visible in the model from [11].

*Contributions:* We provide three main contributions. First, we propose a fine-grained formal model of the 5G-AKA standard that enables a detailed view of the interaction between the various security-critical components.

Second, we perform symbolic analysis of this model with respect to a range of threat models. Our analysis confirms many already discovered issues and subtle assumptions and requirements to achieve security in 5G-AKA.

Third, our analysis reveals that the security of 5G-AKA critically relies on unstated assumptions on the inner workings of the underlying channels. In particular, the automated analysis discovers an attack that exploits a potential race condition. We additionally show that solving the race condition for the honest case does not necessarily prevent the attack. In practice this means that solely based on the 5G-AKA specification, a provider can implement the standard insecurely. We propose fixes and prove that they prevent the attack. We have reported our findings to the 3GPP SA3 working group and are working with a major provider to integrate a fix to the standard.

The complete TAMARIN models can be found at [15].

*Overview:* We structure our work in three main parts. In the first part, we describe the protocol (Section II), the threat model implied by the standard (Section III), security properties (Section IV), and its formalisation (Section V and VI.)

In the second part, we formalise and model a basic threat model, and perform analysis in Sections VII to IX. We then consider the modelling, analysis, results, and consequences of stronger threat models that involve channel and component compromise in Section X.

Finally, in the third part, we discuss the potential implications of these results, we discuss our interaction with the 3GPP working groups and upcoming changes to the standard (TS 33.501 [5]) in Section XI, before concluding in Section XII.

## II. THE 5G-AKA PROTOCOL

The 5G-AKA protocol is the "Authentication and Key Agreement" protocol within the newly proposed 5G standard, and therefore is the core building block for the security guarantees of the standard. 5G-AKA has evolved from the EPS-AKA protocol as used by 4G/LTE [7]. The 5G-AKA protocol is specified within §6.1.3.2 of 3GPP Technical Specification 33.501 [5]; here we model version v0.7.0.

We distinguish between the "home network", e.g., the network that the user signed up with, and the "serving network", which is the actual network that the phone connects to. While the home network can be the same as the serving network, they are different in a roaming scenario.

The 5G-AKA protocol establishes authentication and key agreement between a mobile device, a serving network, and the device's home network. The standard additionally specifies a credential repository, which resides within the home network. The protocol is therefore specified as a sequence of communications between four roles:

- **UE**: the 'User Equipment'. This can be e.g., mobile phones or USB dongles. Each UE is uniquely identified by its SUbscription Permanent Identifier (**SUPI**). In 5G, the SUPI replaces the 'IMSI' in pre-5G standards.
- **SEAF**: the 'Security Anchor Function'. In the roaming context, this is within the serving network, e.g., the network that the phone connects to in a remote location.
- **AUSF**: the 'Authentication Server Function'. This falls within the home network, e.g., the network of the phone's service provider.
- **ARPF**: the 'Authentication credential Repository and Processing Function'. In the home network. Often resides in a secure location, e.g., a Hardware Security Module.

See Figure 1 for a diagram illustrating the parties and channels in the 5G-AKA protocol; in this example, a mobile phone user

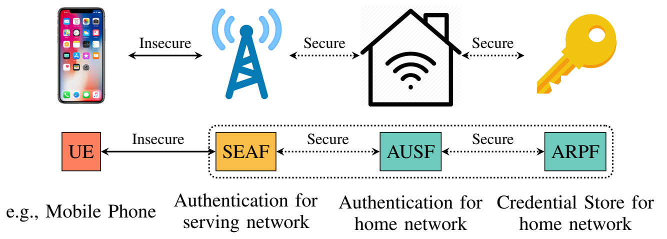
Insecure
Secure
Secure
UE
Insecure
SEAF
Secure
AUSF
Secure
ARPF
e.g., Mobile Phone
Authentication for serving network
Authentication for home network
Credential Store for home network

Fig. 1. Parties and channels involved in the 5G-AKA protocol. The dashed box represents the 5G core network; dashed channels are considered 'secure'.

is roaming, communicating back to their home network.

The UE and ARPF alone share the user's long-term secret symmetric key, **K**. The SUPI should never be exposed publicly: a 'SUbscription Concealed Identifier' or **SUCI** is used to achieve this. The **SIDF**, or the Subscriber Identity De-concealing Function is used to decrypt a SUCI value into a SUPI: this functionality is co-located with the ARPF.

At the end of a successful run of the protocol, all parties can derive *and agree upon* an 'anchor key' $\mathbf{K_{SEAF}}$, from which session keys for communication between the mobile device and base station(s) within the local network are derived. After the protocol, all communication between UE and the core network uses this key or keys derived from it. The secrecy of the key $K_{SEAF}$ is therefore crucial to ensure the security of subsequent operations and communications.

*Normal execution of the 5G-AKA protocol*

We give an overview of the core of the 5G-AKA protocol execution, as described in [5, §6.1.3.2]. We give the corresponding message sequence chart in Figure 2.

1) **SUCI, HN:** The UE sends its encrypted 'concealed ID' (SUCI, encrypted with a variant of the Elliptic Curve Integrated Encryption Scheme) and the name of its home network to a SEAF in the serving network. In the case that the SUPI is not concealed, it just sends the SUPI value unencrypted.
2) **5G-AIR:** The SEAF sends a `5G-AIR` message containing the previous message and the name of the serving network to an AUSF in the relevant home network.
3) **Auth-Info Request:** The AUSF then transmits this information in an '`Auth-Info Request`' message to the home network's ARPF.
4) **Auth-Info Response:** The ARPF **a)** Requests de-concealment of the SUCI into its respective SUPI from the Subscriber Identity De-concealing Function, or SIDF. **b)** The ARPF retrieves the relevant K for this user, and **c)** Generates a 128-bit random number 'RAND', a single AUTN value derived from RAND and the user's long-term key K, an 'Expected Response' value (XRES*), and a session key for the AUSF, $K_{AUSF}$. These are sent to the AUSF in an '`Auth-Info Response`' message. The 'Expected Response' value is a hash of the derived keys, RAND, and SNID (i.e., the ID of the serving network being used); possession of it enables other parties (which may not know K) to verify that the user responded correctly.
5) **5G-AIA:** The AUSF sends a `5G-AIA` message containing AUTN, a hash of the 'Expected Response' (i.e., HXRES*), the new 'anchor key' $K_{SEAF}$ derived from $K_{AUSF}$, and the SUPI of the intended recipient.
6) **Auth-Req:** The SEAF sends RAND and AUTN to the UE in an `Auth-Req` message.
7) **Auth-Resp:** The UE proves its identity, and implicitly, ownership of K, by responding to the SEAF with RES* (i.e., the 'Response') within an `Auth-Resp` message; the UE can now calculate the keys $K_{AUSF}$ and $K_{SEAF}$.
8) **5G-AC and 5G-ACA:** The SEAF calculates the hash of RES* (i.e., HRES*) received from the UE and checks it matches the hash of the '*Expected* Response', HXRES* from the AUSF. If so, the SEAF considers the authentication successful, and sends an Authentication Confirmation (`5G-AC`) message containing the user's SUPI, the serving network's ID, and the response, to the AUSF. The AUSF acknowledges this, replying with a `5G-ACA` message.

5G-AKA is a natural evolution from previous generation AKA protocols, and as such we find similarities between components. Of note for this research, the pair of 5G components AUSF & ARPF appear to have similar functionality to the pair of 2G/3G/4G components HLR & AuC (denoting respectively the 'Home Location Register' and 'Authentication Centre').

In the 2G-, 3G-, and 4G-AKA protocols, the specifications consider the AuC and HLR as a single component, referred to as the "HSS" or Home Subscriber Server. For example, the 4G-AKA specification in TS 33.401 [7] does not even mention the HLR (cf. AUSF). Previous research therefore only models three components instead of four, with good reason: in 2G/3G/4G, the AuC does not participate in the AKA protocol directly. The AuC generates various keys on demand but is not a named party in the protocol's message sequence flow.

In 5G both the AUSF and ARPF have separate, formally encoded roles to play in the protocol specification, as explicitly specified in the 5G-AKA specification in TS 33.501 [5]. We therefore consider and model them as distinct components.

## III. Threat Model

The 5G-AKA documentation does not specify an explicit threat model. Section 5 of TS 33.501 v0.7.0 [5], "Security requirements and features" gives a mixture of threat models and desired security properties from the perspectives of the involved components, and we attempt to extract the most important points relating to the threat model here. We refer to this threat model as $\mathcal{A}_{Standard}$. We return to the required security properties in Section IV. For transparency, we quote the original documentation where possible.

### A. Channel threat model

5G-AKA uses three network channels, as in Figure 1:

1) UE ↔ SEAF
2) SEAF ↔ AUSF
3) AUSF ↔ ARPF

The communications between SEAF, AUSF, and ARPF are within the "5G Core Network" and are specified to use "e2e core network interconnection" channels. In Figure 3 we quote the requirements of these channels from TS 33.501, which suggest that SEAF ↔ AUSF and AUSF ↔ ARPF form a type of secure channel. The required properties do not explicitly require or guarantee delivery of messages, nor of ordering of the receipt of messages. We believe these properties are analogous (or very close) to setting up and maintaining long-term IPSec, (D)TLS, or Diameter sessions over these channels, between the named parties. We return to the subtleties regarding the precise assumptions and formal modelling later in Section VII-B.

The standard does not specify any assumed security for the channel between UE and SEAF, as the signal is over the air. In some sense, providing security here is part of what 5G-AKA aims to provide. We assume the channel between the UE and SEAF is insecure, and model it using a classical Dolev-Yao network adversary model.

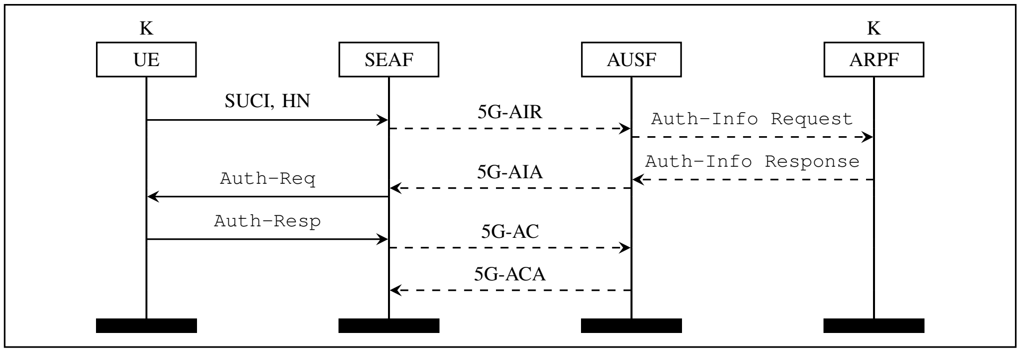

Fig. 2. The normal flow of the 5G-AKA Protocol. Only the UE and ARPF know the user's long-term key, K. Dashed lines indicate secure channels.

> 5.7.4: Requirements for e2e core network interconnection security
>
> A solution for e2e core network interconnection security shall satisfy the following requirements.
>
> - The solution shall provide confidentiality and/or integrity end-to-end between source and destination network for specific message elements identified in this specification. [...]
> - The destination network shall be able to determine the authenticity of the source network that sent the specific message elements protected [...]
> - The solution should be using standard security protocols.
> - The solution shall cover prevention of replay attacks.

Fig. 3. Requirements for e2e core network security (from [5] p. 21)

### B. Component threat model

TS 33.501 v0.7.0 [5] does not describe whether it considers compromise of components within the system as part of its threat model. We assume that compromise of any core network component (SEAF, AUSF, or ARPF) is not allowed in the basic threat model, $\mathcal{A}_{Standard}$. The standard describes (Figure 4) the protections required for the long-term key K within the USIM, so we assume an adversary cannot compromise an honest user's key K. We do however assume that a persistent and capable adversary would be able to compromise the long-term key(s) of *other* USIMs, e.g., ones in its long-term possession. *Separate from our main analysis* we consider a stronger threat model, where compromise of components and secure channels are allowed in Section X: $\mathcal{A}_{Stronger}$.

## IV. Required Security Properties

TS 33.501 v0.7.0 [5] details security requirements on the elements of the 5G ecosystem. We now detail the requirements directly affecting 5G-AKA, and the security properties the standard states or implies 5G should uphold. TS 33.501 contains the text describing "security requirements", considering confidentiality and integrity requirements; we cite this in Figure 6. Section 5 describes these requirements on Authentication and Authorization; we include this in Figure 7.

### A. Secrecy

TS 33.501 v0.7.0 [5] does not explicitly state a requirement for the secrecy of the session key $K_{SEAF}$ (the "anchor key"); possession of this key grants the bearer access to a network on behalf of the UE which derived the key; Figure 5 alludes strongly to the importance of the $K_{SEAF}$, and its cryptographic parent, the $K_{AUSF}$. We consider session key secrecy to be one

> 5.1.4 Secure storage and processing of subscription credentials
>
> The following requirements apply for the storage and processing of the subscription credentials used to access the 5G network:
>
> - The subscription credential(s) shall be integrity protected within the NG-UE using a tamper resistant secure hardware component.
> - The long-term key(s) of the subscription credential(s) (e.g., K in EPS AKA) shall be confidentiality protected within the NG-UE using a tamper resistant secure hardware component.
> - The long-term key(s) of the subscription credential(s) shall never be available in the clear outside of the tamper resistant secure hardware component. [. . . ]

Fig. 4. Secure storage and processing of credentials (from [5] p. 16)

> 6.1 Primary authentication and key agreement
>
> The purpose of the primary authentication and key agreement procedures is to enable mutual authentication between the UE and the network and provide keying material that can be used between the UE and network in subsequent security procedures. The keying material generated by the primary authentication and key agreement procedure results in an anchor key called the $K_{SEAF}$ provided by the AUSF of the home network to the SEAF of the serving network.
>
> Keys for more than one security context can be derived from the $K_{SEAF}$ without the need of a new authentication run. A concrete example of this is that an authentication run over a 3GPP access network can also provide keys to establish security between the UE and a N3IWF used in untrusted non-3GPP access.
>
> The authentication run also results in an intermediate key called the $K_{AUSF}$. The $K_{AUSF}$ may be left at the AUSF based on the home operator's policy on using such key.

Fig. 5. Primary authentication and key agreement (from [5] p. 25)

> 5 Security requirements and features
>
> **5.1.2 User data and signalling data confidentiality**
>
> 5.1.2.1 Requirements on Support and Usage of Ciphering
>
> - The UE shall support ciphering of user data between the UE and the gNB.
> - The UE shall support ciphering of RRC and NAS-signalling. [. . . ]
> - Confidentiality protection of the user data between the UE and the gNB is optional to use.
> - Confidentiality protection of the RRC-signalling, and NAS-signalling is optional to use.
> - Confidentiality protection should be used whenever regulations permit.
>
> **5.1.3 User data and signalling data integrity**
>
> 5.1.3.1 Requirements on support and usage of integrity protection
>
> - The UE shall support integrity protection and replay protection of user data between the UE and the gNB.
> - The UE shall support integrity protection and replay protection of RRC and NAS-signalling. [. . . ]
> - Integrity protection of the RRC-signalling, and NAS-signalling is mandatory to use, except in the following cases: [. . . ]

Fig. 6. Security requirements and features (from [5] p. 15)

of the primary goals of 5G-AKA, goal, even if unstated in the specification. We therefore interpret the requirements from Figure 4 as the following **key secrecy** properties:

S1. The adversary must not be able to learn the long-term secret key K of an honest subscriber (stored within the UE/USIM).
S2. The adversary must not be able to learn an "anchor key" $K_{SEAF}$ for an honest subscriber derived by 5G-AKA, or its cryptographic parent, $K_{AUSF}$.

### B. *Authentication and agreement*

> 5.10 Authentication and Authorization
>
> The 5G system shall satisfy the following requirements.
> **Subscription authentication:** The serving network shall authenticate the Subscription Permanent Identifier (SUPI) in the process of authentication and key agreement between UE and network. [...]
> **Serving network authentication:** The UE shall authenticate the serving network identifier through implicit key authentication. The meaning of 'implicit key authentication' here is that authentication is provided through the successful use of keys resulting from authentication and key agreement in subsequent procedures. [...]
> **UE authorization:** The serving network shall authorize the UE through the subscription profile obtained from the home network. UE authorization is based on the authenticated SUPI. [...]
> **Serving network authorization:**
> *Serving network authorization by the home network:* Assurance shall be provided to the UE that it is connected to a serving network that is authorized by the home network to provide services to the UE. This authorization is 'implicit' in the sense that it is implied by a successful authentication and key agreement run. [...]
> **Access network authorization:** Assurance shall be provided to the UE that it is connected to an access network that is authorized by the serving network to provide services to the UE. This authorization is 'implicit' in the sense that it is implied by a successful establishment of access network security. This access network authorization applies to all types of access networks.

Fig. 7. Authentication and Authorization properties required by TS 33.501 (from [5] p. 23).

We interpret the requirements from TS 33.501, notably Figure 7, as the following **agreement** properties:

A1. The serving network and UE must agree on the identity of the UE.
A2. The UE and serving network must agree on the identity of the serving network.
A3. The home network and serving network must agree on the identity of the UE (and upon agreement, the home network confirms that the UE is a legitimate subscriber).
A4. The UE and home network must agree on the identity of the home network.
A5. The UE and home network must agree on the identity of the serving network (and this agreement implies that the serving network is authorised by the home network).
A6. The UE, serving network, and home network must agree on the anchor key, $K_{SEAF}$.
A7. The anchor key $K_{SEAF}$ must not be replayable, i.e., the UE, home network, and serving network agree that the $K_{SEAF}$ has only ever been accepted by one session.[1]

We describe how we interpret, model, and analyse these informally defined security requirements as more formal secrecy and authentication properties in Sections VI-A and VI-B.

## V. Formal Model of 5G-AKA in Tamarin

We formally model the four-party 5G-AKA protocol v0.7.0 [5] in the Tamarin Prover.[2] The explicit specification of four parties within 5G-AKA is a change from UMTS-AKA (3G) and LTE-AKA (4G), which describe three major network components, rather than four. As we will see later, this opens up new attack possibilities that are not covered if one models AUSF and ARPF as a single entity, as done in [11].

Our modelling and analysis of 5G-AKA takes advantage of the Tamarin security protocol verification tool [21]. We give an overview of the modelling assumptions we made (in the context of 5G), and example 5G-AKA rules in its syntax.

### A. *Symbolic modelling*

We use symbolic analysis, which means that instead of concrete bitstrings, we consider abstract terms. For example, the hash of a term $x$ is represented as the term $h(x)$. The adversary can symbolically manipulate terms, e.g., decrypting terms for which it knows the key. This corresponds to the black-box model of cryptography often called *perfect cryptography*. Depending on the threat model the adversary can eavesdrop, insert, or block messages on channels, or may learn secret information of any component or party it can compromise.

### B. *Modelling choices*

*Counters, 'SQN':* The 5G-AKA protocol makes use of a counter or sequence number, SQN. TS 33.501 [5] refers to TS 33.102 [4, §6.3.2] for the definition and behaviour of this term. The standard explicitly acknowledges that counters wrapping around could lead to repetition of a CK/IK[3] key-pair, and gives a method for protecting against this ("informative Annex C.2" of [4]). We model counters as strictly monotonically increasing, with no possibility of wrapping around.

We do not consider deltas, or allowed increases between maximum previously seen counter values; we permit all SQN values which are strictly greater than the current maximum value. In this way, we are slightly more permissive than many implementations of the standard may be; we do not believe this affects our results. UEs and network operators are not required to implement the given counter-measure; we discuss the implications (or lack thereof) counters have overall on our discovered attacks and further analysis in Section VIII-A.

### C. *Separation of components*

5G-AKA is an evolution of the EPS-AKA protocol from LTE/4G [7]; changes include the inclusion of concealed identities or SUCIs, and the addition of the `5G-AC` and `5G-ACA` messages. Another important difference is the number components formally described by the protocol standard. In LTE/4G, the 'Authentication and key agreement' protocol

[1] Replay protection for all data is required and indicated at multiple points, so we believe this is a reasonable goal for $K_{SEAF}$ as well.

[2] These models have built upon and significantly diverged from initial models of 5G-AKA **v0.3.0** in the *three*-party setting, eventually leading to independent concurrent work: [11].

[3] CK and IK are the (symmetric) Confidentiality and Integrity Keys respectively, both generated from $K_{SEAF}$, by both the UE and SEAF. A repeated SQN would not lead to repeated CK/IK values *directly* as they are derived solely from K and RAND, but if a protocol run can be replayed with a previously seen RAND value (and the ARPF and UE will accept it), then the resultant CK/IK will be the same.

section of TS 33.401 [7, §6.1.1] describes just *three* components: the UE,[4] and the "HSS", or Home Subscriber Server. In 5G we have *four* components involved in the 5G-AKA protocol, i.e., the UE, SEAF, AUSF, and ARPF. The SEAF and MME are broadly analogous in functionality; the HSS's functionality is split between the AUSF and ARPF.

Other research modelling 5G-AKA only considers three major components; it is therefore worth discussing in more detail some of the reasons behind our choice to model four components. Because channels between AUSFs and ARPFs are completely internal to a telecommunications company's network and work over different mediums, we cannot assume that they will be implemented similarly. As further context, we have confirmation from a large telecommunications provider that there are major differences between internal network security, controls, regulation, and network implementation compared to cross-network boundaries – in many instances we believe there is often little to no internal network encryption at all, making this scenario materially different to the one presented in [11].

The standard allows that other channels are implemented by session bound local instances (preventing the attack presented by [11]) while the AUSF ↔ ARPF channel could be implemented by a long-running connection that doesn't provide a similar binding. As we will show later, there is a potential attack based on underspecified assumptions on the AUSF ↔ ARPF channel. This attack would not be prevented by following the recommendations from [11] on the channels that they considered in their work, since they did not model AUSF ↔ ARPF.

### D. Modelling limitations

We do not model the counter 'resync' mode of 5G-AKA: this is addressed by Basin et al. in [11]. We do not consider the later derivation of keys within the key hierarchy after 5G-AKA has finished. We also do not consider distinctions between e.g., the User Plane, Control Plane, Radio Resource Control, Access Stratum, and Non-Access Stratum, except where these make a difference to the 5G-AKA protocol's behaviour. We do not model the EAP-AKA′ protocol (described in [5, §6.1.3.1] and RFC 5448 [10]) as it and the very closely related EAP-AKA protocol have been studied in depth elsewhere [8], [14], [22]. Integrating a model of EAP-AKA′ into our models of 5G-AKA would be useful future work: analysing their composition would be useful and non-trivial, as EAP-AKA′ also makes use of the same long-term key K.

## VI. Formalisation of Security Properties

Having modelled 5G-AKA in Tamarin's multiset rewriting rules, we now model the range of desired security properties as (temporal) first-order logic formulae. These formulae are then evaluated over runs of the protocol generated by the model.

As detailed in Section IV, and as cited in Figures 5 and 7, TS 33.501 requires the informally described security properties S1, S2, A1–A7. These lead us to believe that 5G-AKA should uphold the more traditionally defined properties of session key secrecy (which here implies long-term key secrecy), non-injective agreement on the parties involved, and injective agreement on the session key, $K_{SEAF}$.

We consider session key secrecy, long-term key secrecy, non-injective and injective agreement on a variety of terms.

### A. Secrecy properties

We consider both session key secrecy (of $K_{SEAF}$ and $K_{AUSF}$) and long-term key secrecy (of K) for 5G-AKA.

The session key secrecy lemmas state that for all traces where there was not a long-term key reveal action by the adversary for the UE/SUPI in question, the adversary never learns (or derives) the resultant session key, $K_{SEAF}$ (or in the case of the ARPF, $K_{AUSF}$). As defined in TS 33.501 Annex A.6, $K_{SEAF} = \text{KDFA}(K_{AUSF}, \text{SNID})$; this is calculated at the AUSF. If the adversary is in possession of $K_{AUSF}$, they can derive $K_{SEAF}$, but not the other way round. The ARPF can clearly derive $K_{SEAF}$, but from the point of view of the ARPF we consider the secrecy of the $K_{AUSF}$ to pin-point any compromise precisely: adversarial knowledge of $K_{AUSF}$ implies knowledge of $K_{SEAF}$, but the reverse is not the case.

We consider session key secrecy from the point of view of each of the four parties, as this captures a broader range of properties than just secrecy for the UE. We do not just consider the adversary being able to violate the secrecy of what is ostensibly the correct session key, but also the situation where an adversary can trick a party into accepting an incorrect session key which the adversary knows or can derive. This latter property also involves a violation of authentication, but we discuss those properties in the next section.

The session key secrecy properties in the 5G-AKA models are of the following form:

```
lemma secrecy_UE:
"All UE t #i. Secret(<'UE', UE>, t) @ #i
    & not(Ex SUPI HN #r. RevealKforSUPI(SUPI) @ #r
          & Honest(<SUPI,HN>) @ #i )
    ==> not (Ex #j. K(t) @ #j )"
```

Intuitively, for all traces such that a `Secret` action fact occurs at the UE at time point `#i` for term `t` (the session key), and there is no adversary key-reveal action for the same SUPI as in use at point `#i`, then there does not exist a time point `#j` such that the adversary learns or can derive that same term `t`. We discuss use of the '`RevealKforSUPI(...)`' action fact in more depth in Section VII. We consider session key secrecy properties of this form for all four of the protocol's parties.

The long-term key secrecy lemma roughly says: for all long-term keys (each bound to a specific `SUPI`), where that key `Ki` (= K) was not revealed directly by the adversary, there is no time point such that the adversary learns the long-term key. This is modelled as follows:

```
lemma secrecy_Ki:
    " All SUPI Ki #i. LongTermKey(SUPI,Ki) @ #i
      & not(Ex #r. RevealKforSUPI(SUPI) @ #r)
        ==> not (Ex #j. K(Ki) @ #j)"
```

We believe these lemmas model the informal properties S1 and S2. We provide full results for session key secrecy and long-term key secrecy in Section VIII.

### B. Authentication properties

As 5G-AKA considers four parties (each with different roles), it is not sufficient just to consider traditional two-party

[4] Joint with the USIM; we model them as part of the same entity.

authentication properties defined between the components in possession of the long-term secret key, i.e., the UE and ARPF.

The authentication properties described below systematically cover the range of pairwise authentication properties which we believe the 5G-AKA protocol could be expected to provide. For properties in the serving and home networks, we do not expect 5G-AKA to create confidentiality, integrity, or authentication guarantees between 5G core network parties. The standard describes properties which must be guaranteed by the underlying network connections, and we explore this further in Sections III-A and VII-B. To analyse authentication properties, we place Action Facts within protocol rules. We consider pairwise agreement properties from the points of view of each of the UE, serving network, and home network (‘Commit’ Action Facts). These properties are then considered in relation to each one of the four parties (UE, SEAF, AUSF, and ARPF), who generate the relevant ‘Running’ Action Facts. We do not consider agreement properties *from* the point of view of the ARPF on its own, because: firstly, the ARPF’s only role is to initiate the cryptographic section of the protocol (generating RAND, AUTN, XRES*, and $K_{AUSF}$); none of the messages **before** the Auth-Info Request message involve any keys, randomness, state, or other cryptographic elements; secondly, no messages are returned to the ARPF.

We can combine the results from these pairwise properties (agreeing on single terms) together to achieve the more traditional properties of e.g., non-injective or injective agreement over the identities involved in the protocol run *and* a term such as the session key, as described in [20]. Performing the analysis systematically in this manner helps us to pin-point precisely which terms (if any) cause any violations of agreement.

For example, if all three individual properties “UE and ARPF agree on the identity of the ARPF”, “UE and ARPF agree on the identity of the UE”, and “UE and ARPF agree on $K_{SEAF}$” hold true, this would imply the stronger, traditional property from the point of view of the UE of non-injective agreement between the UE and ARPF on the term $K_{SEAF}$.

*UE:* From the informal authentication properties defined in Section IV, we believe the properties directly concerning the point of view of the UE are A1, A2, A4–A7. Achieving all of these properties would be similar to achieving the traditional property of injective agreement (as described in [20]) on the identities of the UE, serving network, and home network, in combination with the term $K_{SEAF}$. We explore the full range of properties from the point of view of the UE, considering agreement with the SEAF, AUSF, and ARPF on the identities of the parties, and the ‘data’ term, e.g., $K_{SEAF}$.

Modelled formally, these properties follow the pattern illustrated in the following example lemma:

```
lemma agreement_UE_SEAF_ARPF:
  "All a b c d t #i.
      (Commit(a,<a,b,c,d>,t,<'UE','K_SEAF'>) @ #i
  & not(Ex #r.
      RevealKforSUPI(a)@r & Honest(<a,d>)@ #i))
  ==>(Ex a2 b2 c2 t2 #j .
      Running(b2,<a2,b2,c2,d>,t2, <'SEAF','K_SEAF'>) @ #j )"
```

This example models agreement from the UE’s point of view with a SEAF on the identity of the ARPF.

In more detail, this says: For all traces such that there was a ‘Commit(...)’ Action Fact at the UE, where the UE believes the parties involved in the protocol are a, b, c, and d (instantiating as the unique IDs of the UE, SEAF, AUSF, and ARPF respectively), and the UE believes that the session key $K_{SEAF}$ is term t, and there was not an adversary key reveal against the UE’s long-term key K, then there **must** exist at least one ‘Running(...)’ Action Fact from a SEAF which agrees with the UE on the identity of the ARPF. Note that this specific lemma does not require agreement on *any other terms:* e.g., the UE and SEAF involved in the specific Commit and Running Action Facts could completely disagree on the identity of the AUSF, or even on the identities of the two directly involved parties, the UE and SEAF. Proving this property true demonstrates **non-injective agreement** on just the named term, in this case, the ARPF.

We then also consider **injectivity:** achieving injective agreement requires agreement on the same terms (and/or parties) as before, but now also requires that there must be precisely one Commit(...) Action Fact with the specified term. As all of the identities of the parties involved may reasonably be used in repeated protocols, the only terms where we can hope to achieve injectivity are the ‘data’ terms, e.g., $K_{SEAF}$ and $K_{AUSF}$.

Injective agreement lemmas are modelled similarly, but now additionally require that there must not exist another Commit(...) Action Fact agreeing on the same term t at a different time point #k, i.e., such that #i and #k are not the same event.

*Serving Network (SEAF):* From the informal authentication properties defined in Section IV, we believe the properties directly concerning the point of view of the serving network are A1–A3, A6, and A7. The serving network shares privileged, authentic, and non-replayable access to the 5G core network through which it can communicate with the home network. Before the protocol run, the SEAF does not share any secrets relevant to the 5G-AKA protocol with any other parties, nor does it generate any randomness, or maintain state beyond each run of the protocol. We might ordinarily expect that the strongest achievable authentication property with a confidential channel would be non-injective agreement; however, as the secure channel between the AUSF and SEAF is explicitly non-replayable, we can potentially leverage this fact to achieve injective agreement on the parties involved and the session key.

*Home Network (AUSF and ARPF):* From the informal authentication properties defined in Section IV, we believe the properties directly concerning the point of view of the home network are A3–A7. Achieving all of these properties would be similar to achieving the traditional property of injective agreement on the identities of the UE, serving network, and home network, in combination with the term $K_{SEAF}$, as described in [20].

While we separate the components of the home network into the AUSF and ARPF for the sake of modelling the protocol, we consider them to be much more closely related than e.g., the relationship between the AUSF and SEAF. The ARPF receives and sends only one pair of messages, and the contents of the received Auth-Info Request message only indicate that a party wants to start a protocol run; it does not contain any cryptographically generated or signed terms, or any randomness generated by the initiator.

As the ARPF does not participate in the protocol after sending the `Auth-Info Response` message, it cannot know whether the UE responded to its challenge correctly or not. As the AUSF has sufficient information to determine the correctness of the response from the UE, and as the AUSF is part of the home network, we consider the AUSF and ARPF as a pair for the high level properties regarding authentication. Hence, we consider the final group of authentication properties from the point of view of the "home network", rather than either one of the AUSF or ARPF.

We analyse the full range of authentication properties for each party communicating (directly or indirectly) with each other party, and for the terms over which they might meaningfully agree. We provide full results in Section VIII.

## VII. Formalisation of Threat Model

We now consider how we formalised the threat model described in Section III. As this threat model does not allow compromise of components apart from *other* UEs (i.e., no SEAF, AUSF, or ARPF compromise is allowed), we only need to consider adversary key reveal (of other UEs' long term keys) and secure channel modelling.

### *A. Adversary key reveal*

Distinct from other adversary actions we include a rule allowing the adversary to compromise the long-term key K of UEs other than the 'honest' UE we consider directly. When the adversary triggers it, this rule outputs a user's long term key K to the public network channel, allowing the adversary to learn it. When we specify security properties, we include restrictions on the allowed events within protocol executions, such as when the adversary may perform key reveals. We achieve this by clauses preventing certain actions. To prevent the adversary from revealing an honest actor's long term K, we require that there are no events (recorded by 'Action Facts') revealing the key to the adversary. These Action Facts are parameterised by the ID of the party the key is for, so we can allow the compromise of *any* other long-term Ks, i.e., for other UEs.

### *B. Secure channel modelling*

Within our 5G-AKA Tamarin models we model the secure channels within the 5G Core Network using the standard secure channel abstraction, as used and described in [12]. This construction replaces Tamarin's adversary-controlled Dolev-Yao-style channels with secure channels where desired. This takes the adversary-controlled `In(msg)` and `Out(msg)` facts, and replaces them with secure channels. These facts are similar to the form `SndS(A,B,msg)` and `RcvS(A,B,msg)`, sending the term `msg` from `A` to `B`, who are explicitly named parties. We augment the standard construction very simply by including a 'channel name', 'SendType', 'ReceiveType', and associated metadata for ease of later channel and component selection and analysis. In practice, the `channelname` term describes which of the two secure channels the instantiation of the rule considers, and looks like `'seaf_ausf'` or `'ausf_arpf'`, and the `SendType` and `ReceiveType` terms just contain one of the strings `'SUPI'`, `'SEAF'`, `'AUSF'`, or `'ARPF'`. This construction ensures that the adversary cannot read or modify the contents of a message (`msg`) sent over this secure channel; likewise, the sender (`A`) and recipient (`B`) of each message cannot be modified or spoofed by the adversary.

By Tamarin's semantics, only rules with this fact in their conclusion can produce a fact `SndS(...)`, and only rules with the fact `RcvS(...)` in their premise can consume it, i.e., the Adversary cannot construct this itself. Assuming all of the protocol's rules are modelled and constructed correctly, i.e., all rules of the protocol honestly identify the sender and intended recipient, this will also guarantee the authenticity of sender and recipient. N.B. This construction itself says nothing about which session of the protocol the message was intended for, nor the order in which messages are delivered.

Each '`SndS(...)`' and '`RcvS(...)`' fact can only be consumed as a premise to a rule once. A message transmitted through this channel therefore cannot be replayed by the adversary; the adversary can only attempt to trigger the original rule which invoked `SndS(...)` again, but this rule's premises will have to be satisfied again before this can occur.

We believe this construction very closely matches the "e2e core network interconnection" channel requirements precisely as described in [5, §5.7.4] (cited in Section III-A), and hence we use it to model channels 2 and 3, i.e., the channels between the SEAF and AUSF, and the AUSF and ARPF respectively.

## VIII. Analysis and Results

We have described 5G-AKA's desired security properties informally and formally in Sections IV and VI, and the threat model under which we evaluate these properties. Our systematic analysis has allowed us to reach conclusions about which of these properties are upheld. We present our findings for secrecy and authentication.

**Secrecy properties:**

S1. Secrecy of honest subscriber's long term key K: ✓
S2. Secrecy of anchor keys $K_{\mathrm{SEAF}}$ and $K_{\mathrm{AUSF}}$: ✗

**Authentication properties:**

A1. SN and UE agree on the identity of UE: ✗
A2. UE and SN agree on the identity of SN: ✗
A3. HN and SN agree on the identity of UE: ✓
A4. UE and HN agree on the identity of HN: ✗
A5. UE and HN agree on the identity of SN: ✗
A6. UE, SN, and HN agree on $K_{\mathrm{SEAF}}$: ✗
A7. Anchor key $K_{\mathrm{SEAF}}$ must not be replayable: ✗

As the results show, we encountered various secrecy and agreement violations against 5G-AKA. The attack which violates the secrecy of the anchor key $K_{\mathrm{SEAF}}$ is of particular interest, and we discuss this in some detail in this section. We first give an informal overview of this violation before giving an in-depth description. After describing the secrecy violation in Section VIII-A, we consider the authentication violations in Section VIII-B.

The attack proceeds in a similar manner to that presented by Tsay and Mjølsnes's attack against the three-party UMTS-AKA in [26], which is between the serving network and home network. In our four-party attack, we take advantage of a race condition over the AUSF ↔ ARPF channel entirely within the home network (neither additional channel or component is defined in UMTS-AKA or LTE-AKA), rather than the SEAF ↔ AUSF channel at the interface of the serving and home networks.

### A. Secrecy violation

*Overview:* A malicious actor 'B' starts two 5G-AKA sessions with a local serving network at roughly the same time. One session is initiated by replaying an overheard SUCI (of the target, user 'A'), and the other session is with the malicious actor's own USIM and SUCI (for user 'B'). The sessions run in parallel, and result in a race condition; if this occurs, the AUSF will be unable to distinguish between the two responses containing the Authentication Vectors from the credential store (ARPF), and is liable to associate the wrong response (and resultant keys) with the wrong user. In the case that this occurs, the AUSF and SEAF will incorrectly believe that a set of Authentication Vectors and 'anchor key' were intended for user A (and derived from user A's long-term key $K_A$), when they were in fact derived from user B's long-term key $K_B$. As a result, the malicious user B will now be able to derive the anchor key, and use it to impersonate user A to the network. See Figure 8 for the message sequence chart of the attack.

*What does the secrecy violation break?:* We now give a more in-depth description of the secrecy violation.

The specific violated property which we consider now is the secrecy of the 'anchor key' $K_{SEAF}$ (and its cryptographic parent, $K_{AUSF}$), *from the points of view of the SEAF and AUSF*. That is, at the end of the 5G-AKA protocol run:

- the SEAF, AUSF, and a UE will have agreed and be in possession of a cryptographic anchor key, $K_{SEAF}$,
- the SEAF and AUSF believe this key is for an honest and un-compromised UE (in our example, user 'A' with 'SUPI-A' and 'SUCI-A'), and,
- both the SEAF and AUSF believe this key is secret from the attacker. It is not.

Thus, the protocol draft lacks a crucial containment property: an attacker that can compromise or gain access to the long-term key of a user (e.g., 'B') will be able to impersonate *any* user (e.g., 'A') to the SEAF and the AUSF, because it knows the $K_{SEAF}$ for a session that they believe to be for 'A'.

*Detailed attack scenario:* The attack takes place in two (possibly temporally and even geographically) separate phases. In the first phase, the attacker eavesdrops and records a legitimate encrypted/concealed SUPI, also known as a SUCI. In the second phase, the main body of the attack takes place. Full message definitions can be found primarily in TS 33.501 [5] (some are in TS 33.102, [4]). N.B. This can attack occur even *more* easily if the SUPI is transmitted unconcealed, i.e., not encrypted into SUCI form.

*Setup to the attack:*

1) A legitimate user 'A' with ID 'SUPI-A' is registered with its home network (HN). The attack does not require access to its long-term key $K_A$. This honest user initialises the 5G-AKA protocol, sending the SUCI-A (user A's ephemerally encrypted SUPI) and 'HN' to a SEAF. The user might then complete the protocol as normal.
2) The attacker eavesdrops on the public radio transmissions from the previous step, and records the message containing SUCI-A and HN.
3) The attacker purchases a legitimate USIM from the *same* home network as his intended victim; this has ID 'SUPI-B'. The attacker physically attacks and compromises the USIM, and extracts the long-term key $K_B$ of this USIM in its possession.[5]

*Main phase of the attack:* The message sequence chart of the main phase of the attack can be found in Figure 8.

1) The attacker initiates the 5G-AKA protocol by replaying to a SEAF the pre-recorded SUCI-A. The attacker sends a message containing 'SUCI-A' and the name of the user's home network to a SEAF in serving network 'SNID'.
2) The protocol proceeds as normal: the SEAF communicates with an AUSF in the specified home network by sending the '`5G-AIR`' message. This contains 'SUCI-A', and SNID.
3) In parallel with the first session, the attacker starts a 5G-AKA session for the USIM it owns (SUPI-B) with the same home network, via the same serving network (and SEAF). It starts the 5G-AKA session by sending its own concealed ID ('SUCI-B') and the name of the home network to the same SEAF as in the other, parallel session. The SEAF correctly treats this as a separate session.
4) The SEAF communicates with the AUSF in the home network by sending the '`5G-AIR`' message, containing 'SUCI-B' and SNID. The AUSF then sends the '`Auth-Info Request`' message to the home network's ARPF.
5) The SIDF de-conceals SUCI-B into SUPI-B, and the ARPF then responds by sending the '`Auth-Info Response`' message to the AUSF. This message contains terms derived from (the compromised) $K_B$, and the terms RAND, SQN, and SNID, but notably contains no reference to either the SUPI or the SUCI.
6) The '`Auth-Info Response`' message is received by the AUSF, but as this message does not have a SUPI or SUCI attached to it, the AUSF *does not know whether this message was for the session with 'SUCI-A/SUPI-A', or whether it was for the session with 'SUCI-B/SUPI-B'*. The AUSF can legitimately continue its session intended for 'A' with the '`Auth-Info Response`' message that was actually intended for the session with 'B'.
7) The AUSF then proceeds with the protocol, sending the `5G-AIA` message for 'SUPI-A' to the SEAF; this contains the anchor key $K_{SEAF}$ that the ARPF generated for 'SUPI-B', but now the AUSF associates it with 'SUPI-A' (and as a result, so does the SEAF). As the attacker has compromised SUPI-B's long-term key $K_B$ (and RAND and SQN are publicly transmitted during the protocol), the attacker can now construct the anchor key $K_{SEAF}$ that the AUSF and SEAF now believe is the anchor key for 'SUPI-A'. That is, the attacker can derive the $K_{SEAF}$ that the AUSF and SEAF believe to be for the (honest) 'SUPI-A' (and *not* 'SUPI-B'

[5] After discussion with a senior security researcher of a global carrier, we believe this physical extraction of $K_B$ is not necessary, although if this step is completed, from a practical point of view this gives the attacker even greater control over the timing and flow of messages.

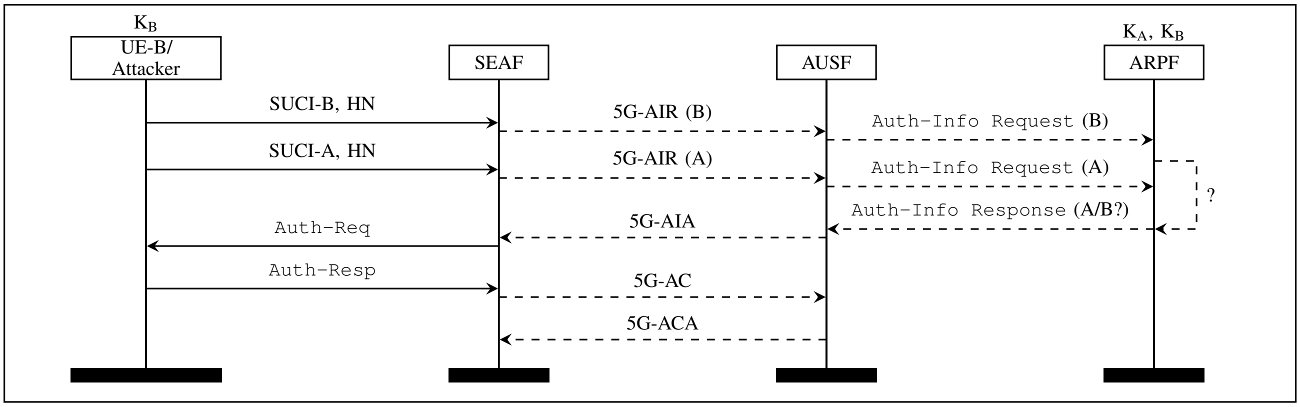

Fig. 8. The **attack** flows of the 5G-AKA Protocol, secrecy violation caused by session confusion.

which the attacker has compromised), completing the attack.

Counters or SQN values do not have any bearing on this attack, as only the ARPF and UE store what the 'correct' value of SQN is. The AUSF and SEAF do not use SQN directly in any calculations or derivations, and hence do not check whether it matches (or is greater than) stored values.

*Replaying ephemerally encrypted SUCIs:* In the concealed setting, our vulnerability relies on the SIDF accepting a previously replayed SUCI value. We discuss why we believe that adversary-replayed values will be accepted in this context.

A UE sends an ephemerally encrypted SUCI to conceal the SUPI, maintaining the privacy and unlinkability of the USIM's global ID. This attack does not violate that property. The UE uses an ephemeral public key (rather than static) to maintain its own unlinkability; hence the onus is on the UE to use a new ephemeral key with each run of the protocol to maintain this property. While the ARPF or SIDF maintaining a complete list of previously used ephemeral public keys is touched upon as a possible suggestion in the (formally withdrawn) Technical Report TR 33.899 [3], we find no evidence within TS 33.501 that this is required or even formally proposed, and therefore no evidence that an ARPF will not accept a re-used SUCI.[6]

Including a counter within the SUCI (then checked against the highest seen value) is sufficient for the concealed setting, but isn't sufficient overall: when the null-scheme is used for SUPI 'encryption', the attack then holds again. In the non-concealed case adding a counter isn't sufficient: there are no cryptographic protections at this stage, so the attacker can increment the last counter it observed. TS 33.501 (notably [5, §6.12.2]) does not mention a counter or any similar values within the SUCI.

Adding a counter prevents the attack described in this document from succeeding (because replaying an overheard SUCI will no longer be sufficient), but it means that an attacker merely has to learn a target user's un-concealed SUPI by some means to be able to impersonate them. We believe the purpose of concealed IDs is intended to maintain the *privacy* of the user's ID, and is not intended as a critical means to guarantee the overall authentication properties of the protocol.

## B. Authentication violations

There are also multiple authentication violations, which are listed in Section VIII. These fall into two categories: authentication of serving network ID, and session confusion.

*Authentication of the SNID:* This is a genuine violation of agreement over the identity of the SNID, as the UE never learns this term in an authenticated message. This causes agreement to fail on $K_{SEAF}$, as this term is derived by the UE from terms including SNID; the ARPF, AUSF, and SEAF will derive an anchor key $K_{SEAF}$ on which the UE will not agree if the adversary has injected an arbitrary SNID into the UE's (unauthenticated) serving network discovery phase.

This violation was originally discovered in concurrent work by Basin et al. [11]. Our four-party models and analysis confirms the existence of this agreement violation.

The adversary can inject an arbitrary SNID onto the UE ↔ SEAF channel: the UE then has no way of validating the authenticity of SNID. The adversary must block the UE's Auth-Resp response message, as otherwise a genuine SEAF would quickly discover that the UE generated HRES* and AUSF generated HXRES* do not match. As the UE then does not receive any rejection messages from the serving network, it assumes the authentication was successful, and so from the UE's point of view, the protocol is finished. The UE will then attempt to communicate normally with a nearby base station.

Outside of the definition of 5G-AKA, TS 33.501 excludes this 'attack': see Section 5.10, cited in Figure 7. This attempted communication and use of the $K_{SEAF}$ with the wrong SNID *after the protocol has finished* will fail, but we agree that this attack still violates serving network authentication in the explicit sense. We leave the question of whether this violation would allow the adversary any separate, meaningful benefit for future work. We propose and formally verify a fix for this authentication violation in Section IX-B.

*Session confusion:* Allowing sessions for different users to be accidentally confused causes the violation of both secrecy and authentication properties. The message flow behind many of the authentication violations is the same as the secrecy violation described in Section VIII-A. Here, session confusion occurs at the same point, i.e., two sets of authentication vectors sent by the ARPF are received by the AUSF at roughly the same time resulting in a race condition, and the intended session for each

[6] TR 33.899 is a study collecting possible proposals from multiple authors across 3GPP, some of which were considered for inclusion within TS 33.501. This document has now been withdrawn, but gives insight into different suggested proposals for 5G security from member companies of 3GPP.

is not reliably identified. We discuss broader implications of the discovered vulnerabilities in Section XI.

## IX. Proposed Fix

The essence of the session confusion-based attacks is identity mis-binding. This leads to two ways to prevent these attacks: binding the identities of the intended parties to each message all the way through the protocol, or ensuring a one-to-one mapping between the high-level 5G-AKA sessions and internal AUSF ↔ ARPF sessions. We propose the latter: while both prevent the secrecy violation, we believe this approach also upholds strictly stronger authentication properties than using the UE's identity as the session ID.

We have formally verified that this solution prevents the secrecy violation and various authentication violations. We provide full formal verification results for this in Section IX-B.

### *A. Proposed fix: tighter session binding*

Session confusion is dependent on the ability of messages between the AUSF and ARPF from one 5G-AKA session to end up in that channel for another 5G-AKA session. Currently, there is nothing in the specification that prevents this. We propose the following method to prevent this:

**The protocol should include a fresh (unique, random) value in `Auth-Info Request`. The ARPF should include this in `Auth-Info Response`, and the AUSF should check that they match.** To ensure similar session binding across the SN / HN boundary, i.e., between the SEAF and AUSF, the SEAF should also include a *different* fresh value in `5G-AIR`; the AUSF should include the same value in `5G-AIA`; the SEAF should then check that they match.

This successfully binds the correct sessions to messages at each stage of the protocol, preventing session mis-binding attacks from occurring. We believe that this modification will have a negligible impact on the efficiency of the protocol.

### *B. Verification results for proposed fix*

We have formally analysed the properties of the 5G-AKA protocol with the proposed fix from Section IX-A. Specifically, we include fresh values in the messages between the AUSF ↔ ARPF and SEAF ↔ AUSF. While our session-ID binding solution correctly fixes the secrecy violations and many of the previous agreement violations, some of the properties, particularly from the point of the UE, are still violated.

We state changed results compared to those from Section VIII. now with tighter session binding: properties S1 and A3 are still upheld; S2, A1, and A4 are now also upheld.

| | | |
|---|---|---|
| S2. | Secrecy of anchor keys $K_{SEAF}$ and $K_{AUSF}$: | ✓ |
| A1. | SN and UE agree on the identity of UE: | ✓ |
| A4. | UE and HN agree on the identity of HN: | ✓ |

This is a distinct improvement on the results from Section VIII, but due to the lack of agreement on the SNID, A2 and A5–A7 are still not achieved.

*Agreement on the ID of the SEAF:* As discussed in Section VIII-B, this is a violation of agreement over the identity of the SNID, as the UE never learns this term in an authenticated message. This causes agreement to fail on the term $K_{SEAF}$, as this term is derived by the UE from a series of terms including SNID; the ARPF, AUSF, and SEAF will derive an anchor key $K_{SEAF}$ on which the UE will not agree if the adversary has injected an arbitrary SNID into the UE's unauthenticated serving network discovery phase.

*Fix for SNID agreement violation, and verification:* To correct this violation of agreement, we propose that the SNID is added to the definition of the MAC (defined in TS 33.102 [4]), as this is keyed by the long-term secret key, K. This would redefine the MAC to: `MAC = f1(K, <SQN, RAND, SNID, AMF>)` where it did not contain the 'SNID' term before. We have formally verified that this minor change now allows the 5G-AKA protocol to gain non-injective agreement on the SEAF's identity from the UE's point of view, and both non-injective and injective agreement on the $K_{SEAF}$ from the UE's point of view. We re-state the complete table of results from the UE's point of view with this final fix.

**Secrecy properties under (Fix 1 + SNID fix):**

| | | |
|---|---|---|
| S1. | Secrecy of honest subscriber's long term key K: | ✓ |
| S2. | Secrecy of anchor keys $K_{SEAF}$ and $K_{AUSF}$: | ✓ |

**Authentication properties (Fix 1 + SNID fix):**

| | | |
|---|---|---|
| A1. | SN and UE agree on the identity of UE: | ✓ |
| A2. | UE and SN agree on the identity of SN: | ✓ |
| A3. | HN and SN agree on the identity of UE: | ✓ |
| A4. | UE and HN agree on the identity of HN: | ✓ |
| A5. | UE and HN agree on the identity of SN: | ✓ |
| A6. | UE, SN, and HN agree on $K_{SEAF}$: | ✓ |
| A7. | Anchor key $K_{SEAF}$ must not be replayable: | ✓ |

With these fixes, we believe 5G-AKA now explicitly achieves all of its desired security properties in the symbolic model.

### *C. Alternative fixes*

We have considered several alternative fixes, but they all seem either more complex or insufficient. For example, one might consider putting unique nonces in other ways in the channels to solve this attack, especially since this is likely to be implemented at a lower level due to engineering concerns. Alternatively, one might create explicit *identity* binding, rather than session binding. This would prevent the identity mis-binding attack, but it would not prevent other agreement violations. While the SUPI is globally unique, and hence plausibly suitable as a session ID value, each session would use the same ID, i.e., the SUPI. This does not prevent two sessions *from the same SUPI* from becoming confused, which would violate agreement on the resultant session key.

### *D. Session binding does not always imply security*

It may be tempting to conclude that any solution to the race condition (in the honest case) prevents the attack. It turns out this is not the case. To prove this, we give an example of a solution that succeeds in preventing honest session confusion, and hence the race condition in the honest case, but is still vulnerable to a variant of the described attack. The underlying idea is that it is possible to prevent honest session confusion in a way that can be subverted by an adversary.

Assume the standard required the UE to choose a fresh nonce and append it to the first message sent to the SEAF; this otherwise contains the SUCI, SUPI, or 5G-GUTI. If this UE-chosen nonce was then appended by all other parties (SEAF, AUSF, and ARPF) to messages within this protocol run, this would be sufficient to prevent honest session confusion. (If the first message contained a SUCI or 5G-GUTI it would also not reveal any information about the user's identity.) The result would be that without adversarial interference, each request from the AUSF to the ARPF now uses a unique value, which enables binding the response uniquely to the request.

This is not sufficient to prevent the secrecy attack. If the 'unique nonce' is chosen by the UE, a variant of the attack is still possible: the adversary can repeat a nonce sent by an honest UE, or it can use the same nonce twice. This leads to the AUSF using the same nonce for two separate requests to the ARPF, which enables confusion of the responses. This improves the probability of session confusion, as no other honest sessions should accidentally become confused with these two.

Our proposed fix in Section IX-A relies upon tighter session binding, but avoids the above error by ensuring the adversary cannot control the session binding term; the adversary cannot influence it as the choice is made within the 5G core network.

There are many other engineering solutions which would *coincidentally* prevent our attack, but we believe this demonstrates that not all solutions to the problem of session confusion necessarily prevent it. We believe this further demonstrates that any solution required to prevent this attack must be mandated by the protocol definition in the standard, and that security critical details must not be left as an implementation decision.

## X. Compromised Channels and Components

We now consider *stronger* adversary capabilities against the previously secure channels and components: $\mathcal{A}_{Stronger}$. All of the following compromise of channels and components are *explicitly disallowed* by the threat model within TS 33.501 (i.e., $\mathcal{A}_{Standard}$); these used to explore further the range of security properties achieved by 5G-AKA in various compromise scenarios. We now describe our modelling for compromised channels and components, followed by analysis and results.

### A. Compromised channel modelling

We consider compromise of each of the 'secure' channels within the 5G Core Network. We model both read-only and full 'Dolev-Yao'-style channel compromise. We model this automatically by adding to the previously described channel communication rules: the definitions of `send_secure` and `receive_secure` are still in place, but they are also joined by definitions for new rules, `send_insecure` and `receive_insecure` (see [15]). These allow us to create fine-grained channel access for the adversary. For the 'Read-Only' compromise of channels, we do not include the `receive_insecure` rule so the adversary cannot inject or modify terms on the channel in question.

These rules allow the adversary unrestricted ability to access any arbitrary channel: we limit the adversary's behaviour to accessing only a particular type of channel through the use of 'restrictions', matching against specific channels.

### B. Compromised component modelling

We consider the compromise of one or more components within the protocol.

*Compromise of the SEAF and/or AUSF:* Before a protocol run, neither the SEAF or AUSF has any shared secret with each other, the UE, or the ARPF within the 5G-AKA protocol. Instead, they leverage their secure and authentic channel access: this is what prevents the adversary from impersonating them to other actors within the 5G core network.

To ensure secure and authentic 5G core network access, the SEAF, AUSF, and ARPF will necessarily have some shared secret(s), but this is explicitly performed at a lower layer of the 5G core network compared to the 5G-AKA protocol.

Without other shared secrets, for modelling purposes, taking over a component's secure network channel access is sufficient to fully impersonate the component; this is the case for the SEAF and AUSF. This is achieved through adding '`component_compromised_send`' and '`component_compromised_receive`' rules to the Tamarin models. As with channels, we restrict which components the adversary can compromise.

*Compromise of the UE and/or ARPF:* The UE and ARPF explicitly share secrets in 5G-AKA. As the UE does not have secure network access, it is sufficient to give the adversary the UE's long-term key K. Within our models, to compromise the ARPF, it is sufficient to compromise the initiating UE's long-term key K in combination with giving the adversary access to the relevant secure network channel.

*Compromise of **other** components:* We consider two main component compromise strands: first, if we compromise a component of type *X*, we say the adversary can compromise *all* components of type *X*; e.g., allowing SEAF compromise implies that all SEAFs can be compromised, whether these are the ones in use by the particular run of the protocol which we consider or not. We refer to this as "All-*X*".

Second, we consider the case where the adversary may compromise all components of type *X apart from "mine"*, e.g., if an honest component such as the UE thinks it is talking to a SEAF with identity 'SNID', then it really is, and the adversary has not compromised this specific SEAF; the adversary can still compromise any other SEAF. We refer to this as "Not-My-*X*".

The All-*X* scenario considers the importance of components involved directly in the protocol; Not-My-*X* considers whether an honest actor needs to trust all actors within the 5G core network, or just the ones with whom it believes it is talking.

### C. Compromised channels: analysis and results

Each channel-compromise threat model introduces new challenges. The channel-compromise threat models which we consider are: (1) No compromise (see Sections VII and VIII). **Read-only channels:** (2a) SEAF ↔ AUSF channel is readable by the adversary, (2b) AUSF ↔ ARPF channel is readable by the adversary, (2c) Both channels are readable by the adversary. **Dolev-Yao channels:** (3a) SEAF ↔ AUSF channel is D-Y compromised, (3b) AUSF ↔ ARPF channel is D-Y compromised, (3c) Both channels are D-Y compromised.

*Compromised channel results:* Compromising secure channels has broadly the expected effect. All results assume adoption of the channel-session binding fix. (1) See Section VIII for 'no compromise' results. (2a,b,c) Read-only compromise of each of the SEAF ↔ AUSF channel, AUSF ↔ ARPF channel, and both channels together causes all security properties to be violated *except* S1, A3, and A4 (see Section IV). (3a) Full D-Y (read/write) compromise of the SEAF ↔ AUSF channel causes all security properties to be violated except S1 and A4. (3b) D-Y compromise of the AUSF ↔ ARPF channel causes all security properties to be violated except S1 and A3. (3c) D-Y-style compromise of *both* secure channels causes all security properties to be violated except S1.

Even with strong session binding, D-Y or read-only compromise of any secure network channel involved in the 5G-AKA protocol is devastating for both secrecy and authentication, especially (and critically) from the UE's point of view.

### D. Compromised components: analysis and results

Achieving results in TAMARIN for compromised components has proved more difficult than the channel compromise results, leading to fewer results terminating automatically. The majority of the component compromise results were achieved through manual direction of TAMARIN's interactive mode, and we provide descriptions of how to repeat these manual proofs in the README associated with `5G-AKA.m4` [15]. Choice of heuristic or manual intervention only affects termination and the duration of computation, not the final result.

Our first scenario (All-*X*) considers the importance of the components involved directly in the protocol; the second (Not-My-*X*) considers whether an honest actor needs to trust *all* components within the 5G core network, or just the ones with whom it believes it is communicating.

*a) Secrecy and authentication results: All-X:* (1) Allowing the adversary to compromise up to and including all **SEAFs** causes **all** security properties to be violated *except* properties S1 and A4 (see Section IV). (2) Allowing the adversary to compromise all **AUSFs** causes all security properties to be violated except S1 and A4. (3) Allowing the adversary to compromise all **ARPFs** causes all security properties to be violated except S1 and A3. We conclude that compromise of any 5G core network component used by 5G-AKA has a severely detrimental effect on the protocol's security properties.

We now consider the results of component compromise *excluding* the components with whom an actor believes they are communicating, i.e., Not-My-*X*.

*b) Secrecy and authentication results: Not-My-X:* (1) Allowing the adversary to compromise all **SEAFs** apart from 'mine' causes does not violate any security properties when the SNID fix is adopted. A1, A2, A5, A6 are only achieved with adoption of the SNID fix. We were not able to achieve termination for A7. (2) Even with the SNID fix, allowing the adversary to compromise all **AUSFs** apart from 'mine' causes all security properties to be violated except S1, S2, A3 and A4, although we believe these violations to be theoretical, and not meaningfully instantiable. (3) Allowing the adversary to compromise all **ARPFs** apart from 'mine' causes does not violate any security properties when the SNID fix is adopted. A2, A5, A6 are only achieved with adoption of the SNID fix. We were not able to achieve termination for A7.

### E. Discussion: $\mathcal{A}_{Stronger}$

We have shown that preventing compromise of each core component is essential for secrecy and authentication properties, as all components have significant influence over whether the protocol achieves the properties it seeks to uphold. For the "Not-My-*X*" scenario, we believe our analysis further demonstrates the importance of explicitly including the SNID in an authenticated part of the protocol's messages.

Compromise of secure *channels* has a similarly devastating effect. Unsurprisingly, Dolev-Yao-like channel compromise allows an adversary to violate most security properties. Read-only access does not allow the adversary to break many agreement properties *between 5G core components*; more interestingly, this read-only access to either secure channel allows the adversary to violate almost all agreement properties *from the UE's point of view*, as this new information allows the adversary to impersonate a serving and/or home network successfully to the UE over the insecure UE ↔ SEAF channel.

The 5G specification requires lawful intercept capability, whether enabled or not. Older (current) methods to implement this are described in TS 33.106 [6]. The 5G specific methods for this are described and explored in TS 33.842 [2], which is currently at version 0.0.0 since November 2017, and is a skeleton document with little to no detail. If one implements lawful intercept by giving law enforcement read-only access to one of the channels, our analysis implies that law enforcement immediately also gains the power to impersonate. This would violate the principle of least privilege. Given such subtleties, it would be prudent to explicitly incorporate any lawful intercept mechanisms in the security analysis of the protocol, such that one can provide assurance that the additional mechanism provides what is required by law, but does not accidentally provide any further capabilities. Analyses like the one we perform here are suitable for this purpose.

While the 5G-AKA protocol meets its desired security properties after inclusion of our proposed fixes, it is still a very fragile protocol. 5G-AKA loses the ability to uphold most of its desired properties very quickly upon compromise of almost anything outside of its explicit (relatively weak) threat model compared to almost all modern key-exchange protocols.

## XI. ATTACK IMPLICATIONS, DISCLOSURE, AND IMPACT ON TS 33.501

The secrecy violation described in Section VIII-A allows an adversary to impersonate *another* user to a serving network. From the point of view of 5G-AKA this allows the attacker to agree an anchor key (gaining serving network access) dishonestly, under newly generated false credentials of a legitimate user. This is a substantial containment problem.

The attack relies on a race condition between two sessions of the protocol. The attack is probabilistic, and an attacker would not be able to guarantee success; however, in any secure protocol, there ought not exist *any* run of the protocol adhering to the threat model which violates the security properties.

N.B. This does **not** allow an attacker to decrypt any honest radio traffic originally generated by the legitimate user.

### A. Potential practical implications

In the real world, we conjecture that this attack might allow an attacker to access a serving network in the name of a legitimate user other than itself. This attacker could then bill services, air-time, or access charges to another user; this is clearly not the intended behaviour or level of security required.

We are not confident of the range of further authentication and authorisation procedures which may or may not be in place distinct from the 5G-AKA protocol, or any billing—authentication procedures: e.g., whether specific billing actions sent back to a user's home network are tied to and verified against a named anchor key or not. We note that an ARPF *would* be able to establish that the anchor key was not for the correct user; but that an AUSF or any other party without direct access to the honest user's long-term key K would not. We believe it is plausible that once access is granted in the form of an anchor key, this key is sufficient to allow a user to perform the normal range of actions within a network.

We acknowledge that there may be other technical measures within any real-world 5G network that make full implementation of this attack impossible in the real world. Regardless, we believe that any authentication and key agreement protocol as defined must meet its own required security properties.

The real-world practicality of this attack will depend on carrier-specific implementations, and to the best of our knowledge these are not available. This makes it hard to provide evidence of practicality, but we believe this research shows we could easily create a 'correct' implementation of the standard that makes the attack feasible. With access to a range of 5G implementations, we hypothesise that we would find vulnerable instances; at the time of research this was not possible.

The strongest statement we can definitively make is that the 5G-AKA protocol on its own does not meet its security requirements. However, as the primary method for authentication and key agreement within 5G, we believe that 5G networks should not rely solely upon secondary mechanisms for security; we believe this is sufficient reason on its own to fix the protocol to prevent this and similar attacks.

### B. Liaison with 3GPP SA3 and CT4

At the time of writing, 5G (and 5G-AKA) is not yet an implemented, used, or complete standard; our publication of this work was intended to highlight issues *during* the standardisation phase, rather than after it has been finalised. As a result, we communicated directly with the relevant working groups.

After discovering the secrecy violation, we prepared a document describing the vulnerability, its potential implications, and our proposed fix. We then contacted members of the 3GPP SA3 (Security) working group, informing them of our research, and requesting they read and comment upon it.

Following the document's distribution, we received mostly supportive feedback from members of SA3 and other researchers. Some SA3 members did not view the vulnerability as an issue with TS 33.501, suggesting that it was not the standard's responsibility to ensure security, and that individual vendors could introduce further measures to guarantee security if they desired. We respectfully but strongly disagree.

Senior security experts from a global carrier kindly helped us to prepare a Change Request [27] to TS 33.501. They submitted this on our behalf to 3GPP SA3. As a result of our publicly released document and change request, SA3 have stated that they believe our described race condition could arise, and that they need to ensure its mitigation. This has resulted in a formal liaison document detailing our described mis-binding, and how this could lead to parties being "unable to correlate the different responses to the respective requests" [17].

The liaison document requests of 3GPP working group CT4 whether they agree that this lack of binding is present, whether it has been taken into account, and if so, how the race condition is avoided (requesting reference to the relevant specification). Finally, the document attaches a change request by another global carrier, which is proposed as a solution. This is a generic adaptation of our proposed fix from Section IX-A.

## XII. Conclusions

In this research we have demonstrated issues within the draft 5G-AKA protocol, particularly one which, if unmitigated, could potentially allow a malicious actor to impersonate an honest user to a network. We propose a possible fix, and we have verified its correctness using the Tamarin Prover. We have worked with 3GPP, encouraging them strongly to adopt our proposed fix, and are pleased with the progress made here.

We believe our results demonstrate the importance of fine-grained component-based formal modelling: without this level of detail in modelling, we would not have discovered the presented race condition.

We recognise that standards often make implicit assumptions about the reality of engineering solutions, and that there may be other mechanisms in place which in practice mitigate the real-world impact of this protocol vulnerability. It might be tempting to think our attack would be prevented if underlying layers (accidentally) provide session binding, but we showed in Section IX-D that this need not be the case. Regardless of whether or not implementations at lower layers accidentally prevent the described attack, our analysis demonstrates that such mechanisms would in fact be security-critical.

We emphasise that security critical properties of any protocol *must not depend on implicit engineering solutions*; the specification of a standard should be such that *any* correct implementation provides the desired security properties. This is not true for the modelled TS 33.501 v0.7.0, [5], or the latest version at the time of writing, v15.1.0.

Many of the encountered issues are exacerbated by continued reliance on symmetric cryptography. We recognise that legacy considerations restrict the choices available to protocol designers, but continuing to rely solely on symmetric cryptography seems wholly inadequate in 2018. Future standards could achieve much stronger security properties in authentication and secrecy with the introduction of a modern, asymmetric key-exchange mechanism at the core of any new 'AKA' protocol.

Identity binding and protocol design is tough, especially within complex, multi-party protocols with subtle assumptions.

We believe that the discovery of these issues further demonstrates the importance of systematic automated verification for security-critical functionality and protocols.

Finally, we want to emphasise the importance of communication between academia and industry: as we describe earlier, Tsay and Mjølsnes [26] discovered a very similar style attack across a different network boundary in 2012, but when designing 5G-AKA, 3GPP do not appear to have taken this into account. If they had, we believe our presented vulnerability could also have been avoided. When we contacted 3GPP with our report, not a single reply mentioned this previous research.

While the implications of some of the issues we discuss can be subtle and tricky to convey, we need to ensure these can still be communicated to industry clearly and accurately. Some parties may not yet be used to receiving and acting upon results and feedback from academia. In spite of this, in our view, it is important to continue to analyse systems, providing feedback to the relevant stakeholders, ensuring that our research is received, read, and most importantly, acted upon.

## References

[1] *TS 33.902: 3G Security: Formal Analysis of the 3G Authentication Protocol*, 3GPP, 1999. [Online]. Available: http://www.3gpp.org/DynaReport/33902.htm

[2] *TR 33.842: Study on Lawful Interception (LI) service in 5G*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), November 2017, version 0.0.0., https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3184.

[3] *TR 33.899: Study on the security aspects of the next generation system (SPECIFICATION WITHDRAWN)*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), August 2017, version 1.3.0.

[4] *TS 33.102: 3G security; Security architecture*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), March 2017, version 14.1.0.

[5] *TS 33.501: Security Architecture and Procedures for 5G System*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), December 2017, version 0.7.0.

[6] *TR 33.106: 3G security; Lawful interception requirements*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), June 2018, version 15.1.0.

[7] *TS 33.401: 3GPP System Architecture Evolution (SAE); Security architecture*, 3rd Generation Partnership Project (3GPP); Technical Specification Group Services and System Aspects (SA3), January 2018, version 15.2.0.

[8] S. Alt, P. Fouque, G. Macario-Rat, C. Onete, and B. Richard, "A cryptographic analysis of UMTS/LTE AKA," in *Applied Cryptography and Network Security - 14th International Conference, ACNS, Proceedings*, 2016, pp. 18–35.

[9] M. Arapinis, L. I. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon, and R. Borgaonkar, "New privacy issues in mobile telephony: fix and verification," in *the ACM Conference on Computer and Communications Security, CCS'12*, 2012, pp. 205–216.

[10] J. Arkko, V. Lehtovirta, and P. Eronen, "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')," *RFC*, vol. 5448, pp. 1–29, 2009. [Online]. Available: https://tools.ietf.org/html/rfc5448

[11] D. A. Basin, J. Dreier, L. Hirschi, S. Radomirovic, R. Sasse, and V. Stettler, "A Formal Analysis of 5G Authentication," in *ACM Conference on Computer and Communications Security*. ACM, 2018.

[12] D. A. Basin, S. Radomirovic, and L. Schmid, "Modeling human errors in security protocols," in *IEEE 29th Computer Security Foundations Symposium, CSF*, 2016, pp. 325–340.

[13] B. Blanchet, "A Computationally Sound Mechanized Prover for Security Protocols," *IEEE Trans. Dependable Sec. Comput.*, vol. 5, no. 4, 2008.

[14] C. Brzuska and H. Jacobsen, "A modular security analysis of EAP and IEEE 802.11," in *Public-Key Cryptography - PKC 2017 - 20th IACR International Conference on Practice and Theory in Public-Key Cryptography, Proceedings, Part II*, 2017, pp. 335–365.

[15] C. Cremers and M. Dehnel-Wild, "5G-AKA Tamarin Models," 2018. [Online]. Available: https://people.cispa.io/cas.cremers/tamarin/5G/

[16] C. Cremers, M. Horvat, J. Hoyland, S. Scott, and T. van der Merwe, "A comprehensive symbolic analysis of TLS 1.3," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017*, 2017, pp. 1773–1788.

[17] S. de Kievit, "S3-181468-v3 - LS to CT4 on avoiding race condition in 5G AKA," 3GPP SA WG3 Meeting #91, 16 – 20 April 2018. [Online]. Available: http://www.3gpp.org/ftp/TSG_SA/WG3_Security/TSGS3_91_Belgrade/Docs/S3-181468.zip

[18] S. R. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino, "LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE," in *25th Annual Network and Distributed System Security Symposium, NDSS*, 2018.

[19] G. M. Køien, "Mutual entity authentication for LTE," in *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference, IWCMC*, 2011, pp. 689–694.

[20] G. Lowe, "A Hierarchy of Authentication Specifications," in *Proceedings 10th Computer Security Foundations Workshop*, Jun 1997, pp. 31–43.

[21] S. Meier, B. Schmidt, C. Cremers, and D. A. Basin, "The TAMARIN Prover for the Symbolic Analysis of Security Protocols," in *Computer Aided Verification - 25th International Conference, CAV. Proceedings*, 2013, pp. 696–701.

[22] H. Mun, K. Han, and K. Kim, "3G-WLAN interworking: security analysis and new authentication and key agreement based on EAP-AKA," in *2009 Wireless Telecommunications Symposium, WTS*. IEEE, 2009, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4977245

[23] P. O'Hanlon, R. Borgaonkar, and L. Hirschi, "Mobile Subscriber WiFi Privacy," in *2017 IEEE Security and Privacy Workshops, SP Workshops 2017*, 2017, pp. 169–178.

[24] K. G. Paterson and T. van der Merwe, "Reactive and Proactive Standardisation of TLS," in *Security Standardisation Research*, 2016.

[25] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," Network Working Group, Internet Engineering Task Force (IETF), RFC 8446, August 2018. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-tls-tls13/

[26] J. Tsay and S. F. Mjølsnes, "A vulnerability in the UMTS and LTE authentication and key agreement protocols," in *Computer Network Security - 6th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security, MMM-ACNS. Proceedings*, 2012, pp. 65–76.

[27] Vodafone, "S3-180727 - pCR to TS33.501 - Session binding to prevent potential 5G-AKA vulnerability," 3GPP SA WG3 Meeting #90Bis, 26 Feb – 2 March 2018. [Online]. Available: http://www.3gpp.org/ftp/TSG_SA/WG3_Security/TSGS3_90Bis_SanDiego/Docs/S3-180727.zip