GSM Interception

21.11.1999

Lauri Pesonen
Department of Computer Science and Engineering
Helsinki University of Technology
Lauri.Pesonen@iki.fi

Abstract

The GSM standard was designed to be a secure mobile phone system with strong subscriber authentication and over-the-air transmission encryption. The security model and algorithms were developed in secrecy and were never published. Eventually some of the algorithms and specifications have leaked out. The algorithms have been studied since and critical errors have been found. Thus, after a closer look at the GSM standard, one can see that the security model is not all that good. An attacker can go through the security model or even around it, and attack other parts of a GSM network, instead of the actual phone call. Although the GSM standard was supposed to prevent phone cloning and over-the-air eavesdropping, both of these are possible with little additional work compared to the analog mobile phone systems and can be implemented through various attacks. One should not send anything confidential over a GSM network without additional encryption if the data is supposed to stay confidential.

Contents

---

1 Introduction

GSM is the most widely used cellular mobile phone system in the world with over 100 million GSM subscribers. GSM was one of the first digital mobile phone systems to follow the analog era. Widely known problems with GSM's analog counter parts were the possibility of phone fraud through cloning phones and thus calling in someone else's expense, and the possibility of someone intercepting the phone call over the air and eavesdropping on the discussion. The GSM system was supposed to correct these problems by implementing strong authentication between the MS and the MSC, as well as implementing strong data encryption for the over-the-air transmission channel between the MS and the BTS.

The GSM specifications were designed by the GSM Consortium in secrecy and were distributed only on a need-to-know basis to hardware and software manufacturers and to GSM network operators. The specifications were never exposed to the public, thus preventing the open science community around the world from studying the enclosed authentication and enciphering algorithms as well as the whole GSM security model. The GSM Consortium relied on Security by Obscurity, i.e. the algorithms would be harder to crack if they were not publicly available. According to the open scientific community, one of the basic requirements for secure cryptographic algorithms is that the security of the crypto system lies solely on the key. This is known as Kerckhoffs' assumption. The algorithm in question should be publicly available, so that the algorithm is exposed to the scrutiny of the public. According to the general opinion no single entity can employ enough experts to compete with the open scientific community in cryptanalysing an algorithm. Thus, the algorithms designed and implemented in secrecy will probably be somehow cryptographically weak and contain design faults. Eventually, the GSM algorithms leaked out and have been studied extensively ever since by the open scientific community. Interesting facts have been discovered since then, during the cryptanalysis of the A3, A5 and A8 algorithms.

In this paper I will attempt to introduce the GSM security model to the reader and explore all the vulnerable points in this model in order to show to the reader that these points can be attacked by an attacker, malicious network operator or a malicious user and that a GSM interception and phone call eavesdropping are possibilities in today's GSM systems regardless of the GSM Consortium's statements [3]. While exploring the different points of attack in the GSM system, I will also show possible

interception methods that can be implemented in order to eavesdrop on a GSM phone call. I will introduce six different types of attacks as well as a couple of variations of some of the attacks. Additionally I will try to shed some light on the gray areas of the field: what is and what is not possible with todays technology, and the current vulnerabilities in the GSM system. I will also make some suggestions how the security of a GSM network could be improved in the future in order to guarantee privacy for the GSM subscribers.

The rest of the paper is organized as follows:

- Chapter 2 introduces the acronyms and terms used throughout the paper.
- Chapter 3 introduces the GSM security model to the reader.
- Chapter 4 introduces the various attacks the GSM system is vulnerable against.
- Chapter 5 compares the GPRS security model to the GSM security model.
- Chapter 6 makes suggestions about possible improvements to the GSM security model in order to improve security.
- Chapter 7 concludes the discoveries made throughout the paper.

2 Definitions

A3	The authentication algorithm used in the GSM system. Currently the COMP128 algorithm is used as the A3/A8 implementation in most GSM networks.
A5	The encryption algorithm used in the GSM system. There are various implementations named A5/1, A5/2, ... The A5/1 is known as the strong over-the-air voice-privacy algorithm. A5/x (A5/2 ...) are weaker implementations targeted at foreign markets out side of Europe. There is also an A5/0 algorithm, which encloses no encryption at all.
A8	The key generation algorithm used in the GSM system. Currently COMP128 algorithm is used as the A3/A8 implementation in most GSM networks.
AuC	Authentication Center. The AuC register is used for security purposes. It provides the parameters needed for authentication and encryption functions (RAND, SRES and Kc). The RAND is a random challenge generated randomly. The other two parameters are generated from the RAND and the subscriber's Ki using the A3 and A8 algorithms. These parameters help to verify the user's identity (SRES) and provide the session key (Kc).
BSC	Base Station Controller. The BSC acts as a common node between multiple BTSs that together form one BSS and the backbone network.
BSS	Base Station Subsystem. The BSS connects the Mobile Station and the NSS. It is in charge of the transmission and reception. The BSS can be divided in two parts: • The Base Transceiver Station (BTS) or Base Station. • The Base Station Controller (BSC).
BTS	Base Transceiver Station, a base station the MS communicates with.
COMP128	A one-way function that is currently used in most GSM networks for A3 and A8. Unfortunately the COMP128 algorithm is broken so that it gives away information about its arguments when queried appropriately. This is an undesired and unacceptable side effect in a one-way function.

GPRS	General Packet Radio Service. GPRS is used to implement high speed data transmission between the MS and some other party. GPRS utilizes multiple BTSs in the same BSS. The MS sends different packets to different BTSs which are reconstructed at the SGSN. This enables the MS to use a higher transmission speed than one transmission channel can handle.
GSM	Global System for Mobile communications, a mobile phone system based on multiple radio cells (cellular mobile phone network).
HLR	Home Location Register. The HLR is part of the AuC. The HLR provides the MSC with triples specifying a random challenge and a SRES and a Kc based on the Ki of a specific subscriber and the random challenge. The HLR is also responsible for knowing the location of the MS at all times.
ISAAC	Internet Security, Applications, Authentication and Cryptography. A small research group in the Computer Science Division at the University of California, Berkeley. http://www.isaac.cs.berkeley.edu/
Kc	The secret session key used to encrypt over-the-air traffic between the BTS and the MS. The Kc is generated after every authentication initialized by the MSC. The Kc is calculated from the Ki and from the random challenge sent by the MSC with the A8 algorithm. The MS and the HLR both calculate the Kc independently of each other. The Kc is never transmitted over-the-air.
Ki	Ki is the secret key shared between the SIM and the HLR of the subscriber's home network.
LSB	Least Significant Bit.
LSFR	Linear Shift Feedback Register. A register that generates an output bit based on its previous state and a feedback polynomial.
MS	Mobile Station, the mobile phone.
MSC	Mobile services Switching Center, the central component of the NSS. The MSC performs the switching functions of the network. It also provides a connection to other networks.
NSS	Network and Switching Subsystem, its main role is to manage the communications between the mobile users and other users, such as mobile users, ISDN users, fixed telephony users, etc. It also includes data bases needed in order to store information about the subscribers and to manage their mobility.
SDA	The Smartcard Developers Association is a non-profit organization trying to provide developers non-proprietary information about smartcards. http://www.scard.org/
SGSN	Serving GPRS Support Node. A SGSN delivers packets to MSs within its service area through multiple BTSs. A SGSN also communicates with a HLR in order to authenticate MSs to enable encrypted communications. In GPRS the SGSN authenticates the MS instead of the MSC.
SIM	Subscriber Identity Module. The SIM identifies a subscriber. The subscriber can use multiple GSM phones with one SIM. All calls are charged on the same account and the subscriber's phone number stays the same. The SIM card contains IMSI, Ki and the A3 and A8 algorithms. The SIM is supposed to be tamper-proof, so that the Ki cannot be retrieved from it.

SRES	Signed RESponse. This is the response the MS returns to a challenge made by the MSC during the MS authentication thus authenticating itself to the MSC (or SGSN in the case of GPRS).
SS7	Signaling System 7 is used in most intelligent networks as a signalling protocol. SS7 is defined by ITU-T.
Symmetric Cryptography	In symmetric cryptography, the same key is used for both encryption and decryption.
VLR	Visitor Location Register. The VLR stores triples generated by the HLR when the subscriber is not in his home network. The VLR then provides the MSCs with these triples when necessary.

3 Introduction to the GSM Security Model

The GSM Security Model is based on a shared secret between the subscriber's home network's HLR and the subscriber's SIM. The shared secret, called Ki, is a 128-bit key used to generate a 32-bit signed response, called SRES, to a Random Challenge, called RAND, made by the MSC, and a 64-bit session key, called Kc, used for the encryption of the over-the-air channel. When a MS first signs on to a network, the HLR provides the MSC with five triples containing a RAND, a SRES to that particular RAND based on the Ki and a Kc based again on the same Ki. Each of the triples are used for one authentication of the specific MS. When all triples have been used the HLR provides a new set of five triples for the MSC [9].

When the MS first comes to the the area of a particular MSC, the MSC sends the Challenge of the first triple to the MS. The MS calculates a SRES with the A3 algorithm using the given Challenge and the Ki residing in the SIM. The MS then sends the SRES to the MSC, which can confirm that the SRES really corresponds to the Challenge sent by comparing the SRES from the MS and the SRES in the triple from the HLR. Thus, the MS has authenticated itself to the MSC. Figure 1.

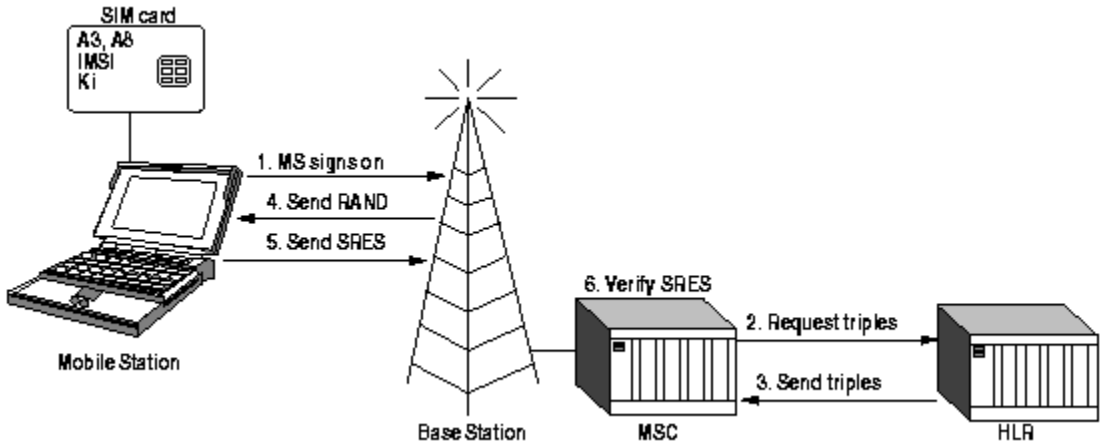

Figure 1, Mobile station authentication

The MS then generates a Session Key, Kc, with the A8 algorithm using, again, the Challenge from the MSC and the Ki from the SIM. The BTS, which is used to communicate with the MS, receives the same Kc from the MSC, which has received it in the triple from the HLR. Now the over-the-air communication channel between the BTS and MS can be encrypted.

Each frame in the over-the-air traffic is encrypted with a different keystream. This keystream is generated with the A5 algorithm. The A5 algorithm is initialized with the Kc and the number of the frame to be encrypted, thus generating a different keystream for every frame. This means that one call can be decrypted when the attacker knows the Kc and the frame numbers. The frame numbers are generated implicitly, which means that anybody can find out the frame number at hand. The same Kc is used as long as the MSC does not authenticate the MS again, in which case a new Kc is generated. In practice, the same Kc may be in use for days. The MS authentication is an optional procedure in the beginning of a call, but it is usually not performed. Thus, the Kc is not changed during calls. Figure 2.

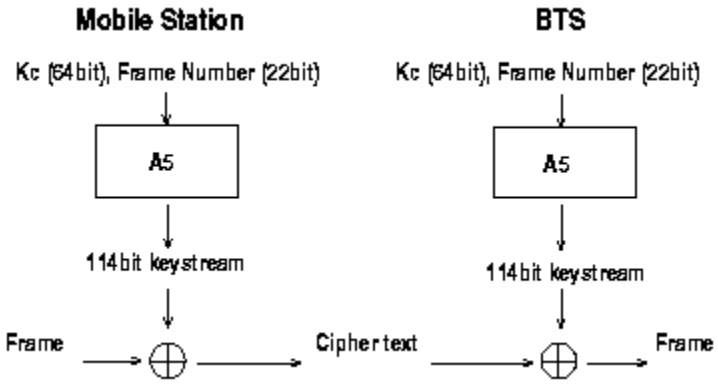

Figure 2, Frame encryption and decryption

Only the over-the-air traffic is encrypted in a GSM network. Once the frames have been received by the BTS, it decrypts them and send them in plaintext to the operator's backbone network. [9]

3.1 A3, The MS Authentication Algorithm

The A3 is the authentication algorithm in the GSM security model. Its function is to generate the SRES response to the MSC's random challenge, RAND, which the MSC has received from the HLR. The A3 algorithm gets the RAND from the MSC and the secret key Ki from the SIM as input and generates a 32-bit output, which is the SRES response. Both the RAND and the Ki secret are 128 bits long. Figure 3.

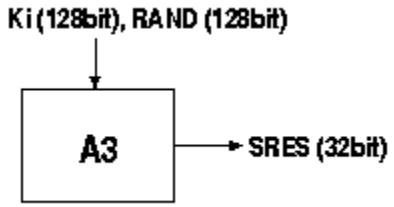

Figure 3, Signed response (SRES) calculation

Nearly every GSM operator in the world uses an algorithm called COMP128 for both A3 and A8 algorithms. COMP128 is the reference algorithm for the tasks pointed out by the GSM Consortium. Other algorithms have been named as well, but almost every operator uses the COMP128 except a couple of exceptions [2]. Figure 5.

The COMP128 takes the RAND and the Ki as input, but it generates 128 bits of output, instead of the 32-bit SRES. The first 32 bits of the 128 bits form the SRES response [6].

3.2 A8, The Voice-Privacy Key Generation Algorithm

The A8 algorithm is the key generation algorithm in the GSM security model. The A8 generates the session key, Kc, from the random challenge, RAND, received from the MSC and from the secret key Ki. The A8 algorithm takes the two 128-bit inputs and generates a 64-bit output from them. This output is the 64-bit session key Kc [6]. See Figure 4. The BTS received the same Kc from the MSC. HLR was able to generate the Kc, because the HLR knows both the RAND (the HLR generated it) and the secret key Ki, which it holds for all the GSM subscribers of this network operator. One session key, Kc, is used until the MSC decides to authenticate the MS again. This might take days.

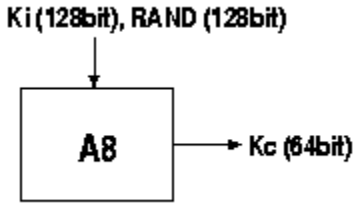

Figure 4, Session key (Kc) calculation

As stated in 3.1, COMP128 is used for both the A3 and A8 algorithms in most GSM networks. The COMP128 generates both the SRES response and the session key, Kc, on one run. The last 54 bits of the COMP128 output form the session key, Kc, until the MS is authenticated again. See Figure 5. Note that the key length at this point is 54 bits instead of 64 bits, which is the length of the key given as input to the A5 algorithm. Ten zero-bits are appended to the key generated by the COMP128 algorithm. Thus, we have a key of 64 bits with the last ten bits zeroed out. This effectively reduces the keyspace from 64 bits to 54 bits. This is done in all A8 implementations, including those that do not use COMP128 for key generation, and seems to be a deliberate feature of the A8 algorithm implementations [6].

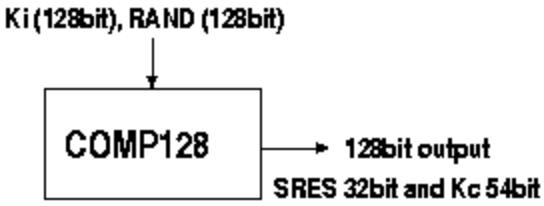

Figure 5, COMP128 calculation

Both the A3 and A8 algorithms are stored in the SIM in order to prevent people from tampering with them. This means that the operator can decide, which algorithms to use independently from hardware manufacturers and other network operators. The authentication works in other countries as well, because the local network asks the HLR of the subscriber's home network for the five triples. Thus, the local network does not have to know anything about the A3 and A8 algorithms used.

3.3 A5/1, The Strong Over-the-Air Voice-Privacy Algorithm

The A5 algorithm is the stream cipher used to encrypt over-the-air transmissions. The stream cipher is initialized all over again for every frame sent. The stream cipher is initialized with the session key, Kc, and the number of the frame being de/encrypted. The same Kc is used throughout the call, but the 22-bit frame number changes during the call, thus generating a unique keystream for every frame [1]. See Figure 6.

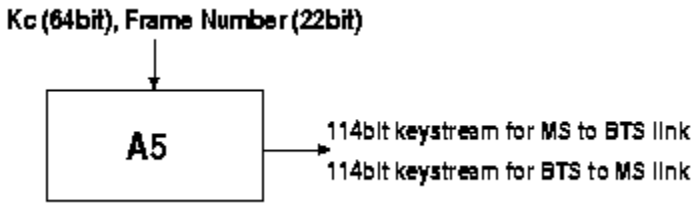

Figure 6, Keystream generation

The A5 algorithm used in European countries consists of three LSFRs of different lengths [11]. See Figure 7. The combined length of the three LSFRs is 64 bits. The outputs of the three registers are XORred together and the XOR represents one keystream bit. The LSFRs are 19, 22 and 23 bits long with sparse feedback polynomials. All three registers are clocked, based on the middle bit of the register. A register is clocked if its middle bit agrees with the majority value of the three middle bits. For example, if the middle bits of the three registers are 1, 1 and 0, the first two register are clocked or if the middle bits are 0, 1 and 0, then the first and third register are clocked. Thus, at least two registers are clocked on every round [8]. See Figure 8.

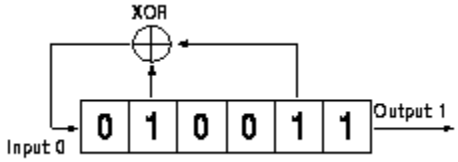

Figure 7, An example LSFR with feedback polynomial of $x^6 + x^4 + x$

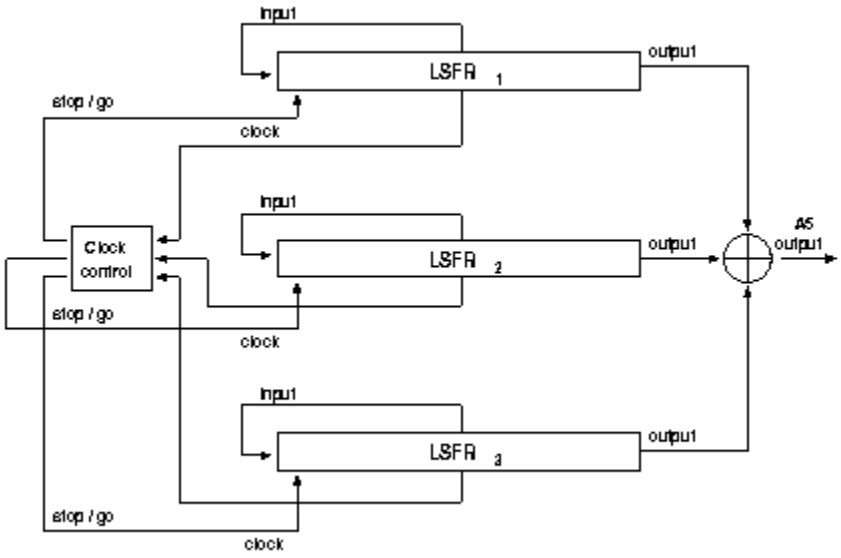

Figure 8, A5 LSFR construction

The three LSFRs are initialized with the session key, Kc, and the frame number. The 64-bit Kc is first loaded into the register bit by bit. The LSB of the key is XORred into each of the LSFRs. The registers are then all clocked (the majority clocking rule is disabled). All 64 bits of the key are loaded into the registers the same way. The 22-bit frame number is also loaded into the register in the same way except that the majority clocking rule applies from now on. After the registers have been initialized with the Kc and the current frame number, they are clocked one hundred times and the generated keystream bits are

discarded. This is done in order to mix the frame number and keying material together. Now 228 bits of keystream output are generated. The first 114 bits are used to encrypt the frame from MS to BTS and the next 114 bits are used to encrypt the frame from BTS to MS. After this, the A5 algorithm is initialized again with the same Kc and the number of the next frame [1][7].

Since the first GSM systems, other A5 algorithms have been designed and implemented. The main motivation has been that the original A5 encryption algorithm is too strong to export to the Middle East. Thus, the first 'original' A5 algorithm was renamed A5/1. Other algorithms include A5/0, which means no encryption at all, and A5/2, a weaker over-the-air privacy algorithm. Generally, the A5 algorithms after A5/1 have been named A5/x. Most of the A5/x algorithms are considerably weaker than the A5/1, which has the time complexity of 2^54 at most as, shown above. The estimated time complexity of A5/2 is as low as 2^16. This encryption is used in the USA. The other A5 implementations have not leaked. Thus, there are no real facts about them, just guesses and assumptions [2].

4 Possible Interception Attacks

The interesting question about the GSM security model is whether a call can be eavesdropped, now that at least one of the algorithms it depends on has been proven faulty.

Scientist around the world seem to be unanimous that the over-the-air interception and real time decoding of a call is still impossible regardless of the reduced key space [2]. But there seem to be other ways of attacking the system that are feasible and seem to be very real threats. There are also many attacks that are realistic, yet do not abuse any of the faults in the security algorithms.

4.1 Brute-Force Attack against A5

A real-time brute-force attack against the GSM security system is not feasible, as stated above. The time complexity of the attack is 2^54 (2^64 if the ten bits were not zeroed out). This requires too much time in order to be feasible in eavesdropping on GSM calls in real time. It might be possible to record the frames between the MS and the BTS and launch the attack afterwards though.

If we have a Pentium III class chip with approximately 20 million transistors and the implementation of one set of LSFRs (A5/1) would require about 2000 transistors, we would have a set of 10,000 parallel A5/1 implementations on one chip [10]. If the chip was clocked to 600 MHz and each A5 implementation would generate one output bit for each clock cycle and we would need to generate 100+114+114 output bits, we could try approximately 2M keys per second per A5/1 implementation. A keyspace of 2^54 keys would thus require about 900,000 seconds, 250 hours, with one chip. The attack can be optimized by giving up on a specific key after the first invalid keystream bit. This would cut the required time down by one third. The attack can also be distributed between multiple chips, thus drastically decreasing the time required [12].

4.2 Divide-and-Conquer Attack against A5

A divide-and-conquer attack manages to reduce the complexity from 2^54 of the brute-force attack to 2^45, which is a relatively dramatic change (2^9 = 512 times faster) [2]. The divide-and-conquer attack is based on a known-plain-text attack. The attacker tries to determine the initial states of the LSFRs from a known keystream sequence. The attacker needs to know 64 successive keystream bits that can be retrieved if the attacker knows some cipher text and the corresponding plain text [8]. This depends largely on the format of the GSM frames sent back and forth. The GSM frames contain a lot of constant information, e.g. frame headers. The required 64 bits might not always be known, but 32 to 48 bits are

usually known, sometimes even more [12]. Keep in mind that the attacker needs only one 64-bit plain text segment.

In short the divide-and-conquer attack is implemented by guessing the content of the two shorter LSFRs and then computing the third LSFR from the known keystream. This would be a 2^40 attack, if the clockings of the first two registers were not dependent on the third register. Because the middle bit of the third register is used for clocking, we have to guess about half of the bits in the third register between the clock bit and the LSB as well. This fact increases the time complexity from 2^40 to 2^45 [2].

However, J. Golic has proposed another divide-and-conquer attack based on the same assumptions with the average complexity of 2^40.16 [8]. Golic showed that only 2^62.32 internal states could be reached from the 2^64 initial states. Based on this assumption, he describes how to obtain linear equations by guessing n bits in the LSFRs. By solving these linear equations, one could recover the initial states of the three LSFRs. The complexity of solving the linear equations is 2^41.16. On average, one would resolve the internal state with 50 per cent chance in 2^40.16 operations.

Golic also proposed a Time-Memory Trade-Off Attack based on the Birthday Paradox in the same paper [8]. The objective of the attack is to recover the internal states of the three LSFRs at a known time for a known keystream sequence corresponding to a known frame number, thus reconstructing the session key, Kc.

4.3 Accessing the Signalling Network

As the two examples above clearly state, the A5 algorithm is not secure cryptographically, as there is another more feasible attack than the brute-force attack and it is not secure in practice either, because the brute-force attack in itself is not very hard to implement with current hardware. Yet, the algorithm is secure enough to prevent over-the-air call interception and real-time encryption cracking. Unfortunately, the air waves between the MS and the BTS are not the only vulnerable point in the GSM system.

As stated earlier, the transmissions are encrypted only between the MS and the BTS. After the BTS, the traffic is transmitted in plain text within the operators network [9][12].

This opens up new possibilities. If the attacker can access the operator's signaling network, he will be able to listen to everything that is transmitted, including the actual phone call as well as the RAND, SRES and Kc. The SS7 signaling network used in the operator's GSM network is completely insecure if the attacker gains direct access to it [12].

In another scenario, the attacker could attack the HLR of a particular network. If the attacker can access the HLR, he will be able to retrieve the Kis for all the subscribers of that particular network. Luckily the HLR is usually a bit more secure than the rest of the network, thus making it a slightly less probable point of entry, yet not completely improbable either keeping in mind the potential gain involved [12].

Accessing the signaling network is not very difficult. Although the BTSs are usually connected to the BSC through a cable, some of them are connected to the BSC through a microwave or even a satellite link. This link would be relatively easy to access with the right kind of equipment. Most of the commercially available equipment for GSM eavesdropping seem to use this particular vulnerability. Unfortunately I cannot to verify this, because the equipment and specifications are available only to law enforcement personnel and such. The microwave link might be encrypted, however, depending on the hardware manufacturer, thus making it slightly more difficult to monitor it [12]. It is really a question

about whether the attacker wants to crack the A5 encryption protecting the session of a specific MS or the encryption between the BTS and the BSC and gaining access to the backbone network. The possibility of accessing the cable leaving the BTS should not be ruled out either. This might be a very real threat and an attack could go undetected for a long time, if implemented carefully. The ability to tap on to the data transmitted between the BTS and BSC would enable the attacker to either monitor the call by eavesdropping on the channel throughout the call or he could retrieve the session key, Kc, by monitoring the channel, intercept the call over the air and decrypt it on the fly. Now that he knows the Kc, the real-time encryption is not a problem.

Another approach is through social engineering. This approach should not be underestimated although it sounds ludicrous. The attacker might pretend to be a repair man or such, enter a suitable building and install a wire tap. He might also bribe an engineer to do it for him or to give him all the Kis for all the subscribers of that particular operator. The possibilities are countless and real.

4.4 Retrieving the Key from the SIM

The security of the whole GSM security model is based on the secret Ki. If this key is compromised the whole account is compromised. Once the attacker is able to retrieve the Ki, he can not only listen to the subscribers calls, but also place calls billed to the original subscriber's account, because he can now impersonate the legitimate subscriber. The GSM network has trip wires for this: If two phones with the same ID are powered at the same time, the GSM network notices this, makes a location query for the phones, notices that the 'same' phone is in two different locations at the same time, and closes the account, thus preventing the attacker and the legitimate subscriber from placing calls [2]. But this is not relevant if the attacker is only interested in listening to the calls of the subscriber, as is assumed in this paper. In this case, the attacker can stay passive and just listen to the call, thus staying invisible to the GSM network.

The Smartcard Developer Association and the ISAAC security research group discovered a flaw in the COMP128 algorithm that effectively enabled them to retrieve the secret key, Ki, from a SIM [4][5]. The attack was performed on a SIM they had physical access to, but the same attack is applicable when launched over-the-air as well.

The attack is based on a chosen-challenge attack that works, because the COMP128 algorithm is broken in such a way that it reveals information about the Ki when the appropriate RANDs are given as arguments to the A8 algorithm. The SIM was accessed through a smartcard reader connected to a PC. The PC made about 150.000 challenges to the SIM and the SIM generated the SRES and the session key, Kc, based on the challenge and the secret key. The secret key could be deduced from the SRES responses through differential cryptanalysis. The smartcard reader used in implementing the attack could make 6.25 queries per second to the SIM card. So the attack required about eight hours to conduct. The results had to be analyzed as well, but this was apparently very quick, compared to the actual attack. Thus, the attacker needs to have physical access to the target SIM for at least eight hours. This is still very reasonable.

Again this vulnerability is also applicable in a social engineering scenario. One can assume that a corrupt GSM dealer would clone SIM cards in this way and then sell the cloned cards to third parties who wish to remain anonymous and do not want to buy legitimate SIMs. One could also try to sell a cloned SIM to a certain person in order to be able to eavesdrop on his calls later. A corrupt employee might also provide the attacker with the SIM card of the victim, so that the attacker can clone the SIM and later eavesdrop on the owner's calls. These are all very realistic scenarios in which the vulnerability found in the COMP128 algorithm compromises the whole security model of the GSM system, thus

leaving the subscribers in the open with no security at all.

4.5 Retrieving the Key from the SIM over the Air

The SDA and ISAAC researchers are confident that the same SIM-cloning attack could be launched over the air as well. Unfortunately, they can probably not confirm their suspicions, because the necessary equipment is illegal in the United States. The over-the-air attack is based on the fact that the MS is required to respond to every challenge made by the GSM network [4][12]. If the signal of the legitimate BTS is over powered by a rogue BTS of the attacker, the attacker can bomb the target MS with challenges and re-construct the secret key from these responses. Again the MS has to be available to the attacker over the air for the whole time it takes to conduct the attack. It is not known how long the attack would take when conducted over the air. Estimates vary from eight to thirteen hours [4].

The attack might be conducted in a subway, where the signal of the legitimate BTS is not available, but the phone is still turned on. The subscriber would be unaware of such an attack though the fact that the battery of the phone has run out slightly quicker than usual might make him suspicious. The attack can also be performed in parts: instead of performing an eight-hour attack, the attacker could tease the phone for twenty minutes every day on the victim's way to work. Once the SIM is cloned, the SIM-clone is usable until the subscriber gets a new SIM, which in practice does not happen very often.

In another scenario, the subscriber is on a business trip in another country. The attacker has somehow bullied the local GSM operator to perform this attack on the subscribers phone. The attacker would again be able to reconstruct the Ki based on the MS's SRES answers and the attack would probably go unnoticed, because the challenges originate from a legitimate network. Keep in mind that the local network does not know anything about the Ki, because the triples originate from the HLR of the subscribers home network. Thus, the local network has to deduce the Ki from the A3 responses.

4.6 Retrieving the Key from the AuC

The same attack used in retrieving the Ki from a SIM card can be used to retrieve the Ki from the AuC. The AuC has to answer to requests made by the GSM network and return valid triples to be used in MS authentication. The procedure is basicly identical to the procedure used in the MS to access the SIM card. The difference is that the AuC is a lot faster in processing requests than a SIM card is, because it needs to process a lot more requests compared to one SIM card. The security of the AuC plays a big role in whether this attack is possible or not and that is out of the scope of this paper [12].

4.7 Cracking the A8 Algorithm

Another possibility is that someone will be able to crack the A8 key generation algorithm and retrieve the secret key, Ki, based on the random challenge, RAND, the session key, Kc, and the SRES response (assuming the same algorithm is used for both A3 and A8 as is the case with COMP128) with a minimal amount of work. For example, the attacker may find a RAND that produces the Ki as a result (an over simplified example). All three variables are obtained relatively easily. The RAND and SRES are sent over the air in plain text and the session key Kc can be relatively easily deduced from the encrypted frames and the known plain text given enough time. A vulnerability like this in the key generation algorithm would of course devastate the whole GSM security model and give the GSM Consortium something to think about when designing their next security algorithms.

5 GPRS Security vs. GSM Security

In the GPRS system, the frames are transmitted as cipher text from the MS to the SGSN. This is done because the GPRS system uses multiple timeslots in parallel in order to achieve a greater transmission rate. One GPRS phone can be allocated multiple timeslots by the network, thus increasing the transmission rate of that MS. The frames can be sent in 'parallel' timeslots to the same BTS or to two different BTSs if the MS is handed over from one BTS to another [12].

To a BTS the use of one timeslot is seen as a separate call. Thus, the BTS is unable to put the frames from different timeslots together. This means that there has to be a network component that is able to receive the frames from one MS, defragment them and send them onwards to the actual destination. The BTSs are also unable to decrypt the frames, because consecutive frames on one channel have not got consecutive frame numbers. See Figure 9. To simplify the implementation, the frames are decrypted at the SGSN where all of the frames end up and it is thus easy to keep track of frame numbers. The solution is based on the ease of implementation and has not been implemented in order to increase system security. As a side effect, the GPRS system effectively prevents eavesdropping on the backbone between the BTS and SGSN, because the frames are still encrypted at this point. In GPRS, the triples from the HLR are transmitted to the SGSN and not to the MSC. Thus, security of GPRS depends largely on the placement and security of the SGSNs [12].

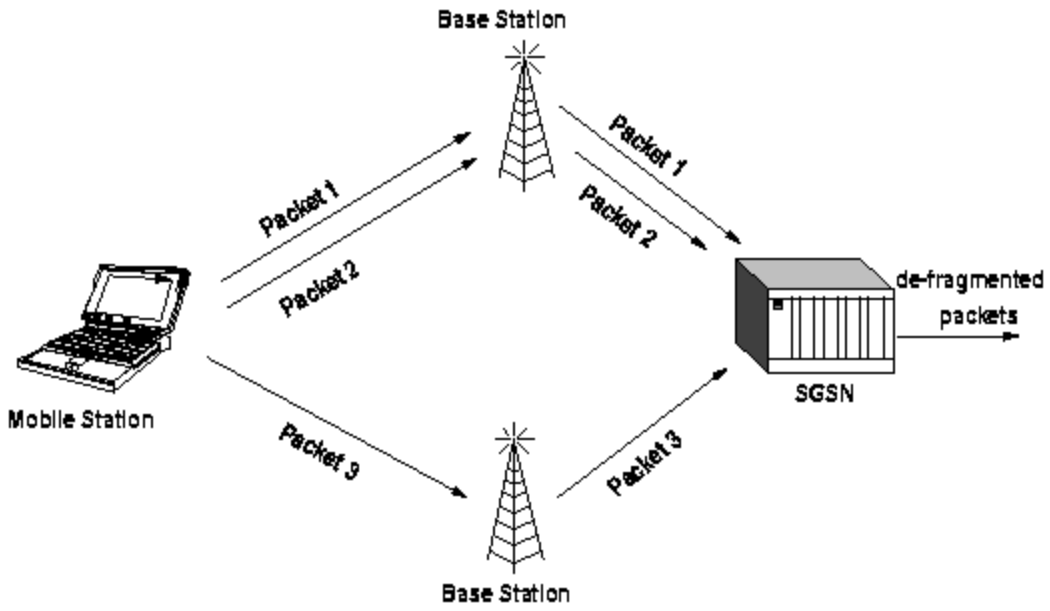

Figure 9, GPRS architecture

The GPRS system uses a new A5 implementation as well, which is not known publicly. This and the fact that the frames are not decrypted at the BTS, but at the SGSN, rules out a couple of attacks. First, it is very hard to attack the A5 implementation when it is not known. Secondly, the Kc is not transmitted to the BTSs and the transmission channel between the BTS and the SGSN is encrypted making it thus useless to monitor the backbone between the BTS and the SGSN. This does not mean that the GPRS security model would somehow be more secure than the GSM-only security model. It means that identical attacks do not work with GPRS that work with a GSM-only network. As soon as the A5 implementation used in GPRS leaks out, the GPRS security model is vulnerable to new attacks. And the implementation will leak out eventually or the design is successfully reverse-engineered. As was states above, the security of a crypto system should be based solely on the key. However the majority of the attacks against the GSM-only system are applicable against GPRS as well. E.g. the SIM-cloning attack. Additionally, the GPRS model introduces another security threat through the use of SGSNs, which

know the triples from the HLR. This means that the security of the GPRS network depends largely on the positions of the SGSNs in the network architecture and the security of the SGSNs. If the SGSNs are vulnerable to an attack, than the triples are vulnerable as well.

6 Possible Improvements

Security could be improved in some areas with relatively simple measures. The operator could use another cryptographically secure algorithm for A3. This would require issuing new SIM-cards to all subscribers and updating HLR software. This would effectively disable the attacker from cloning SIM-cards, the most dangerous attack, which is discussed above. This would also be the easiest improvement introduced here, because the network operator can make the changes itself and does not need the support of hardware or software manufacturers or the GSM Consortium.

Another solution would be to employ a new A5 implementation with strong encryption so that a brute-force attack is not feasible in any case. This would disable the attacker from recording transmitted frames and cracking them in his spare time. This improvement would require the cooperation of the GSM Consortium. The hardware and software manufacturers would have to release new versions of their software and hardware that would comprise with the new A5 algorithm.

Third solution would be to encrypt the traffic on the operators backbone network between the network components. This would disable the attacker from wiretapping the backbone network. This solution could probably also be implemented without the blessings of the GSM Consortium, but the cooperation of the hardware manufacturers would still be required.

In sum, none of the improvements above are too hard to implement. They all present new expenses mostly to the network operator and are not thus very attractive from the network operator's point of view. Thus, these improvements will probably not be implemented until the insecurity of the GSM networks becomes public knowledge and the network operators are forced to improve the security of the network. All three improvements would be necessary in order to secure the network against all attacks introduced in this paper.

7 Conclusions

The GSM security model is broken on many levels and is thus vulnerable to numerous attacks targeted at different parts of an operator's network.

Assuming that the security algorithms were not broken, the GSM architecture would still be vulnerable to attacks targeting the operator's backbone network or HLR and to various social engineering scenarios in which the attacker bribes an employee of the operator, etc.

Further more, the secretly designed security algorithms incorporated into the GSM system have been proven faulty. The A5 algorithm used for encrypting the over-the-air transmission channel is vulnerable against known-plain-text and divide-and-conquer attacks and the intentionally reduced key space is small enough to make a brute-force attack feasible as well. The COMP128 algorithm used in most GSM networks as the A3/A8 algorithm has been proved faulty so that the secret key Ki can be reverse-engineered over-the-air through a chosen challenge attack in approximately ten hours.

All this means that if somebody wants to intercept a GSM call, he can do so. It cannot be assumed that the GSM security model provides any kind of security against a dedicated attacker. The required

resources depend on the attack chosen. Thus, one should not rely solely on the GSM security model when transferring confidential data over the GSM network.

In addition to the possibility of call interception, the faulty COMP128 algorithm makes SIM cloning a threat as well, thus making it possible for an attacker to place calls at someone else's expense. This is not in the scope of this paper.

However, the reality is that although the GSM standard was supposed to correct the problems of phone fraud and call interception found in the analog mobile phone systems by using strong crypto for MS authentication and over-the-air traffic encryption, these promises were not kept. The current GSM standard and implementation enables both subscriber identity cloning and call interception. Although the implementation of cloning or call interception is a little bit more difficult, due to the digital technology that is used, compared to the analog counterparts, the threat is still very real, especially in cases where the transmitted data is valuable. Basically, we are where we used to be with the analog cell phones when it comes to security although the GSM Consortium tries to deny it [3].

References

[1] Anderson Ross, A5 - The GSM Encryption Algorithm, 17.6.1994, [referred 30.9.1999]
< http://chem.leeds.ac.uk/ICAMS/people/jon/a5.html >

[2] Anon., Crack A5, [referred 29.9.1999]
< http://jya.com/crack-a5.htm >

[3] Anon., GSM Alliance Clarifies False & Misleading Reports of Digital Phone Cloning, [referred 29.9.1999]
< http://jya.com/gsm042098.txt >

[4] Anon., GSM Cell phones Cloned, [referred 29.9.1999]
< http://jya.com/gsm-cloned.htm >

[5] Anon., GSM Cloning, [referred 24.10.1999]
< http://www.isaac.cs.berkeley.edu/isaac/gsm-faq.html >

[6] Briceno M. & Goldberg I. & Wagner D., An Implementation of the GSM A3A8 Algorithm, [referred 29.9.1999]
< http://www.scard.org/gsm/a3a8.txt >

[7] Briceno M. & Goldberg I. & Wagner D., A Pedagogical Implementation of A5/1, [referred 29.9.1999]
< http://www.scard.org/gsm/a51.html >

[8] Golic J. Dj., Cryptanalysis of Alleged A5 Stream Cipher, [referred 29.9.1999]
< http://jya.com/a5-hack.htm >

[9] Margrave David, GSM Security and Encryption, [referred 30.9.1999]
< http://www.net-security.sk/telekom/phreak/radiophone/gsm/gsm-secur/gsm-secur.html >

[10] Racal Research Ltd., GSM System Security Study, 10.6.1988, [referred 29.9.1999]
< http://jya.com/gsm061088.htm >

[11] Schneier B., Applied Cryptography, 2nd Ed., Wiley, New York, 1996, 758, [referred 29.9.1999]

[12] Kari H., Email Communication

Further Information

Shepherd S. J., Cryptanalysis of the GSM A5 Cipher Algorithm, IEE Colloquium on Security and

Cryptography Applications to Radio Systems, Digest No. 1994/141, Savoy Place, London, 3 June 1994

Shepherd S. J., An Approach to the Cryptanalysis of Mobile Stream Ciphers, IEE Colloquium on Security and Cryptography Applications to Radio Systems, Digest No. 1994/141, Savoy Place, London, 3 June 1994

> I assume both of the documents mentioned above would reveal a lot about the A5 stream cipher and its weaknesses. Unfortunately, the document has been declared classified as UK EYES ONLY for security reasons by the secret service of Great Britain.