Journal of Universal Computer Science, vol. 19, no. 16 (2013), 2437-2452
submitted: 15/7/13, accepted: 27/9/13, appeared: 1/10/13 © *J.UCS*

Security Issues and Attacks on the GSM Standard: a Review[1]

Giuseppe Cattaneo
(Dipartimento di Informatica
Università di Salerno, I-84084, Fisciano (SA), Italy
cattaneo@dia.unisa.it)

Giancarlo De Maio
(Dipartimento di Informatica
Università di Salerno, I-84084, Fisciano (SA), Italy
demaio@dia.unisa.it)

Umberto Ferraro Petrillo
(Dipartimento di Scienze Statistiche
Università di Roma *"La Sapienza"*, I-00185, Roma, Italy
umberto.ferraro@uniroma1.it)

Abstract: The Global Systems for Mobile communications (GSM) is actually the most widespread mobile communication technology existing nowadays. Despite being a mature technology, its introduction dates back to the late eighties, it suffers from several security vulnerabilities, which have been targeted by many attacks aimed to break the underlying communication protocol. Most of these attacks focuses on the A5/1 algorithm used to protect over-the-air communication between the two parties of a phone call. This algorithm has been superseded by new and more secure algorithms. However, it is still in use in the GSM networks as a fallback option, thus still putting at risk the security of the GSM based conversations. The objective of this work is to review some of the most relevant results in this field and discuss their practical feasibility. To this end, we consider not only the contributions coming from the canonical scientific literature but also those that have been proposed in a more informal context, such as during hacker conferences.

Key Words: GSM, mobile security, security attacks, encryption
Category: C.2, C.2.1, C.2.0

1 Introduction

The GSM is actually the most widespread mobile communication technology, accounting for more than five billion subscriptions. Far from being just a personal communication technology, it has become the medium of choice for implementing and delivering a vast array of services ranging from mobile banking applications to electronic ticketing. This widespread use is also motivating the interest of researchers in evaluating the security mechanisms provided by GSM to protect user

[1] A preliminary short version of this paper appeared in [Cattaneo et al. 2013].

communication. In particular, the GSM protocols suffer from many weaknesses which allowed for the development of several attacks able to break confidentiality and privacy of subscribers. The objective of this paper is to review some of the most relevant security attacks to the GSM-related technologies, including also those techniques that, although not being presented in a formal scientific context, have proved to be very effective in practice.

1.1 Organization of the Paper

The rest of the paper is organized as follows. In Section 2, we briefly introduce the architecture of a GSM network with an emphasis on the security aspects. In Section 3, we discuss some of the most relevant attacks proposed so far in the scientific literature. In Section 4, we briefly outline the security issues existing in LTE networks. Finally, in Section 5, we draw some conclusions about the vulnerability of GSM communication networks with respect to the new communication technologies

2 The GSM Standard

The GSM has been developed by the ETSI as a standard [3GPP 1998] to describe protocols for second generation digital cellular networks used by mobile phones. It offers several services based on voice transmission and data transmission.

The main elements of a GSM network (see Figure 1), as described in [Wikipedia 2012], are:

– **The Mobile Station.** It is made up of the Mobile Equipment (ME) and of the Subscriber Identity Module (SIM). The ME refers to the physical phone itself, and it is uniquely identified by the International Mobile Equipment Identity (IMEI) number, burned into it by the manufacturer. The SIM is a small smart card that is inserted into the phone and carries information specific to the subscriber, such as International Mobile Subscriber Identity (IMSI), Temporary Mobile Subscriber Identity (TMSI) and the encryption keys (Ki and Kc).

– **The Core Network.** It carries out call switching and mobility management functions for mobile phones roaming on the network of base stations. It is made of several components. The Mobile Switching Center (MSC) is the primary service delivery node for GSM. It is responsible for setting up and releasing the end-to-end connections. Moreover, it manages mobility and hand-over requirements occurring during a call. Finally, it takes care of monitoring in real-time the cost of a call and charging it to the mobile phone subscriber. The Home Location Register (HLR) is a central database that

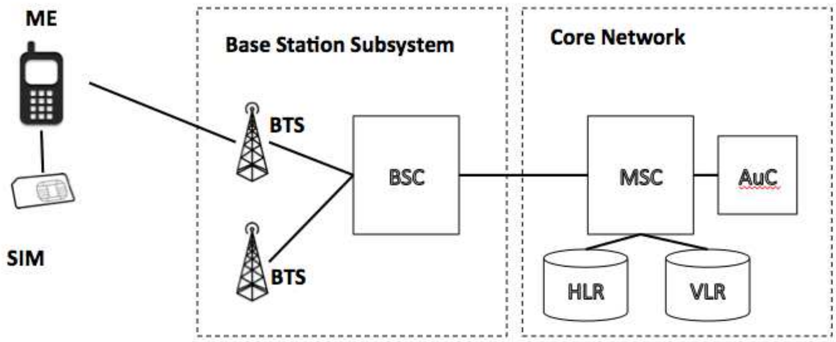

Figure 1: Structure of a GSM Network

contains details of each mobile phone subscriber that is authorized to use the GSM core network. The Visitor Location Register (VLR) is a database that contains the same type of information held by the HLR, but limited to the subscribers currently in a particular area. Finally, the Authentication Center (AuC) is the component responsible for generating the necessary cryptovariables for the authentication and the encryption on the network. The security of the entire process relies on a secret Ki shared between the AuC and the SIM. Ki is securely burned into the SIM during its manufacturing and is also securely replicated onto the AuC. Notice that Ki is never transmitted between the AuC and SIM, but is combined with the IMSI to produce a challenge/response for identification purposes.

– **The Base Station Subsystem.** It is responsible for handling traffic and signaling between a mobile station and the core network. In the Base Station Subsystem, the Base Transceiver Station (BTS) is the component that deals with the transmission and reception of radio signals with the mobile station. The coverage area of a BTS is called *cell*. The BTS contains the equipment for encrypting and decrypting communications with the mobile station and is controlled by a Base Station Controller (BSC).

2.1 Security Features

The GSM standard defines several security mechanisms for protecting both the integrity of the network and the privacy of the subscribers. Whenever a ME tries to join a GSM network, it has to pass through an authentication procedure required to verify the identity of the subscriber using it. This denies the possibility for a subscriber to impersonate another one and guarantees that only authorized subscribers may access the network. When connected, the signaling and data channels over the radio path between a base station and the ME are protected by means of an encryption scheme. This ensures the confidentiality of the conversations. In the following we provide more details about these schemes and about the cryptographic machinery they use.

These schemes do not require sensitive information to be transmitted over the radio channel. Instead, the authentication is performed using a a challenge-response mechanism. The conversations are encrypted using a temporary, randomly generated ciphering key (Kc). The mobile phone identifies itself by means of the Temporary Mobile Subscriber Identity (TMSI), which is issued by the network and may be changed periodically (i.e. during hand-offs) for additional security.

2.1.1 Authentication

As discussed in [Margrave 1995], the GSM network authenticates the identity of a subscriber using the following challenge-response mechanism (see Figure 2 for an overview of the involved security components).

1. The Authentication Center (AuC) generates a 128-bit random number ($RAND$) and sends it to the mobile phone.

2. The mobile phone computes the 32-bit signed response ($SRES$) based on the encryption of $RAND$ with the authentication algorithm ($A3$) using the individual subscriber authentication key (Ki). The computation is entirely done within the SIM. This provides enhanced security, because the confidential subscriber information such as the individual subscriber authentication key (Ki) is never released from the SIM during the process.

3. On the network, upon receiving the signed response ($SRES$) from the subscriber, the AuC compares its value of $SRES$ with the value it has received from the mobile phone. If the two values match, the authentication is successful and the subscriber joins the network. The AuC actually does not store a copy of $SRES$, in fact it has to query the HLR or the VLR in order to retrieve it, as needed. Otherwise, the connection is terminated and an authentication failure message is sent to the mobile phone. Also in this case, the individual subscriber authentication key (Ki) is never transmitted over the radio channel.

It is worth noting that GSM only authenticates the user to the network (and not vice versa). So, the security model offers confidentiality and authentication, but not the non-repudiation.

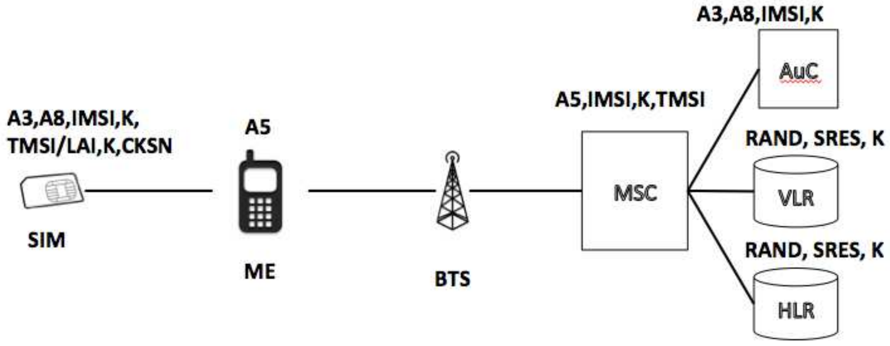

Figure 2: GSM Security Architecture

2.1.2 Data Confidentiality

The SIM contains the implementation of the key generation algorithm (A8) which is used to produce the 64-bit ciphering key (Kc) to be used to encrypt and decrypt the data between the ME and the base station. It is computed by applying the same random number ($RAND$) used in the authentication process to the ciphering key generating algorithm (A8) with the individual subscriber authentication key (Ki). Additional security is provided by the periodic change of the ciphering key. Similarly to the authentication process, the computation of the ciphering key (Kc) is done within the SIM. An additional level of security is provided by the periodic change of the ciphering key, making the system more resistant to eavesdropping. This change may occur at regular intervals as required by network design and security considerations.

Encrypted communications between the MS and the network is done using one of the A5 ciphering algorithms. Encrypted communication is initiated by a ciphering mode request command from the GSM network. Upon receipt of this command, the mobile station begins encryption and decryption of data using the selected ciphering algorithm and the ciphering key (Kc). The A5 algorithms are implemented in the hardware of the ME, as they have to encrypt and decrypt data on the fly.

2.1.3 The A5 Ciphering Algorithms

In the GSM protocol, the data is sent as sequence of frames, where each frame contains 228 bit. Each plaindata frame is XORed with a pseudorandom sequence generated by one of A5 stream cipher algorithms. These algorithms, namely A5/1, A5/2 and A5/3, are used for ensuring over-the-air voice privacy.

The A5/1 algorithm was developed in late 1987 and is used within Europe and United States. It takes as input a key of 64 bit and a public *initial vector* of 22 bit. The algorithm is based on three *linear feedback shift registers* (LFSR) long 19, 22 and 23 bit respectively. The keystream is built by running an algorithm, called *clock*, that produces 1 bit at each step. The output of the clock algorithm is the XOR of the leftmost bit of the three LFSR registers. Each register has associated a *clocking bit*. At each cycle, the clocking bits of the registers are given as input to a *majority function* that computes the *majority bit*. A register is *clocked* if the clocking bit agrees with the majority bit. Hence, at each step at least two or three registers are clocked, and each register steps with probability 3/4. This mechanism implies that each register generates a sequence which may be repeated not earlier than $2^l - 1$ clocks, where l is the length of the register. The initialization phase takes place by setting all registers to zero and, then, performing 100 cycles of clock. Then, the algorithm is ready to produce the keystream, one bit at time.

The A5/2 algorithm was introduced in 1989 in order to extend the GSM standard to a wider range of countries while complying with the cryptography-related export restrictions existing in the United States. It is a deliberate weakened version of the A5/1 which is almost identical to its counterpart except for an additional LFSR used to produce the three clocking bits. The output bit of this additional register is the XOR of the rightmost bits of the three other LFSRs and the three bits produced by the *majority functions* on each LFSR. Since 2007 A5/2 is not implemented anymore in mobile phones for security reasons.

Finally, the A5/3 algorithm was developed in 1997 and is based on the MISTY cipher [Matsui 1997]. In the 2002 it was modified in order to obtain a faster and more hardware-friendly version, called KASUMI [3GPP 1998]. In a few words, it is a block cipher using 64 bit blocks, 128 bit keys and a recursive Feistel structure with 8 rounds, each consisting of 3 rounds, where each round consists of 3 more rounds of nonlinear SBox operations.

3 Attacks

There is a wide category of attacks against mobile communications that do not depend on network weaknesses. These include mobile phones malware, identity theft by SIM cloning and so on. Some other attacks, such as phishing with SMS, may exploit human factors as well. A good review of such security issues can be found in [Castiglione et al. 2009]. On the contrary, this work focuses on attacks that exploit vulnerabilities of GSM protocols.

Most of these attacks target the A5 family of ciphering algorithms. The exact formulation of these algorithms is still officially secret. However, the research community has been able to recover it through a mix of reverse engineering and cryptoanalysis. Namely, the general design of A5/1 was leaked in 1994 and the first cryptanalysis of A5/1 has been performed by Golic [Golic 1997].

In this section we review some of the most interesting attacks proposed so far, distinguishing by passive and active attacks.

3.1 Passive Attacks

After the general design of A5/1 was leaked, several weakness of this algorithm have been exposed by the scientific community. In Table 1 we propose a resume of the passive attacks on the A5/1 algorithm considered in this paper. The first attack targeting the A5/1 algorithm has been proposed by Golic [Golic 1997], which introduced an effective Time-Memory Trade-Off (TMTO) attack based on the birthday paradox. This technique is applicable to any cryptosystem with a relatively small number of internal states like A5/1, which has 2^{64} states defined by three shift registers. The basic idea of the TMTO is to pre-compute a large set of states A, and to consider the set of states B through which the algorithm progresses during the generation of output bits. Any intersection between A and B allows the identification of an actual state of the algorithm. The proposed attack would be practicable only having 15 TB of pre-calculated data or three hours of known conversation, which is not very realistic [Biryukov et al. 2001].

Biryukov *et al.* presented two attacks based on a TMTO [Biryukov et al. 2001]. The first attack requires two minutes of known-conversation data and one second of processing time, while the second attack requires two seconds of plaintext data and several minutes of processing time. The amount of required storage varies from about 140 GB to 290 GB. Unfortunately, the execution time of the proposed attack grows exponentially with the decreasing of the input sequence. The attack exploits many weaknesses of A5/1, like the possibility of identifying states by prefixes of their output sequences, the ability to quickly retrieve the initial state of an intermediate frame and the possibility to extract the key from the initial state of any frame. The major drawback of this attack is that it requires a considerable amount of known-conversation data, which is in practice not always available.

Barkan *et al.* proposed an improvement of this technique in [Barkan et al. 2003]. Their main advancement of their method is the possibility to drop the unrealistic requirement about the availability of a plaintext of the conversation. The authors firstly describe a ciphertext-only attack on A5/2 that requires a few dozen milliseconds of encrypted off-the-air cellular conversation and finds the correct key in less than a second on a personal computer. Then, their proposal is extended to a more-complex ciphertext-only attack on A5/1 exploiting a weakness of the GSM protocol. In particular, the authors observed that error-correction codes are employed in GSM frames

Table 1: A brief resume of the performance of the main passive attacks on the A5/1 considered in this paper, as they were presented in their original formulation.

Attack	Attack time	Requirements
[Golic 1997]	Several hours	Three hours of known conversation or about 15 TB of pre-calculated data
[Ekdahl and Johansson 2003]	Less than five minutes	Few minutes of known conversation and few MBs of pre-calculated data
[Biryukov et al. 2001]	One second	Two minutes of known conversation and up to 240 GB of pre-calculated data
[Barkan et al. 2003]	Less than one second	A few dozen milliseconds of encrypted conversation and several TBs of pre-calculated data
[Nohl 2009]	In almost real-time	Two TBs of pre-calculated data

before encryption, which introduce a highly structured redundancy in the encrypted traffic.

The method adopted to retrieve the encryption key is computationally much faster than the one presented before, with it leading to the possibility of decrypting a conversation in almost real-time. The main drawback of the proposed attack is the very long time required for the pre-computation phase. For instance, with the hardware available at that time, 140 computers and 22 hard disks of 200 GB would have been necessary to calculate such data in one year, in order to decrypt conversations lasting at least 5 minutes.

A different strategy, based on a correlation attack, was introduced by Ekdahl *et al.* [Ekdahl and Johansson 2003]. The main advantage is that whereas TMTO attacks have a complexity which is exponential with the shift register length, here the complexity is almost independent from it. This attack exploits the weakness that the key and the frame counter are initialized in a linear fashion, which enables to separate the session key from the frame number in binary linear expressions. This allows to decrypt a conversation in less than 5 minutes, provided that few minutes of plaintext conversation are available. Moreover, the time and space requirements for the tables precomputation are much smaller than in previous attacks.

The attack by Ekdahl *et al.* has been further improved by Maximov *et al.* [Maximov et al. 2005], who exploited the weakness that some redundancy is part of the plaintext. In particular, they identified two kinds of redundancy: the first, introduced by Barkan *et al.* [Barkan et al. 2003], is due to the fact that coding of GSM frames is done before encryption, which results in linear relationships in the plaintext since the parity check symbols are also encrypted; the second is due to the fact that special frames, composed by a large number of zeros, are sent during silence periods. The resulting attack needs less then 1 minute of computation, and a few seconds of known conversation.

All the attacks presented so far had very high computational cost and/or were based on unrealistic assumptions. Instead, the first practicable attack, implementable by means of open-source software and commodity hardware, has been made public by Nohl in 2009 [Nohl 2009]. This work showed that A5/1 is vulnerable to generic pre-computation attacks. In fact, for a cipher with small key (64 bit in the case of A5/1), it is possible to construct a *code book*, i.e., a kind of table that provides a mapping between all possible ciphertexts and plaintexts. It can be exploited to perform a known-plaintext attack. For the case of A5/1, if an adequate number of plaintext/ciphertext couples are known, it is possible to recover the encryption key. In the case of GSM, a number of predetermined control messages can be leveraged as known plaintexts [Nohl 2010a].

Considering all the possible combinations, Nohl estimated that a code book for A5/1 would have been sized 128 Petabyte and would have taken more than 100,000 years to be computed on a standard PC. In their talk, Nohl and Paget revisited techniques for computing the code book faster and for storing it compressed. In substance, he proposed a tweaked A5/1 engine optimized for parallelization with CUDA technology, a parallel computing platform (see [NVIDIA Corporation, 2012] for more details) based on the usage of extremely optimized graphic processing units (GPUs). He estimated that using this technique a full code book for A5/1 can be computed in 3 months on 80 GPUs. Some tweaks presented in subsequent talks [Paget and Nohl 2009, Nohl 2010a] allowed to lower this boundary to 1 month on 4 ATI GPUs. Moreover, he proposed the use of a combined approach for data storage which makes use of distinguished point and rainbow tables [Lee and Hong 2012], by means of which it is possible to reduce the size of the code book to just 2 TB.

Nohl estimates that the attack has a 99% success rate when data from a phone registered to the network can be collected, which maximizes the amount of known control frames. Otherwise, the success rate drops to 50%, since only a small number of frames with known plaintext is available. In a subsequent talk, Nohl and Munaut performed a demonstration on how it is possible to find phones and decrypt their calls [Nohl 2010b]. Moreover, near real-time decryption has been hypothesized by means of a distributed cracking network.

As a result of these experiments, Nohl and Munaut were able to create a set of rainbow tables (i.e., precomputed tables for reversing cryptographic hash functions) for decrypting GSM conservations. These tables have been made public in 2010, along with an open source tool able to retrieve the key of an intercepted communication [Nohl 2010a]. Even if frequency hopping sequences on the GSM air interface are known, the main limit of this attack is that process raw data from the radio channel is challenging. In 2009, Nohl hypothesized some improvements to the available hardware and software equipment to reduce difficulty. In a subsequent talk, Nohl and Munaut performed a complete demonstration on how it is possible to find phones and decrypt their calls [Nohl 2010b].

The entire cracking process can be accomplished by means of the following open-source software:

– GnuRadio [GNU Software Foundation 2012] to record data from the radio channel

– Airprobe [Airprobe 2010] to parse GSM control data

– Kraken [Labs 2011] to crack the A5/1 session key based on the parsed data

– Airprobe again to decode voice data

Clearly, in order to capture radio data, a programmable radio equipment is needed, such as the Universal Software Radio Peripheral (USRP), essentially a computer-hosted software radio (see [Ettus Research, 2013] for more details).

User tracking is possible thanks to information leaked through the global SS7 network. Currently, there are a number of services available on the Internet which offer phone location lookup, such as [You Get Signal 2000]. They also show that even a reprogrammed cheap phone can be used to intercept a voice call. In particular, they used two Motorola C123 with a custom firmware (OsmocomBB [OsmocomBB Team 2012]) allowing for GSM packet sniffing. The first phone is in charge of recording control messages exchanged by the victim, while the second phone is aimed to hop on the same frequencies as the target phone in order to record the voice call. Intercepted data can be subsequently decrypted by cracking the encryption key used by A5/1, as mentioned before. This demonstration is the definitive proof that a complete and passive GSM call sniffing is feasible and can be accomplished by means of cheap hardware and open-source software.

The final result takes less than a minute and few seconds of known conversation. On the other hand the attack can be performed in few seconds (less than 1 minute) and it needs few seconds of known conversation. Moreover the time necessary for the tables precomputation has been significantly reduced along with the necessary memory needs. It should be noted that this last attack could be carried out event when no plaintext conversation is available. In this case, it is possible to use some redundancy bits usually existing in a standard GSM conversation as plaintext. There are at least two types of redundancy that can be used to this end. The first, suggested by Barkan *et al.* in [Barkan et al. 2003], consists of the parity-check symbols used to encode the GSM data frame before their encryption. The second traces back to the way the GSM protocol encodes the silence in the conversation in order to save communication traffic, as documented in [ETSI 2000]. In such a case, the mobile phone encrypts and sends to the base station a special frame, consisting of a large number of zeros, followed by two frames (of the same type) per second.

3.2 Active Attacks

There is a number of attacks against telecommunication networks which can be classified as active attacks. With respect to the passive attacks mentioned before, they exploit some design weaknesses of the telecommunication infrastructure which make possible to introduce a false mobile tower controlled by the attacker. The major security hole exploited by the fake tower, also called IMSI Catcher, is that the GSM specification only requires authentication of the handset to the network, but not authentication of the network to the handset. The IMSI Catcher acts between the victim mobile phone(s) and the real towers provided by the service provider, and it is able to both control communication parameters, like encryption algorithms, and eavesdrop traffic. Such an attack falls into the category of Man-In-The-Middle (MITM) attacks.

Some MITM attacks against GSM have been introduced in [Barkan et al. 2003]. They suppose that the victim is connected to a fake base station, which is able to intercept and forward the data sent by the phone to the network and vice versa. In order to perform authentication, the attacker connects to the network, which sends an authentication request to it. The attacker forwards the request to the victim, which computes SRES and returns it to the attacker. The attacker can now authenticate to the network by using SRES. Essentially, the attacker impersonates the network to the victim and the target phone to the network.

At this point, independently from the encryption algorithm chosen by the network, the attacker can request the victim to use a weak cipher like A5/2 (or even no encryption). Then, the attacker can employ cryptanalysis of A5/2 to retrieve the encryption key. It is worth noting that the key generation algorithm only depends on the RAND parameter specified by the network. As consequence, the encryption key used between the victim and the attacker is the same used between the attacker and the network, so that the attacker can decrypt all the traffic even if a secure encryption algorithm like A5/3 is requested by the network. The same attack can be also performed to decrypt GPRS traffic.

In 2000 a method for identifying a mobile phone user and for eavesdropping outgoing calls has been patented. It has been invalidated in 2012 by the Court of Appeal of England and Wales. In the meantime, a large number of IMSI catcher devices have been commercialized.

Paget and Nohl showed how it is possible to catch IMSI of a subscriber by means of an active attack [Paget and Nohl 2009]. Their attack makes use of a fake base station that could even be built from open source components, like OpenBTS, a 52 MHz clock, Asterisk, and a "cheap" Universal Software Radio Peripheral (USRP). The data can be collected and decoded by means of open source software like Wireshark.

In 2010 a practical attack to GSM has been presented by Paget [Paget 2010] using open source components. It exploits the vulnerability that the mobile phone connects to the strongest base station signal. Since the base station has full control over communication protocols, the handset can be instructed in order to use no traffic encryption (A5/0). In this way, the attacker can intercept all the traffic in plaintext. The equipment used for the demonstration has been an hacked IM-ME [Goodspeed 2010] and an USRP, connected to a laptop running OpenBTS and Asterisk. Since identifiers are well known, the fake BS can spoof any GSM network. Moreover, when in presence of a 3G (UMTS) signal, Paget shows that it is possible to force the victim to down to 2G by jamming the 3G frequencies. In this way, the attack introduced before can be performed again. A limit of this attack is that only outbound calls can be intercepted, since the phone results disconnected to the real network. The solution proposed by Paget is to perform a MITM attack, where the attacker also impersonates the victim telephone to the carrier. The attacker can negotiate the weakest cipher possible (A5/2 or A5/1) for traffic encryption, which can be subsequently cracked.

The UMTS standard introduced mutual authentication of the handset and the network in order to prevent man-in-the-middle attacks. It is done by the combination of two security mechanisms: the authentication token AUTN and the integrity protection of the security mode command message. The authentication token ensures the timeliness and origin of the authentication challenge, thus preventing replay of authentication data. The integrity protection prevents an attacker from fooling the handset into using a weak encryption scheme (or no encryption). Meyer and Wetzel [Meyer and Wetzel 2004] presented a scenario in which an attacker can impersonate a valid GSM base station with respect to an UMTS subscriber, even when UMTS authentication and key agreement are used. This attack requires that the victim phone supports both the GSM and the UTRAN radio interface. This requirement is still today met by most of commercialized handsets. The attack exploits the weakness that, unlike in standard

UMTS networks, in hybrid GSM/UMTS networks the security mode command is not integrity protected, since GSM does not support integrity protection. As consequence, the message can be easily forged by an attacker. The authors show that the attacker can retrieve the RAND and AUTN parameters from the network by just knowing the IMSI of the victim. Afterwards, it can impersonate the network and request a weak encryption (or no encryption at all) in order to retrieve the encryption key, as in the previous case.

4 LTE

Long-Term Evolution [3GPP 2013a] (LTE) is the upcoming 4G standard for wireless communication for mobile phones and terminals. It is based on the GSM/EDGE and UMTS/HSPA network technologies, with a different radio interface (orthogonal frequency-division multiplexing) and several improvements to the core network. The standard, developed by the 3GPP, aims to improve the speed of 3G networks up to 100Mbit/s downstream and 50Mbit/s upstream.

LTE provides a number of security improvements [3GPP 2011, 3GPP 2013b], mostly regarding the key-derivation function used in the AKA protocol and the encryption algorithms used to protect the communication. Furthermore, the integrity protection is mandatory for all the messages after and including the Security Mode Command, which enhances protection against MITM attacks.

Despite these improvements, some vulnerabilities have been recently pointed out by the research community. Tsay *et al.* [Tsay and Mjølsnes 2012] report a previously undetected flaw in the specifications of both UMTS AKA and LTE AKA, which may be exploited by both outside and inside attackers in order to break user authentication to the serving network. Inside attackers may impersonate the user and use wireless services on his behalf. Essentially, it is possible since the serving network, after contacting the home network for the processing of the user parameters, cannot verify that the response is really bound to the user itself.

Bassil *et al.* [Bassil et al. 2013] claim that LTE is vulnerable to signaling attacks. They show how a set of malicious users may take advantage of the signaling overhead required to setup and release dedicated bearers [3GPP 2013c] in order to overload the network.

5 Discussion and Final Remarks

The large number of attacks developed so far, although not always easy to be put in practice, seems to indicate that security should be a serious issue for GSM users. This is especially true for all those subjects who use this network to carry out confidential activities such as committing financial transactions or exchanging military-related information.

GSM carriers seem to have underestimated these threats, as witnessed by the several solutions for providing security to GSM-based communications (see, e.g., [Castiglione et al. 2011, Castiglione et al. 2012, De Santis et al. 2010, GSMK 2012, Huang et al. 2010, Huang 2011]) proposed in the scientific literature and/or available on the market.

Even though new generation mobile telecommunication systems, such as UMTS and LTE, introduce stronger algorithms for authentication, encryption and data integrity, their interoperability with GSM protocols makes these enhancements almost useless. In fact, as long as old protocols will be supported by the network, it will be not possible to avoid impersonation attacks which exploit inherent design weaknesses of GSM.

References

[3GPP 2011] 3rd Generation Partnership Project. “Comparison of LTE Security and UMTS Security”. (2011) `http://3gpphelp.blogspot.com/2011/10/comparison-of-lte-security-and-umts.html`.

[3GPP 2013a] 3rd Generation Partnership Project. “3GPP - LTE”. (2013) `http://www.3gpp.org/LTE`.

[3GPP 2013b] 3rd Generation Partnership Project. “3GPP System Architecture Evolution (SAE); Security Architecture”. (2013) `http://www.3gpp.org/ftp/specs/html-info/33401.htm`.

[3GPP 2013c] 3rd Generation Partnership Project. “Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3”. (2013) `http://www.3gpp.org/ftp/Specs/html-info/24301.htm`.

[3GPP 1998] 3rd Generation Partnership Project. “Technical Specifications for GSM Systems”. (1998) `http://www.3gpp.org/`.

[Airprobe 2010] Airprobe. (2010) `https://svn.berlin.ccc.de/projects/airprobe/`.

[Barkan et al. 2003] Barkan, E., Biham, E., Keller, N. (2003). “Instant Ciphertext-only Cryptanalysis of GSM Encrypted Communication”; Proc. Advances in Cryptology-CRYPTO 2003.

[Bassil et al. 2013] Bassil, R., Elhajj, I. H., Chehab, A., and Kayssi, A. (2013). “Effects of Signaling Attacks on LTE Networks”; Proc. *27th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, 2013, 499–504.

[Biryukov et al. 2001] Biryukov, A., Shamir, A., and Wagner, D. (2001). “Real-time Cryptanalysis of A5/1 on a PC”; Proc. *Fast Software Encryption*, Volume 1978 of *Lecture Notes in Computer Science*, 1–18. Springer.

[Castiglione et al. 2012] Castiglione, A., Cattaneo, G., Cembalo, M., De Santis, A., Faruolo, P., Petagna, F., and Ferraro Petrillo, U. (2012). “Engineering a Secure Mobile Messaging Framework”; *Computers & Security*, 31(6), 771–781.

[Castiglione et al. 2011] Castiglione, A., Cattaneo, G., De Maio, G., and Petagna, F. (2011). “SECR3T: Secure End-to-End Communication over 3G Telecommunication Networks”; Proc. *5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, 2011, 520–526.

[Castiglione et al. 2009] Castiglione, A., De Prisco, R., and De Santis, A. (2009). “Do you Trust your Phone?”; Proc. *10th International Conference on Electronic Commerce and Web Technologies (EC-WEB)*, 2009, Volume 5692 of *Lecture Notes in Computer Science*, 50–61. Springer.

[Cattaneo et al. 2013] Cattaneo, G., De Maio, G., Faruolo, P., and Ferraro Petrillo, U. (2013). "A Review of Security Attacks on the GSM Standard"; Proc. *Information and Communication Technology*, volume 7804 of *Lecture Notes in Computer Science*, 507–512. Springer.

[De Santis et al. 2010] De Santis, A., Castiglione, A., Cattaneo, G., Cembalo, M., Petagna, F., and Ferraro Petrillo, U. (2010). "An Extensible Framework for Efficient Secure SMS"; Proc. 4th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2010, 843–850.

[Ekdahl and Johansson 2003] Ekdahl, P. and Johansson, T. (2003). "Another Attack on A5/1"; *IEEE Transactions on Information Theory*, 49(1), 284–289.

[ETSI 2000] ETSI (2000). "Digital Cellular Telecommunications System (Phase 2+); Full rate speech; Comfort Noise Aspect For Full Rate Speech Traffic Channels"; Technical report, European Standard (Telecommunications series).

[Ettus Research, 2013] Ettus Research (2013).

[GNU Software Foundation 2012] GNU Software Foundation (2012). GNU Radio.

[Golic 1997] Golic, J. D. (1997). "Cryptanalysis of Alleged A5 Stream Cipher"; Proc. *EUROCRYPT*, 1997, Volume 1233 of *Lecture Notes in Computer Science*, 239–255. Springer.

[Goodspeed 2010] Goodspeed, T. (2010). Travis Goodspeed's Blog: IM ME GoodFET Wiring Tutorial. `http://travisgoodspeed.blogspot.it/2010/03/im-me-goodfet-wiring-tutorial.html`.

[GSMK 2012] GSMK (2012). Cryptophone. `http://www.cryptophone.de/`.

[Huang 2011] Huang, H. (2011). "Strongly Secure One Round Authenticated Key Exchange Protocol with Perfect Forward Security"; Proc. *Provable Security*, Volume 6980 of *Lecture Notes in Computer Science*, 389–397. Springer.

[Huang et al. 2010] Huang, Y.-F., Leu, F.-Y., and Wei, K.-C. (2010). "Constructing a Secure Point-To-Point Wireless Environments by Integrating Diffie-Hellman Pkds and Stream Ciphering"; Proc. *International Conference on Complex, Intelligent and Software Intensive Systems (CISIS)*, 2010, 384–390.

[Labs 2011] Labs, S. R. (2011). "Decrypting GSM Phone Calls". `https://srlabs.de/`.

[Lee and Hong 2012] Lee, G. W. and Hong, J. (2012). "A Comparison of Perfect Table Cryptanalytic Tradeoff Algorithms"; Cryptology ePrint Archive, Report 2012/540. `http://eprint.iacr.org/`.

[Margrave 1995] Margrave, D. (1995). "GSM Security and Encryption"; `http://www.hackcanada.com/blackcrawl/cell/gsm/gsm-secur/gsm-secur.html`.

[Matsui 1997] Matsui, M. (1997). "New Block Encryption Algorithm MISTY"; Proc. *Fast Software Encryption*,1997, Volume 1267 of *Lecture Notes in Computer Science*, 54–68. Springer.

[Maximov et al. 2005] Maximov, A., Johansson, T., and Babbage, S. (2005). "An Improved Correlation Attack on A5/1"; *Selected Areas in Cryptography*, 2005, Volume 3357 of *Lecture Notes in Computer Science*, 1–18. Springer.

[Meyer and Wetzel 2004] Meyer, U. and Wetzel, S. (2004). "A Man-in-the-Middle Attack on UMTS"; Proc. *3rd ACM workshop on Wireless security, WiSe*, 2004, 90–97, New York, NY, USA. ACM.

[Nohl 2009] Nohl, K. (2009). "Subverting the Security Base of GSM"; Hacking at Random.

[Nohl 2010a] Nohl, K. (2010a). "Attacking Phone Privacy"; Black Hat USA.

[Nohl 2010b] Nohl, K. (2010b). "Wideband GSM sniffing"; 27th Chaos Communication Congress.

[NVIDIA Corporation, 2012] NVIDIA Corporation (2012). Parallel Programming and Computing Platform: CUDA . `http://www.nvidia.com/object/cuda_home_new.html`.

[OsmocomBB Team 2012] OsmocomBB Team (2012). `http://bb.osmocom.org/trac/`.

[Paget 2010] Paget, C. (2010). "Practical Cellphone Spying"; DEF CON 18.

[Paget and Nohl 2009] Paget, C. and Nohl, K. (2009). "GSM: SRSLY?"; Proc. 26th Chaos Communication Congress.

[Tsay and Mjølsnes 2012] Tsay, J.-K. and Mjølsnes, S. F. (2012). "A Vulnerability in the UMTS and LTE Authentication and Key Agreement Protocols"; Proc. *6th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security: Computer Network Security (MMM-ACNS'12)*, 2012, Volume 7531 of *Lecture Notes in Computer Science*, 65–76. Springer.

[Wikipedia 2012] Wikipedia (2012). Network Switching Subsystem — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/Network_switching_subsystem`.

[You Get Signal 2000] You Get Signal (2000). "Phone Number Locator - Cell Phone Location with a Reverse Lookup and Google Maps"; `http://www.yougetsignal.com/`.