# A Vulnerability in the UMTS and LTE Authentication and Key Agreement Protocols

Joe-Kai Tsay and Stig F. Mjølsnes

Department of Telematics
Norwegian University of Sciences and Technology, NTNU
{joe.k.tsay,sfm@item.ntnu.no}

**Abstract.** We report on a deficiency in the specifications of the Authentication and Key Agreement (AKA) protocols of the Universal Mobile Telecommunications System (UMTS) and Long-Term Evolution (LTE) as well as the specification of the GSM Subscriber Identity Authentication protocol, which are all maintained by the 3rd Generation Partnership Program (3GPP), an international consortium of telecommunications standards bodies. The flaw, although found using the computational prover CryptoVerif, is of symbolic nature and can be exploited by both an outside and an inside attacker in order to violate entity authentication properties. An inside attacker may impersonate an honest user during a run of the protocol and apply the session key to use subsequent wireless services on behalf of the honest user.

**Keywords:** Applied Cryptography, Vulnerability Assessment, Security Protocols, Authentication, Mobile Network Security, LTE, UMTS

## 1 Introduction

These are exciting times in the development of mobile networks. The Global System for Mobile communication (GSM) and UMTS mobile networks are a worldwide success with now about 6 billion supscriptions [16], and still growing. New mobile systems are rolled out, including the 3GPP recent developments named 'Long Term Evolution' (LTE) and 'System Architecture Evolution' (SAE), which have become a forerunner for the fourth generation (4G) generation mobile communication system. The new system is called 'Evolved Packet System (EPS), emphasizing the all-IP packet switching design throughout the system onto the user's mobile terminal. [1] As more and more people take advantage of the accelerated internet access through their mobile phones, the recent international concern about securing the cyberspace and critical infrastructures certainly must include mobile networks. There is a multitude of security issues in such large networked systems. Here we will focus on the mobile terminal access security by means of an authentication and key agreement protocol. Weaknesses

[1] Although EPS is the proper technical term for this new 3GPP mobile system generation of SAE/LTE, we will use the most well-known name LTE.

in this protocol may not only lead to revenue loss to mobile operators but might also facilitate cyber crime.

The LTE AKA protocol is based on the Universal Mobile Telecommunications System (UMTS) AKA protocol, which is widely used today for third generation (3G) wireless networks, and which itself is the successor of the GSM Subscriber Identity Authentication (SIA) protocol. With the persistent spread of these mobile network systems, these authentication protocols have arguably become the most widely used security protocols today. While there exist formal analyses of UMTS AKA in the *Symbolic Model* of security (also called the *Dolev-Yao* model and inspired by [14]), this is in fact the first analysis of LTE AKA to date.

We report on a preliminary result of an ongoing analysis [18] of UMTS AKA and LTE AKA with the tool CryptoVerif [13] that can prove the security of protocols directly in the computational model. We discover a previously undetected flaw in the specifications of both UMTS AKA and LTE AKA. We note that the specifications of the GSM SIA protocol [9, 8] suffer, strictly speaking, from the same vulnerability (cf. Section 3.3). The vulnerability can be exploited by both outside and inside attackers in order to break authentication of a user to a serving network. Furthermore, inside attackers may impersonate an honest user and use wireless services on his behalf without the user being present on the network at that time. We reported the vulnerability to the 3GPP where the issue is currently under investigation. We have not tested current implementations for susceptibility to these attacks (cf. Section 3.1). We propose a simple correction to UMTS/LTE AKA and are working on CryptoVerif proofs of correspondence (*i.e.* authentication) and secrecy properties for the session key.

**Related Work** Annex B of the 3GPP technical report in [1] documents a formal analysis of the UMTS AKA protocol using a BAN logic variant. The analysis verifies authentication and secrecy properties. The flaw that we present here is not detected in [1] because strong assumptions (the *prerequisites on SN's side*) are used which already eliminate the weakness in the protocol. The GSM SIA protocol does not provide authentication of the access network to the user and the interoperability of the GSM and UMTS systems perpetuates this attack possibility, reported in [17]. Our analysis is not directed to the problems of interoperability between LTE/UMTS/GSM. A redirection attack on the UMTS AKA is reported in [19], which exploits the observation that the user is not able to authenticate the identity of the *serving* network because this is not included in the authentication vector provided by the home network. The new LTE AKA specification is designed to fix this weakness. A recent paper focuses on the privacy properties of the UMTS AKA protocol [10]. They use the tool ProVerif [11] for a formal analysis, and the paper describes an attack that enables the adversary to track a user. This is done by exploiting different error messages that are returned by UMTS AKA. The analysis models the UMTS AKA as a simplified two-party protocol between a user and the core network. However, by

reducing UMTS AKA to a two-party protocol, the weakness uncovered in the present work is concealed.

**Structure of this work** In Section 2, we will give an overview of the Mobile Network architecture and give a description of the UMTS AKA and LTE AKA protocols. In Section 3, we describe the flaw we found in the specifications of UMTS and LTE AKA and its consequences, where we discuss the relevance of the flaw for GSM SIA in Section 3.3. We conclude in Section 4.

## 2 UMTS and LTE Authentication and Key Agreement

### 2.1 Overview of the Mobile Network Architecture

For both UMTS and LTE the basic network architectures are very similar. In comparison to UMTS, the network elements used for LTE are upgraded and mostly renamed. However, they fulfill the analogous tasks in both cases. In order to avoid unnecessary confusion over terminology, we give a unified description of the network architectures of UMTS and LTE at the level of detail necessary for understanding our analysis and the vulnerability presented below. Basically, the mobile network architecture comprises three parts, that is, the user's mobile equipment $U$, the Radio Access Network (RAN), and the Core Network (CN). The user equipment consists of the mobile equipment and a tamper-resistant chip card, the *Universal Subscriber Identity Module* (USIM). The USIM is issued by a mobile operator to a subscriber and contains the International Mobile Subscriber Identity (IMSI), the permanent key of the subscription shared between subscriber and operator, and the cryptographic algorithms for the authentication protocol. In the following, we will use the terms *user*, *subscriber* and *user equipment* interchangeably. Each mobile operator runs an *Authentication Center* ($AuC$) server within its core network that contains the security related information of all the subscribers of the operator and generates temporary security credentials to be used by a user and a core network to establish authentication guarantees and set up session keys. The core network is divided into a *serving network* $S$ and a *home network* $H$, where the latter contains and maintains the AuC and the serving network is responsible for the communication to the user equipment through the radio access network.

The serving network and the home network do not necessarily belong to the same security domain, *i.e.* they may be controlled by different mobile operators. A subscriber $U_1$ of a mobile operator $OP_1$ with home network $H_1$ may *roam* into the domain of mobile operator $OP_2$'s radio access network maintained by serving network $S_2$. If $OP_1$ has a roaming agreement with $OP_2$, then $U_1$ will be able to access the mobile network through $S_2$'s radio access network. In this case, the connections between $S_2$ and $H_1$ are called *inter-domain connections*.

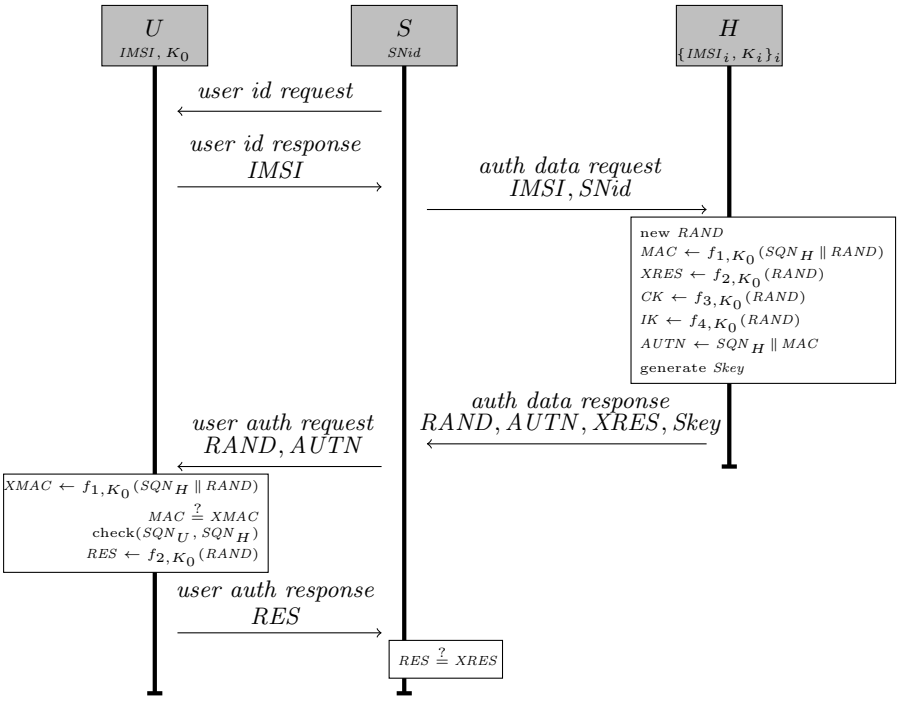


**Fig. 1.** The UMTS/LTE Authentication and Key Agreement Protocol. The session key in UMTS is $Skey \leftarrow CK \,\|\, IK$, and in LTE it is $Skey := K_{ASME} \leftarrow KDF(SQN_H \,\|\, CK \,\|\, IK \,\|\, SNid)$.

### 2.2 The UMTS & LTE AKA Protocols

Figure 1 shows the message sequence diagram description of the authentication and key agreement protocol in a unified way for UMTS and LTE on a similar level of detail as depicted in [3, 7]. The protocol is executed between user $U$, visited serving network $S$ and $U$'s home network $H$. $U$ and $H$ share the long-term key $K_0$ and a set of algorithms $f_1, \ldots, f_4$ and, in the case of LTE, also a key derivation function $KDF$. The functions $f_1, f_2$ are so called *message authentication functions*, and $f_3, f_4$ are so called *key generating functions*[2]. Moreover, $U$ maintains a counter $SQN_U$ and $H$ a counter $SQN_H$ for $U$.

A protocol run starts with $S$ sending a *user id request* and $U$ responding with its *IMSI*[3]. Next follows the *authentication data transfer*, in which $S$ sends an *authentication data request* to $H$, that consists of $U$'s *IMSI* and $S$'s identifier *SNid*, and $H$ answers with an *authentication data response*. $H$ chooses a

[2] We choose to do without the *anonymity key*, *i.e.* $f_5 \equiv 0$, which is an option in the specifications. We also omit the AMF constant.

[3] In fact, $U$ may alternatively respond with a temporary mobile subscriber identity (TMSI), which reduces but does not fully avoid the use of the *IMSI*.

fresh nonce $RAND$ and computes, with the key $K_0$ and its sequence number $SQN_H$, the so-called *message authentication code* $MAC$, the *expected response* $XRES$, the *cipher key* $CK$, the *integrity key* $IK$, and the *authentication token* $AUTN$ as depicted in Figure 1, where $\|$ denotes concatenation. The main difference between the UMTS AKA and LTE AKA is the *session key* $Skey$. In LTE AKA, the session key is computed over the identifier of $S$ (cf. caption of Figure 1). There is also the option that $H$ sends $S$ multiple *authentication vectors* $(RAND_i, AUTN_i, XRES_i, Skey_i)$ for $i = 1, \ldots, n$ at once in order to reduce the traffic between $S$ and $H$ but we will not focus on the use of this option.

In the *user authentication request*, $S$ forwards only $RAND$ and $AUTN$ to $U$. From the received $RAND$, $AUTN$, the user $U$ extracts $SQN_H$, computes the *expected message authentication code* $XMAC$ and compares it to $MAC$ contained in $AUTN$. If they are equal then $U$ performs a check on the sequence numbers $SQN_H$ and $SQN_U$[4]. If either of this two checks fail, then $U$ sends some error messages to $S$ (in fact, the error messages may be different, therefore allowing the linkability attack of [10]). Otherwise $U$ computes the *response* $RES$ and sends it to $S$. Finally, $S$ compares the response received from $U$ with the expected response received from $H$; if they are equal then the UMTS/LTE AKA run was successfully completed.

Intuitively, the UMTS/LTE AKA establishes the session key $Skey$ between $U$ and $S$, therefore, $Skey$ must satisfy some secrecy property. Furthermore, the protocol aims to authenticate $U$ to $S$. Both properties require $S$ to trust $H$ to provide a correct authentication data response. The sequence numbers allow to detect possible replays of authentication tokens. The UMTS/LTE AKA protocol, as depicted in Figure 1, does not offer authentication of $S$ to $U$. This known weakness has been described in [19]. User $U$ may at most know that $H$ generated the received nonce and authentication token for some service network.

Following the UMTS/LTE AKA, serving network $S$ and user $U$ need to negotiate the cryptographic algorithms (*security mode*) used to protect subsequent wireless communication between $S$ and $U$. Note that these algorithms are, in particular for inter-domain connections, not pre-determined. The messages of this negotiation are protected by (keys derived from) $Skey$. This is especially relevant for the case of LTE, where $Skey$ is generated over $S$'s identifier $SNid$. In LTE, by receiving the *NAS security mode command* directly following the user authentication response, $U$ should be able to authenticate $S$, as this message constitutes a key confirmation of the session key $K_{ASME}$. According to [7], the NAS security mode command from $S$ to $U$ has following form.

$$S \longrightarrow U : eKSI, UE\ sec\ capabilities, ciphering\ algo, integrity\ algo, NAS\text{-}MAC$$

where $NAS\text{-}MAC$ is a message authentication code under a key derived from $K_{ASME}$ over the rest of the message, which consists of non-secret components. We denote by *LTE AKA+1* the LTE AKA protocol together with this NAS security mode command message.

[4] Checking and increasing the sequence numbers can be different for UMTS and LTE

## 3 Attacking and Correcting UMTS & LTE AKA

Here we present a weakness found in the authentication protocol specifications of both UMTS and LTE AKA with the help of the tool CryptoVerif [12]. Although CryptoVerif has semantics in the computational model, the flaw in the protocols is of symbolic nature. Unlike other provers that work in the symbolic model, CryptoVerif does not output attack traces; instead we found the attack by interpreting the *last game* in a sequence of game transformations performed by CryptoVerif. It is the same flaw that is present in the specifications of both UMTS AKA and LTE AKA. Although UMTS AKA has previously been formally analyzed [10, 1], none of the previous analyses have detected this flaw. How GSM SIA is affected by the flaw is discussed in Section 3.3.

### 3.1 Communication Security Between *S* and *H*

It is obvious that the communication between $S$ and $H$ needs to be protected in some way, otherwise, *e.g.* if there is no confidentially protection, the exchanged session key(s) are sent in the clear. The specifications of the AKA protocols in [7] and [3] mention little about the security protection of the authentication data transfer.

However, for UMTS and LTE, the specifications [5, 6] detail the protection of IP-based communication between network elements. Here a distinction is made between inter-domain communication, where standardized solutions are necessary, and *intra-domain* communication, where the communicating parties are controlled by the same mobile operator. For inter-domain connections over IP-based networks, [5, 6] mandate the protection of the communication between network elements using IPsec. For intra-domain connections over IP-based networks (*i.e.* communication over $Z_b$ *interfaces*), [5, 6] state that the protection of communication is regarded as an internal issue of each domain operator. In particular, the use of IPsec for intra-domain communication between $S$ and $H$ is *optional*, even though the communication may involve long distance signaling.

Furthermore, in the case of UMTS, the communication between $S$ and $H$ can also be carried out on the *global Signaling System No. 7 network* instead of an IP-based network. The specification [4] details the protection for such communication between $S$ and $H$ using *Mobile Application Part security* (MAPsec). In comparison to IPsec, Mapsec protects messages on the application layer.

Both IPsec and MAPsec should, according to [5] and [4], offer following protection: *data integrity, data origin authentication, anti-replay protection*, and *confidentiality*. In addition, IPsec should offer *limited protection against traffic flow analysis*. Nonetheless, the attack presented below does not violate any of these properties. We found it while assuming that the messages sent between $S$ and $H$ are encrypted and then integrity protected through a message authentication code by long-term keys shared between S and H. The *encrypt-then-mac* scheme is indeed the principle used in both IPsec and MAPsec.

### 3.2 Session-mixup Attack against Authentication Data Response

We consider, as usual, an adversary who is in full control of the messages sent between instances of the roles of user $U$, serving network $S$, and home network $H$. We assume that $H$ acts as a trusted third party. When $S$ sends an authentication data request to $H$ for authentication parameters of $U$, the authentication data response by $H$ to $S$ is bound to $U$ as it includes message components that are generated under the long-term key shared between $H$ and $U$. However, $S$ cannot verify this (even though we assume authenticated encryption of the messages between $H$ and $S$), as $S$ does not know the key shared between the user equipments and $H$. There should certainly be some mechanism for $S$ to associate an authentication response to the correct $U$ if there is no attacker around. But as such a mechanism is not specified in the AKA protocols, we do not model them as part of the message parts that are protected by the authenticated encryption. We present a scenario in which an inside attacker may take advantage of this and we omit, due to space restrictions, the outside attack scenario[5].

**An *Inside* Attack** In this scenario we consider an attacker $A$ who is a subscriber $U$ of $H$. Say $U'$ is another subscriber of $H$ who is honest. If $A$ knows the $IMSI'$ of $U'$, which $A$ can learn either by listening on the network or by deploying a device called *imsi catcher*, then $A$ can execute the attack that is depicted in Figure 2 without $U'$ even being present. In this case, $A$ does not need to be able to intercept messages sent over the base stations. The attacker sends out two user identity responses: $IMSI'$ and his own subscriber identity $IMSI$. Then $S$ will run two concurrent AKA sessions, one for $U$ and one for $U'$, and sends two authentication data requests to $H$. When $H$ sends the authentication data responses for $S$ and $U$, then adversary $A$ redirects this message such that it is mistaken by $S$ as the response by $H$ for $S$ and $U'$ while he blocks the authentication data response that $H$ generated for $S$ and $U'$. Notice that this *session mixup* can be created by the attacker without breaking any cryptographic primitive and does generally not violate the specifications. Next the attacker redirects the messages sent by $S$ intended for $U'$ to $U$. So $U$ correctly receives the user authentication request containing message components that were generated by $H$ for $U$ (and $S$). Therefore, attacker $A$, who is registered as $U$, can generate the correct response and relay it to to $S$ such that $S$ believes that the response was generated by $U'$. The other session that $S$ opened for $U'$ is halted by $A$; it cannot be completed because $A$ does not know the keys that $U'$ shares with $H$. Anyhow, $A$ can impersonate $U'$ to $S$. Furthermore, the attacker and $S$ share a session key; it was in fact generated by $H$ for $U$ and $S$. At the same time, $S$ believes that this session key was generated by $H$ for $S$ and $U'$. Therefore, the attacker is able to execute subsequent communication steps and use the derived keys to use the wireless service provided by $S$ on behalf of $U'$. [6] $S$ will bill $H$ for the service that attacker $A$ received on $U'$ behalf, and $H$ will bill $U'$.

[5] which can, however, be easily derived from the inside attack.

[6] The attack is not fended off by the use of TMSIs. And the attacker's job is simplified in practice if multiple authentication vectors are sent at once.

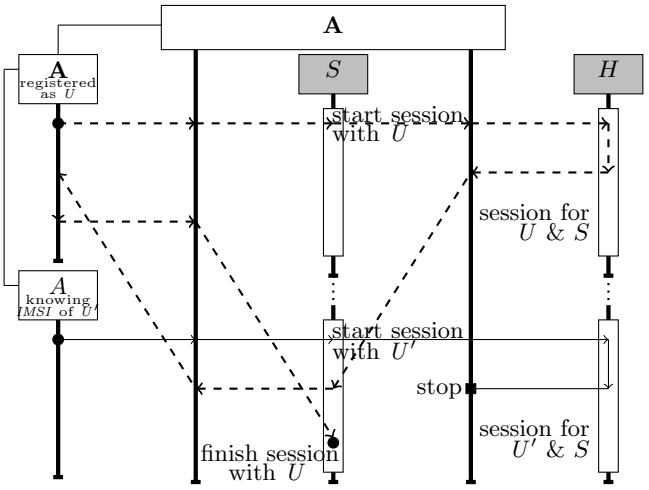


**Fig. 2.** Message flow of an *inside* attack against UMTS and LTE AKA (not showing the user id request). The attacker impersonates honest user $U'$ to $S$ and shares the session key(s) with $S$, without $U'$ being involved.

### 3.3 The GSM Subscriber Identity Authentication Protocol

The GSM SIA protocol [9, 8] is the 2G predecessor of UMTS AKA. It suffers from the same design flaw as UMTS and LTE AKA: there is no proper binding of the response sent by the home network (called *Authentication Vector Response*) to the corresponding request. Therefore, the attack of Figure 2 could also be deployed against GSM SIA. However, the case of GSM SIA is different. The specifications [9, 8] are only concerned about adversaries that attack the radio path, *i.e.* the connection between user equipment and base stations, while completely neglecting other connections. It does not violate the GSM specifications if there is no protection of the authentication vector response and the session key is transmitted in the clear by the home network. An attacker that is able to listen on the connections within the core network does not need to resort to the session-mixup attack to successfully violate GSM security as he can easily obtain the session keys. However, GSM operators that would like to protect the connection between home and serving networks, *e.g.* with MAPsec, need to be extremely careful so that message parts that prevent a session-mixup attack are sufficiently protected.

### 3.4 Possible Corrections

The UMTS/LTE AKA (and GSM SIA) protocols can easily be safeguarded if $S$ is enabled to determine, even under active attacks, for which user *IMSI* a response by $H$ was generated. We present two approaches to correcting the UMTS/LTE

AKA (from which the analogous corrections of the GSM ISA can be immediately derived).

The AKA protocol can be protected against active attackers when it is slightly modified by computing and adding a value $f(IMSI, X)$ to the authentication data response, where $f(.)$ is some function, which $S$ is able to compute and which satisfies some injectivity properties (*e.g.* $f$ may be a hash function), and $X$ some value known to $S$. Therefore, the authentication data transfer between $H$ and $S$ for $U$ is changed to

$$S \longrightarrow H : IMSI, SNid$$
$$H \longrightarrow S : f(IMSI, X), RAND, AUTN, XRES, Skey$$

We assume here that there is encryption and message authentication on all messages exchanged between $S$ and $H$. For instance one could choose $f(IMSI, X) \equiv IMSI$.

As an alternative fix, $S$ could generate a fresh request identifier, *e.g.* a nonce $n_S$ and include it in the authentication data request for $U$. The corresponding response must then include a function $g$ (computable by $S$ and with some injectivity properties) over this nonce and some other data $X$ known to $S$. In that case, the authentication data transfer should be modified to the challenge-response exchange

$$S \longrightarrow H : n_S, IMSI, SNid$$
$$H \longrightarrow S : g(n_S, X), RAND, AUTN, XRES, Skey,$$

where $n_S$ is a fresh nonce. For instance, $g$ could be the identity on nonces. Again, we assume that there is encryption and message authentication on all messages exchanged between $S$ and $H$.

### 3.5 Feasibility of Real-World Attacks under IPsec or MAPsec

Our attacks against the specifications of UMTS and LTE AKA work even if messages between $S$ and $H$ are encrypted as well as integrity protected by a message authentication code, which is what IPsec and MAPsec are doing. Although the UMTS and LTE AKA protocols are flawed, there are various scenarios in which real-world implementations of UMTS/LTE AKA could be immune to our attacks.

IPsec protects the TCP layer data. This alone does not prevent the attacks above (because IPsec would typically use the same session key for authentication data requests for both $U$ and $U'$). But if, in addition to using IPsec, $S$ uses different ports to send its requests to $H$, then the port numbers are appended to the sender/receiver addresses and become part of the protected TCP data. Therefore, they could be used by $S$ to assign the responses by $H$ correctly to each user. However, the specifications for UMTS and LTE do not detail how concurrent IPsec sessions are managed. We note that the AKA protocol is also likely run on top IPsec and other protocols, e.g. the diameter protocol [15]. If such

protocols handle sessions properly and the used session identifier are protected by IPsec, then the session-mixup attack is fended off.

Likewise, MAPsec can also be used in combination with a certain way of managing sessions that prevents our attacks. In [2], which is the *implementation (stage-3) specification* for MAP, the use of an *invoke ID* is mentioned that is part of the authentication data request and the corresponding response and needs to be *unique for each serving network*. If a serving network $S$ uses a separate invoke ID for each request, then $S$ could assign each response correctly to the corresponding request and our attacks would no work. But again, the specifications are not detailed enough on how the invoke id is used in concurrent sessions, and using the same invoke ID for several sessions is not ruled out.

Whether actual implementations of UMTS/LTE AKA follow the strategy of combining IPsec or MAPsec with using unique ports, invoke IDs or session IDs for concurrent authentication data requests is unknown to us. While this seems to be a very natural way to implement session handling with IPsec and MAPsec, it does not seem to be required by the specifications, and therefore some implementations of UMTS/LTE AKA may indeed be vulnerable in the real-world.

We question whether it is prudent practice to make the security of the UMTS/LTE AKA protocol implicitly reliant on a specific way how IPsec or MAPsec should be implemented, especially without stating it explicitly. Instead we believe that it would be preferable to strengthen the AKA protocols directly by making the binding of $H$'s authentication data response for an intended $U$ explicit in the protocol specifications. Notice also that, for intra-domain connections (and in the GSM case), operators can implement their proprietary solutions instead of using IPsec or MAPsec. Therefore, such systems may currently be vulnerable to our session mix-up attack as well, even if the implementations were guided by [3, 7]. So correcting the UMTS/ LTE AKA protocol directly will also better assist operators who wish to employ secure proprietary solutions for intra-domain connections.

## 4 Conclusions and Future Work

We present a security analysis of the UMTS and LTE AKA, in which we used the tool CryptoVerif to uncover a flaw in the specifications of UMTS and LTE AKA (and GSM SIA) with rather serious consequences. An inside attacker can authenticate as another honest subscriber to a serving network, and use the wireless services on his behalf. We suggest corrections to the protocols and, in ongoing work, we use the tool CryptoVerif to verify entity authentication and key secrecy properties for the corrected UMTS and LTE AKA protocols.

We believe that, even if real-world implementations of UMTS and LTE AKA that use IPsec or MAPsec happen to be immune against our attack, the uncovered flaw provides a valuable lesson to network domain operators who would like to protect their core networks' communication with proprietary solutions

(e.g., in the case of GSM, or for IP-based intra-domain connections in the case of UMTS and LTE).

For future work, we are interested in exploring, ideally in cooperation with 3GPP and mobile network operators, to what extend real-world systems are vulnerable to our attack. We would also like to expand our analysis scenarios of the protocol execution that are not covered in the present work, *e.g.* the scenarios that are related to the use of TMSIs and sequence numbers. Moreover, it would be interesting to obtain not only asymptotic security guarantees but also exact security guarantees with respect to the (fixed) key lengths used in the specifications of UMTS and LTE AKA.

**Acknowledgements** We thank Valtteri Niemi for helpful discussions.

## References

1. 3GPP TR 133.902. 3g security; formal analysis of the 3g authentication protocol. `http://www.3gpp.org/ftp/Specs/html-info/33902.htm`.
2. 3GPP TS 29.002. Digital cellular telecommunications system (phase 2+); universal mobile telecommunications system (umts); mobile application part (map) specification. `http://www.3gpp.org/ftp/Specs/html-info/29002.htm`.
3. 3GPP TS 33.102. Lte; 3g security; security architecture. `http://www.3gpp.org/ftp/Specs/html-info/33102.htm`.
4. 3GPP TS 33.200. 3g security; network domain security (nds); mobile application part (map) application layer security. `http://www.3gpp.org/ftp/Specs/html-info/33200.htm`.
5. 3GPP TS 33.210. Lte; 3g security; network domain security (nds); ip network layer security. `http://www.3gpp.org/ftp/Specs/html-info/33210.htm`.
6. 3GPP TS 33.310. Lte; network domain security (nds); authentication framework (af). `http://www.3gpp.org/ftp/Specs/html-info/33310.htm`.
7. 3GPP TS 33.401. Lte; 3gpp system architecture evolution (sae); security architecture. `http://www.3gpp.org/ftp/Specs/html-info/33401.htm`.
8. 3GPP TS 42.009. Digital cellular telecommunications system (phase 2+); security aspects. `http://www.3gpp.org/ftp/Specs/html-info/42009.htm`.
9. 3GPP TS 43.020. Digital cellular telecommunications system (phase 2+); security related network functions. `http://www.3gpp.org/ftp/Specs/html-info/43020.htm`.
10. M. Arapinis, L. I. Mancini, E. Ritter, and M. Ryan. Formal analysis of umts privacy. *CoRR*, abs/1109.2066, 2011. `http://arxiv.org/abs/1109.2066`.
11. B. Blanchet. An Efficient Cryptographic Protocol Verifier Based on Prolog Rules. In *14th IEEE Computer Security Foundations Workshop (CSFW-14)*, pages 82–96, Cape Breton, Nova Scotia, Canada, June 2001. IEEE Computer Society.
12. B. Blanchet. A Computationally Sound Mechanized Prover for Security Protocols. In *IEEE Symposium on Security and Privacy*, pages 140–154, May 2006.
13. B. Blanchet. A computationally sound mechanized prover for security protocols. *IEEE Transactions on Dependable and Secure Computing*, 2007. To appear. Technical report version available at `http://eprint.iacr.org/2005/401`.
14. D. Dolev and A. Yao. On the security of public-key protocols. *IEEE Trans. Info. Theory*, 2(29):198–208, 1983.

15. IETF. Diameter base protocol rfc 3588, September 2003. http://www.ietf.org/rfc/rfc3588.txt.
16. International Telecom Union. ICT indication database, 2011. http://www.itu.int/ITU-D/ict/statistics/.
17. U. Meyer and S. Wetzel. A man-in-the-middle attack on umts. In *Proceedings of the 3rd ACM workshop on Wireless security*, WiSe '04, pages 90–97, New York, NY, USA, 2004. ACM.
18. S. F. Mjølsnes and J.-K. Tsay. Computational security analysis of the umts and lte authentication and key agreement protocols. *CoRR*, abs/1203.3866, 2012.
19. M. Zhang and Y. Fang. Security analysis and enhancements of 3gpp authentication and key agreement protocol. *IEEE Transactions on Wireless Communications*, 4(2):734–742, 2005.