Chapter 2
Security in Cellular Networks and Communications

Chuan-Kun Wu[1]

Abstract

Cellular Communication has become more and more important in our daily life. The objective of cellular communications has changed from mainly for voice communications as in many years ago, to that mainly for data transmission. The terminal devices for cellular communications also have many more functions other than the functionality for voice communication. Today most cellphones are also personal data assistances (PDAs) as well. Some advanced cellphones are like computers having many applications that used to be for computers. For example, they are able to access the Internet, through which users can conduct a variety of Internet transactions, download and upload data, enjoy on-line entertainment. This is particularly the case in 3G and later generation of networks which are targeted at high speed and wide bandwidth wireless communications. In order to enable sophisticated functionalities in a cellular phone terminal, an operating system is often needed. While a modern and future model of cellphone can give a lot of convenient services to our daily life, it also introduces many security threats, not only threatening the cellphone terminals, but also the cellular communications. This chapter tends to give a primary introduction of common security techniques in cellular communication networks. It is hard to predict what kind of security threats can be encountered in the future.

[1] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China, E-mail: ckeu@iie.ac.cn.

L. Chen et al., *Wireless Network Security*

Key Terms:

Cellular communications, Universal Mobile Telecommunications System, Authentication and Key Agreement (AKA), authentication vector, privacy protection.

2.1 Introduction

Today people are living in two worlds, a real world and a virtual world. The real world is getting virtually smaller due to the development of transport systems (cars, railways, and aircrafts), and the virtual world is also getting smaller due to the development of wireless communications, which enable people to be connected anywhere, anytime. One of the devices that make most of the contribution to the situation is the kind of cellphones, and the number of cellphones being used today has become very large, which is still growing[1]. Behind the cellphones which are terminal devices of wireless communications, it is the cellular communication systems and perhaps the Internet that connect people together.

A cellular network is a radio network with many fixed-location transceivers known as base stations distributed over land areas. Since the signal of each of the base stations covers only a limited area, there must be sufficient number of base stations in order to have a good signal coverage. In open land areas (without buildings), experiments show that hexagonal distribution of the base stations can achieve a good signal coverage with relatively fewer number of such base stations than other distributions. These hexagons look like cells geometrically (see Fig. 2.1), and hence such a network is called a

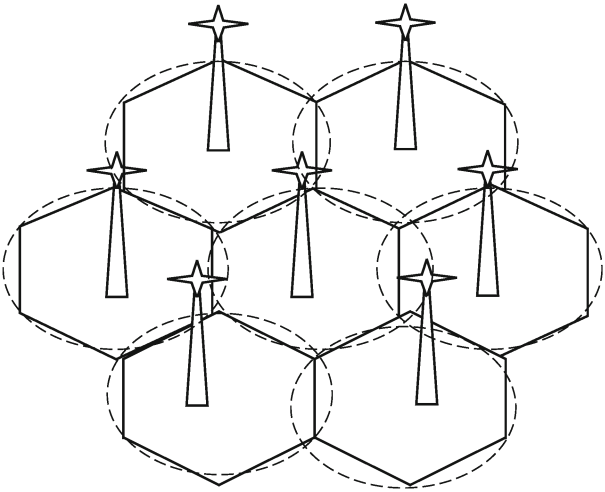

Fig. 2.1 Geometrical view of a cellular network.

cellular network.

One notable advantage of the cellular networks over the traditional ones is the mobility for the network users. With cellular networks, users with a mobile device (transreceiver) can move while the communication still holds. Since there is an overlap of the signals, the transreceivers are able to detect the signal of another base station and switch to the new base station before the signal from the connected base station vanishes. Technically when a mobile user moves from the coverage of one base station to another, the signal of the approaching station becomes stronger, while that of the previous serving base station becomes weaker. At certain stage, there is a signal switch from one base station to another, which may not even give any interruption to the transreceiver or even noticeable to the user. This process is called handover. It is also noted that with cellular networks, signal frequency can be reused, provided that neighboring cells use different ranges of signal frequencies.

The first generation of cellular communication was mostly for the purpose of wireless voice communications and small amount of text such as beeper, and the communication technique used was analogue signals. Just like the wired communications, no or little security techniques regarding the voice communication were considered at that time, because voice communication was treated as having little to do with commercial or other confidential information leakage.

The second generation (2G) of cellular communication systems emerged in the 1990's, primarily using the GSM (Global System for Mobile Communications) standard. There is a substantial difference between the first generation analog cellular communications and the second generation digital cellular communications. The use of digital technique enables many features that were not available in the first generation of cellular networks, including the following: (1) the communication has more robustness against noises, and the quality of voice can be ensured even for long distance communications by using the technique of error-correcting codes; (2) authentication and encryption services are available in the second generation cellular networks which ensure that the wireless communication is secure against "wire-tapping"; (3) short text service as a low-cost service has attracted much interest particularly from young people. Due to the above features of the second generation of cellular networks, and due to the advancement of electric and electronic technologies which make the cellphone terminals getting more and more handy, fancy, and with more and more of other applications. The number of users for the second generation cellular communications has grown to be very large. This figure is still growing, and it is reported that 300 million to 500 million new users are added to the total number of GSM users in the world each year in recent years.

The GSM employs some cryptographic techniques for user authentication and for data encryption. There have been some security weakness for the cryptographic algorithms and the authentication protocols revealed. Due to the increasing demand on data transmission with wireless networks, broadband wireless communications with broad applications are being developed, which leads to the 3rd generation cellular networks (3G) and the long term evolution (LTE) networks. Each of the new generation of cellular networks involve stronger security mechanisms and more sophisticated services.

2.2 Security architecture of cellular communication networks

2.2.1 The first generation of cellular communication networks

Because the first generation of cellular communications used analogue signal which is difficult to provide security services, and at the time when the first generation of cellular communication was in use, the security requirement was not so high, hence the security issues in cellular communications have been addressed only from the second generation of cellular communications with digitalized implementations. So, there is no security provision in the first generation of cellular communication networks.

In fact, in an analogue wireless communication system, an attacker could easily eavesdrop the communication of a cellular phone. A simple radio receiver that can cover the signal frequency for cellular communications can make the eavesdropping easily. There is no confidentiality of the communication data (voice). Moreover, it was technically not too difficult for an attacker to wiretap the identity of a cellphone, so that it is able to make a duplication of the cellphone, and then redirect all the call charges made from the duplicate phone to the owner of the original cellphone. Due to the small scale of the network and small number of cellphone users at that time, these kinds of attacks were not found to be serious threats. The demand for moving from the first generation of analogue communication networks to the second generation of digital ones is not only due to the security concern, but mostly due to the need of more digitalized services, such as text communication and other kinds of digital data exchanges.

2.2.2 The second generation of cellular communication networks

From now on, when we talk about the architecture of cellular communication networks, we mean digitalized communication networks, and they have to be second generation or a later generation of cellular communication networks unless specified otherwise.

Since today most cellular network users are mobile phone ones, where the mobile services cover far more than voice communications, the cellular communication networks are often called mobile networks, and the cellphones are often called mobile phones. Without confusion, in this chapter, by mobile phones and mobile networks we mean the cellphones and the cellular networks respectively.

In cellular communication networks, the transreceivers are also called user equipments (UEs), which are typically identified by a subscriber identity module (SIM, commonly known as SIM card), in combination with a cellphone (or a mobile phone, or a mobile device). A SIM card provides a tamper-proof environment for holding some secret information and execution of some security algorithms. The service providers, known as cellular network providers or mobile network providers, can be identified by two components, the home location register (HLR), which has an authentication center, and a visitor location register (VLR), which is composed of a collection of base stations. The HLR is responsible for issuing each mobile user a unique identity (ID), known as international mobile subscriber identity (IMSI), and a shared secret key. The IMSI and the secret key will be used for the subscriber to authenticate itself to the network. All this information is held by the authentication center in a secure database. On the user side, the information is kept in the SIM card. When a user tries to use the communication services during roaming, the user tries to reach a nearby base station first by sending the ID of the mobile user. The base station collects the information from the user, sends it to a processing unit, the VLR, which then communicates with the network authentication center residing in the HLR to authenticate the user. There can be a long distance communication between a base station and the authentication center, since this part of communication can go through wired networks, or even specific wires owned or hired by the network providers, where strong security techniques can be applied. The security threats during this part of communication are not a big concern for public research. Therefore, the security concerns in cellular communication networks are mainly in the air interface from a mobile user to a nearby base station.

In general, when talking about the security of cellular communications, the wired part of communication is treated as sufficiently secure, the authentication center is treated as being trustworthy, and the SIM card is treated as a tamper-proof hardware device. Although these assumptions are not unconditionally true, since chances for these assumptions to become false is very small, the assumption is widely acceptable. Since cellular communication networks are mostly for the use of mobile communications, we will also alternatively call them mobile communication networks in this chapter. It is noted that there are other kinds of mobile networks such as vehicular ad hoc networks, but this chapter only concerns with the cellular networks for mobile phone communications.

2.2.3 The third generation of cellular communication networks

The second generation of cellular communication networks seems to be able to provide most of the services we need. However, the problem concerned is not just about what kind of services are available. It is also about the quality of services. The prominent improvements of the 3rd generation of cellular communication networks over the 2nd generation ones include the improved security architecture (mutual authentication versus one-way authentication), improved security algorithms, and different radio frequency ranges providing larger communication bandwidth.

In a third generation of cellular communication network, there are three essential network components: the user equipment (UE), a home subscriber server (HSS) which has similar functionalities as an HLR in a 2G network, and many mobility management entities (MME), which have similar functionalities as the VLRs in a 2G network. The authentication process is also very similar to that of 2G networks, but a mutual authentication is enabled in 3G networks. This means that a network has to authenticate itself to the mobile users, apart from the users needing to authenticate themselves to the network.

2.2.4 The 3+ generation of wireless communication networks

The 3+ generation of wireless communication networks including long term evolution (LTE) networks (also known as 3.5G), WiMax (4G), WiFi (4G). Apart from LTE which is based on 3G network and hence has very similar architecture with that of 3G networks, other networks such as WiMax and WiFi have very different architectures and are not appropriate to be called cellular networks, and in this case their security problems are beyond the scope of this chapter and are not considered.

2.3 Security techniques in GSM networks

It is noted that the security threats become more serious with the scale of applications increases, i.e., under the same environment, a small scale application encounters fewer attacks than a larger scale application in the same environment. This is reasonable because on one hand, a larger scale application attracts more interests including those from attackers, hence more attacks may occur. On the other hand, launching an attack often involves some cost, so that the scale of an application is also a reason concerned with whether it is worth for the potential attackers to launch an attack.

It is known that the number of GSM users is far larger than that of those using other communication networks following the first generation in analogue signals, we will mainly focus on the security of GSM networks in this section.

2.3.1 User authentication in GSM

In GSM system, all the mobile users have to authenticate themselves to the network before the network can provide services. In the GSM networks, a mobile user denoted as user equipment (UE) which includes a compatible mobile device (e.g. a cell phone) and a SIM card, get their unique ID from a mobile network authentication center, which is located in or collaboratively working within a home location register (HLR). When a mobile user connects to the network, a nearby base station is contacted, which is managed by a visitor location register (VLR, often multiple base stations are being managed by a same VLR). The Authentication and Key Agreement (AKA) protocol for a GSM network can be depicted in Fig. 2.2.

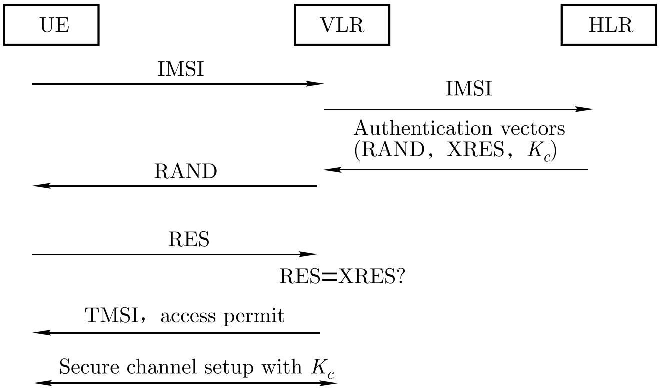

Fig. 2.2 Authentication and key agreement (AKA) process in GSM.

From Fig. 2.2, it can be seen that, when a mobile user tries to connect to the mobile networks, it first sends its IMSI serving as its identity to a nearby base station, which collects information and transfers to the VLR. The VLR is able to find which HLR the IMSI user belongs to, and sends the IMSI to the corresponding HLR. The HLR also serves as an authentication center and after checking the user as valid, creates a number of authentication vectors, and sends the authentication vectors to the VLR. In each of the authentication vectors, there are three components, they are a 128-bit random number RAND, a 32-bit expected response XRES=A3 (K_i, RAND), and a 64-bit data encryption key K_c=A8 (K_i, RAND), where A3 and A8 are two standard encryption algorithms in the GSM system, and K_i is the user key shared by the mobile user and the authentication center. When the VLR receives the authentication vectors, it chooses one of the authentication vectors, sends the RAND in the authentication vector to the mobile user. When the mobile user receives the RAND, its SIM card has the user key K_i and the encryption algorithms A3 and A8, and is able to create RES=A3 (K_i, RAND), and sends RES back to the VLR. The VLR then compares the RES received from the mobile user with the XRES as in the authentication vector received from the HLR. If the user is a valid one, then the equality of RES=XRES should hold, and hence the authentication is passed. In this case, the VLR generates a temporary mobile subscriber identity (TMSI), and has setup a secure communication channel with the mobile user under the protection of a common session key K_c, because the mobile user is also able to compute K_c. Then the data transmission between the mobile user and the contacting base station will be protected by an encryption algorithm named as A5. The use of TMSI is to provide privacy protection of the IMSI to certain degree. This issue will be further discussed later.

2.3.2 The authentication algorithms A3 and A8

The algorithms of A3 used for user/equipment authentication and the algorithm A8 used for generating a session key K_c are all based on a cryptographic algorithm named as COMP128, while the algorithm A5 used for data encryption is a stream cipher. The COMP128 is a keyed hash function that takes a 128-bit key and a 128-bit random number as input, and generates a 96-bit hash code. The 96-bit output then is split into a 32-bit XRES and a 64-bit K_c, as shown in Fig. 2.3.

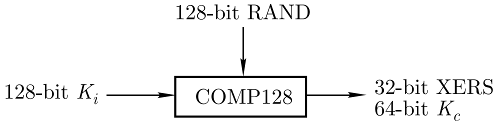

Fig. 2.3 The generation of XRES and K_c using COMP128.

The COMP128 works as follows: first it loads a 128-bit key and a 128-bit RAND by concatenation into a 32-byte array, and then a compression function is called for 5 times, mainly functioning at the positions where the 128-bit key is loaded.

The algorithm COMP128 was meant to be an industry standard yet to remain secret to the public. However with partial information being accidentally released via the Internet, the Smartcard Developer Association (SDA) and two U.C. Berkeley researchers, Ian Goldberg and David Wagner, jointly broke the COMP128 algorithm which leads the cloning of SIM cards possible. They demonstrated that the A8 algorithm takes a 64-bit key, but ten key bits were set to zero. The attack on the A8 algorithm demonstrated by Goldberg and Wagner takes just 2^{19} queries to the GSM SIM, which takes roughly 8 hours. Later it was shown by Josyula et al. of IBM that COMP128 can be broken in less than a minute.

Noticed that, even though the COMP128 algorithm was shown to be insecure, the GSM system had to provide services for even increasing number of users, and hence the algorithm cannot be abandoned. In order to provide a better security for later GSM users, some modifications for the COMP128 were made, which leaded to COMP128 version 2 and version 3. The broken of COMP128 algorithm (version 1) is a typical example to show that "security by obscurity" is not a good practice to provide security. The COMP128-2 and COMP128-3 are also secret algorithms which have not been subject to cryptanalysis. COMP128-3 fixes the problem of COMP128-1 where 10 bits of the Session Key (K_c) were set to zero.

It should be noted that the algorithms A3 and A5 are used within a specific network and their subscribers. In a different network, the A3 and A5 algorithm can well be different. So there is no need for global standardization of them, although they have been globally standardized. A good effect of using non-standard algorithms is that upgrading of the algorithms can be done relatively easily, and security flaws revealed in one network may not be a threat to another network due to the use of a different set of algorithms.

2.3.3 The data encryption algorithms A5

In the GSM system, the A5 algorithm is very different from the A3 and A8 algorithms. It is a stream cipher used to provide confidentiality for messages in over-the-air transmission. It is different also in the sense that the A5 algorithm has to be standardized globally so that services during worldwide roaming can be provided.

The A5 algorithm also has three versions, where the first two versions were meant to be kept secret from the public. However the general design was leaked in 1994, and the algorithms were entirely reverse engineered in 1999 by Marc Briceno, and security analysis on the algorithms became public since then. The first version of the algorithm, named as A5/1 or the original A5 algorithm was developed in 1987 mainly for GSM users in Europe. However for the purposes of export, the second version of the algorithm, named as A5/2, was developed with the security being deliberately weakened, and was used in the United States. It is probably to the surprise of the designers how weak the A5/2 algorithm is, which could even affect the security of the systems using the A5/1 algorithm by default.

The A5/1 stream cipher algorithm uses three linear feedback shift registers (LFSRs) over the binary field GF(2) of lengths 19, 22 and 23 respectively, with feedback polynomials being $x^{19} + x^{18} + x^{17} + x^{14} + 1$, $x^{22} + x^{21} + 1$ and $x^{23} + x^{22} + x^{21} + x^{8} + 1$ respectively. All the three LFSRs clock irregularly in a stop/go fashion. More precisely, there is a clocking bit for each of the three registers. At each clock cycle, a register is clocked unless its clocking bit does not agree with either of the clocking bits for the other two registers. This non-clocking happens when the clocking bit of the current register is 1 (0) when the other two clocking bits are both 0 (1), and happens with probability 1/4. The final output bit of the A5/1 algorithm is the exclusive-or of the output bits of the three registers.

2.3.4 The security weakness of the algorithms A5

A number of different attacks have been found soon after the A5/1 algorithm being revealed to the public, while the attacks on the A5/2 algorithm are more efficient. In 1999, Ian Goldberg[2] cryptanalyzed A5/2 in the same month when it was made public, and showed that it was so weak that can be broken in real time.

In 2003, Barkan et al.[3], and Ekdahl and Johansson[4] published their attacks on A5/1 algorithm, and a more efficient attack was given in[5] which can break A5/1 in real time, or at any later time. In fact, Barkan's approach was not directly to attack the A5/1 algorithm, it is to use the efficient attack on A5/2 algorithm to trigger the encryption key used by the A5/1 algorithm since both of the algorithms are assumed to use a same encryption key. Barkan's attack can be depicted in Fig. 2.4.

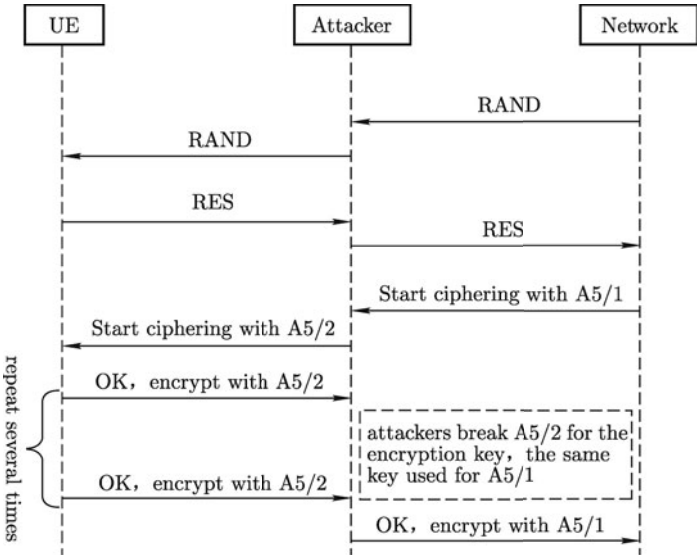
UE
Attacker
Network
RAND
RAND
RES
RES
Start ciphering with A5/1
Start ciphering with A5/2
OK, encrypt with A5/2
attackers break A5/2 for the encryption key, the same key used for A5/1
OK, encrypt with A5/2
OK, encrypt with A5/1
repeat several times

Fig. 2.4 Attacking on A5/1 via attacking on A5/2.

Alternatively, the attacker can record the RAND, the RES, and the immediate conversation, and later on use the same RAND and RES to forge another AKA process with the network claiming to use the A5/2 algorithm.

Once the A5/2 is broken, the key must be the same as what was used in the previous recorded conversation so that it can be decrypted.

2.3.5 The algorithms A5/3: a complete new version

To overcome the weakness of the A5/1 algorithm (the A5/2 was abandoned due to its security being too weak), the third version of the algorithm, named as A5/3, was introduced in the GSM system. The A5/3 changed the philosophy of "security by obscurity". Instead it used an algorithm KASUMI[6], with some small modifications for easier hardware implementation and to meet other requirements for 3G mobile communications security. A5/3 is a strong encryption algorithm created as part of the 3rd Generation Partnership Project (3GPP).

Different from A5/1 and A5/2, the KASUMI algorithm used by A5/3 is a block cipher. The KASUMI algorithm can be described briefly as follows: given a 128-bit key and a 64-bit message as its inputs, the message is split into a 32-bit left half and a 32-bit right half, denoted as $m = L_0||R_0$. Then the KASUMI processes the message in 8 rounds of iteration, and finally outputs $L_8||R_8$.

As in many other block ciphers, an initial key is used to produce many round-keys. In KASUMI, each round of encryption needs three round-keys KL_i, KO_i and KI_i. Write the initial 128-bit key as

$$K = K_1||K_2||K_3||K_4||K_5||K_6||K_7||K_8,$$

each K_i is a 16-bit string. Define

$$K' = K \oplus C = K'_1||K'_2||K'_3||K'_4||K'_5||K'_6||K'_7||K'_8,$$

where $C = 0 \times 123456789\text{ABCDEFFEDCBA}9876543210$. Then the round-keys are as follows:

$$\begin{aligned}
KL_{i,1} &= K_i \lll 1\\
KL_{i,2} &= K'_{i+2}\\
KO_{i,1} &= K_{i+1} \lll 5\\
KO_{i,2} &= K_{i+5} \lll 8\\
KO_{i,3} &= K_{i+6} \lll 13\\
KI_{i,1} &= K'_{i+4}\\
KI_{i,2} &= K'_{i+3}\\
KI_{i,3} &= K'_{i+7}
\end{aligned}$$

where "$X \lll j$" means cyclic shift of X to the left by j bits, and the subscript index should take the modulo 8 value.

There are three core functions in the KASUMI, they are named as FL, FO, and FI. The function FL in round i, denoted as FL_i, is defined as taking the 32-bit round-key $KL_i = KL_{i,1}||KL_{i,2}$ and a 32-bit data $I = L||R$ as inputs, and produces a 32-bit output. More precisely, first do the bitwise AND operation for L and $KL_{i,1}$, then the result performs a cyclic shift to the left by one bit, then the result is XOR'ed with R to get the right half of the output R': $R' = ((L \wedge KL_{i,1}) \lll 1) \oplus R$. Then do the bitwise OR operation for R' and $KL_{i,2}$, perform a cyclic shift of the result to the left by one bit, and then the result is XOR'ed with L to get the left half of the output $L' = ((R' \wedge KL_{i,2}) \lll 1) \oplus L$. Finally the output of the function $FL_i(KL_i, I)$ is $I' = L'||R'$.

The function FO in round i, denoted as FO_i, is defined as taking as input a 32-bit data I, a 48-bit round-key $KO_i = KO_{i,1}||KO_{i,2}||KO_{i,3}$ and a 48-bit round-key $KI_i = KI_{i,1}||KI_{i,2}||KI_{i,3}$, and produces a 32-bit output. More precisely, denote $I = L_0||R_0$, for $j = 1, 2, 3$, perform the following operations:

$$\begin{cases} R_j = FI_{i,j}(L_{j-1} \oplus KO_{i,j}, KI_{i,j}) \oplus R_{j-1} \\ L_j = R_{j-1} \end{cases}$$

Then the output of the FO_i function is the 32-bit data block $L_3||R_3$.

It is noted that the computation of the function FO_i involves another $FI_{i,j}$ function. For a given i and j, where $1 \leqslant i \leqslant 8$ and $1 \leqslant j \leqslant 3$, an FI-function $FI_{i,j}$ takes a 16-bit data x and a 16-bit subkey $KI_{i,j}$ as input, and produces a 16-bit data. More precisely, the data x is split into a 9-bit left part l_0 and a 7-bit right part r_0, similarly the subkey $KI_{i,j}$ is also split into a 9-bit left part and a 7-bit right part as $KI_{i,j} = KI_{i,j,1}||KI_{i,j,2}$. There are also two S-boxes needed, one is S_9 that maps a 9-bit input into a 9-bit output, and the other is S_7 that maps a 7-bit input into a 7-bit output. The detailed definition of the S-boxes is not specified here, as they are widely and publicly available. Denote $LS_7(y)$ as the least significant 7-bit part of y. Then the function $FI_{i,j}$ is defined by the following series of operations:

$$\begin{array}{ll} l_1 = r_0 & r_1 = S_9(l_0) \oplus (00||r_0) \\ l_2 = r_1 \oplus KI_{i,j,2} & r_2 = S_7(l_1) \oplus LS_7(r_1) \oplus KI_{i,j,2} \\ l_3 = r_2 & r_3 = S_9(l_2) \oplus (00||r_2) \\ l_4 = S_7(l_3) \oplus LS_7(r_3) & r_4 = r_3 \end{array}$$

The output of $FI_{i,j}$ is the 16-bit data block $l_4||r_4$.

With the introduction of the core functions in KASUMI, it is easy to introduce the encryption process. Given a 128-bit key K and a 64-bit input message m, the message is first split into two 32-bit halves, $m = L_0||R_0$. Then for each round i with $1 \leqslant i \leqslant 8$, the operation of KASUMI on the i-th round is as follows:

$$R_i = L_{i-1} \quad L_i = R_{i-1} \oplus f_i(L_{i-1}, RK_i)$$

where RK_i is the round key which in fact is defined as a triplet of subkeys (KL_i, KO_i, KI_i). The function f_i differs in odd rounds and even rounds. For round number $i = 1, 3, 5, 7$, the f-function is defined as:

$$f_i(L_{i-1}, RK_i) = FO_i(FL_i(L_{i-1}, KL_i), KO_i, KL_i)$$

and for round number $i = 2, 4, 6, 8$, the f-function is defined as:

$$f_i(L_{i-1}, RK_i) = FL_i(FO_i(L_{i-1}, KO_i, KI_i), KL_i)$$

The final output of the algorithm is the 64-bit data block $L_8||R_8$.

Although KASUMI is a minor modification of MISTY suitable for hardware implementation, it is surprise to note that, in 2010, Dunkelman et al. published a paper[7] claiming that they could break KASUMI with a related key attack and very modest computational resources. Interestingly, the attack is ineffective against MISTY.

2.3.6 The inherent security weakness of 2G networks

It is not known how far the A5/3 algorithm can secure the GSM system in the sense of air data confidentiality, and given the weakness of A5/3 having been found[8], a new algorithm A5/4 may be in place in the near future. However, the inherent AKA process of GSM system has some fatal weakness. More precisely it only provides one-way authentication. i.e., it enables the users to authenticate themselves to the network, and does not provide functionality for the network to authenticate itself to the end users. This may cause some attacks by false base stations. Fig. 2.5 depicts how a false base station could eavesdrop a victim mobile user.

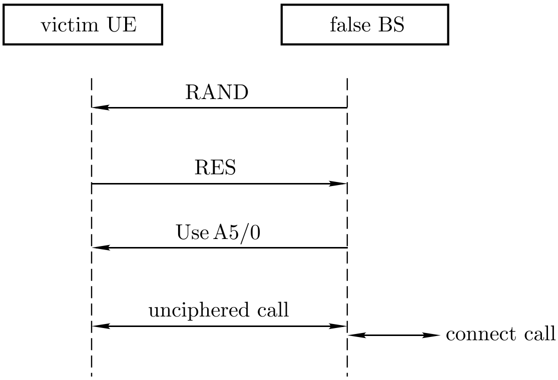

Fig. 2.5 Eavesdropping by a false base station.

2.4 Security techniques in 3G networks

The technology for mobile communication evolves quickly, especially in recent years. The initial purpose of cellular communication networks were meant to serve for voice communications, while the second generation of cellular communication systems can also provide short message services and even some extended services such as GPRS. The most significant improvement of the second generation of mobile networks over the first generation is the information security, including the functionality for end user authentication and data confidentiality. However, the lack of mutual authentication of the GSM system suffers the attack by false networks. On the other hand, the GSM system has limited channel bandwidth, which is perhaps enough for voice communications. With the development of mobile networks, the increased functionality, and the demand of mobile devices, a wider wireless bandwidth is needed while a higher security is to be provided. This leads to the 3G mobile communication networks (or 3G networks for short).

There are different techniques in 3G networks, including WCDMA, CDMA2000, and TD-SCDMA networks. There are many core techniques in common, that is code division multiple access (CDMA) techniques, and there are also essential differences between any of the networks. The most similar networks from architecture point of view are WCDMA and CDMA2000. They are also the most widely used 3G networks. Here we will introduce the security architecture of these networks (say WCDMA), and without confusion we simply name it as 3G network security architecture.

2.4.1 The mutual authentication in 3G networks

In a 3G network, the network components are commonly known as user equipment (UE), eNodeB (essentially a base station), a mobility management entity (MME), and a home subscriber server (HSS) who also serves as network authentication server. The authentication and key agreement (AKA) process can be depicted in Fig. 2.6. The functionality of MME is very much like the VLR as in GSM systems, that of HSS is very much like that of HLR as in GSM systems, where the eNodeB is a connection between end user and an MME. To make it simple and comparable with the AKA process in GSM networks, we treat eNodeB as part of MME when dealing with the security functionalities. The process of authentication and key agreement in a 3G network can be depicted in Fig. 2.6.

It is seen from Fig. 2.6 that the AKA process in 3G networks are almost the same as that in 2G networks, except that in 3G networks, the authentication vectors are 5-tuples versus triplicates as in 2G networks, and there is an AUTH send from the network to the end user for verifying the network authenticity. The temporary user equipment identity in 3G networks is named

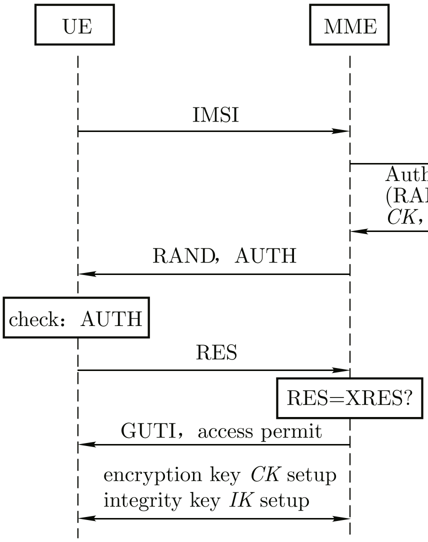

Fig. 2.6 The AKA process in a 3G network.

as globally unique temporary identity (GUTI).

2.4.2 The confidentiality algorithm f_8 and the integrity algorithm f_9

Although there are some similarities of the AKA processes in different networks, there are substantial differences as well. In 3G networks, the 5-tuple authentication vectors are generated by algorithms totally different from that in 2G networks. More precisely, there are 11 security algorithms defined in 3G networks, they are f_0, f_1^*, $f_1 \sim f_9$, where f_0 is a pseudorandom number generator that generates random challenges, f_1 is used to generate a message authentication code (MAC) to be part of the authentication token AUTH, f_1^* is used for the resynchronization of message authentication, f_2 is used to generate the expected response (XRES) corresponding to the challenge RAND, f_3 is used to generate an encryption key CK, f_4 is used to generate an integrity key IK, f_5 is used to generate an anonymity key AK. The functions $f_1 \sim f_5$ are responsible for generating the authentication vectors, and Fig. 2.7 shows how they work in general.

Note from Fig. 2.7 that the inputs of f_1 also includes AMF and SQN, they are authentication management field (AMF) and sequence number (SQN), two parameters known to both the end user and the home subscriber server. The common inputs to all the five functions are RAND and K, where K is the long term user key shared between the mobile user (in a SIM or USIM card)

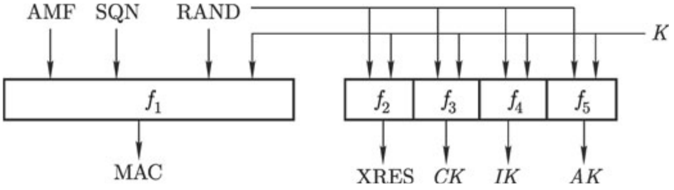

Fig. 2.7 Authentication vector generation in 3G networks.

and the network authentication center resides in HSS. The output of f_1 is not yet the authentication token AUTH, which in fact can easily be computed given the inputs and outputs of $f_1 \sim f_5$. In fact, AUTH=(SQN$\oplus AK$)||AMF|| MAC, where $\oplus$ means bitwise XOR operation and || is concatenation.

The other functions, $f_6 \sim f_9$, are as follows: f_6 and f_7 are used to provide enhanced user identity encryption, where f_6 is the process for encryption, and f_7 is the inverse of f_6. f_8 is a stream cipher that encrypts the user-network air-interface communication after mutual authentication is successful, and f_9 is an algorithm for generating a message authentication code (MAC) for the signaling messages. Fig. 2.8 shows the structure of algorithms f_8 and f_9.[9]

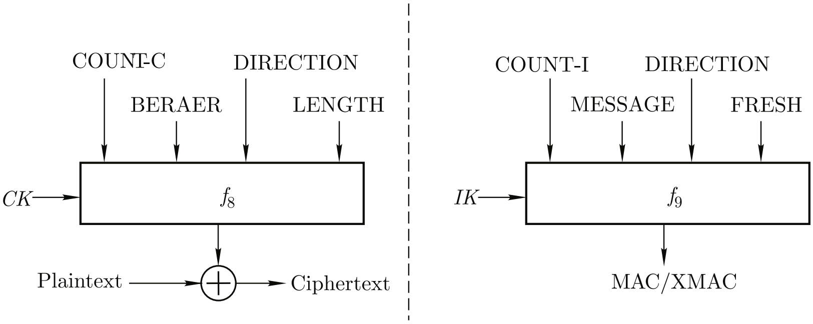

Fig. 2.8 The structure of f_8 and f_9.

Although there are 11 functions in the 3G networks, apart from the supplementary security functions such as f_0, f_1^*, f_6 and f_7, most security functions take KASUMI and AES as the core algorithm.[10,11] Although f_8 is a stream cipher, and KASUMI is a block cipher, it uses the output feedback (OFB) mode of KASUMI to build a stream cipher from a block cipher. As has been pointed out in section 3, since the algorithm KASUMI has some security problems revealed, the related functions that are built upon the KASUMI algorithm may also have security problems[12]. Fortunately, this situation is improved in the LTE networks.

2.5 Security techniques in LTE networks

The 3GPP organization has been working on the quality and capability of mobile communications. With some limitations of the 3G networks emerged, including the security limitations, a new generation of networks named long term evolution (LTE) is proposed. The LTE networks are targeted at an even higher rate data transmission than that of 3G networks and hence can provide more services. Naturally with increased number and intensity of services, the security becomes more sensitive. Given that the 3G networks use KASUMI as the core cryptographic algorithm on top of which many security functions are built, and the KASUMI algorithm has some security problems revealed. The LTE networks tend to employ different suits of algorithms, named as EEA and EIA, mainly for providing confidentiality (the function f_8) and integrity (the function f_9) services respectively. Since many of the security functions (e.g. $f_1 \sim f_7$) can be different from network to network, some internal modification within a network operator is practically possible. However, the security functions for data confidentiality and integrity have to be globally standardized and their security is of great interest and is also the most concerned.

2.5.1 The confidentiality and integrity algorithm sets for LTE

The first suite of the algorithms, 128-EEA1/128-EIA1, is based on a stream cipher called SNOW-3G, designed by the Security Algorithms Group of Experts (SAGE), part of the European standards body ETSI. The second suite of the algorithms, 128-EEA2/128-EIA2, is based on the Advanced Encryption Standard (AES). The third suite of the algorithms, 128-EEA3/128-EIA3, is based on a newly proposed stream cipher named ZUC. Due to the well availability of the AES algorithm discussions, we will only look into the algorithms SNOW-3G and ZUC, yet in a very brief manner, which intend to show their structural similarities and differences. First, we give a diagram to show the structure of SNOW-3G (see Fig. 2.9).

From Fig. 2.9, it is seen that SNOW-3G has two essential parts[13], a linear feedback shift register (LFSR) of order 16 defined over the finite field GF(2^{32}), where α is a specific field element, and a finite state machine (FSM) composed of three memory registers R_1, R_2 and R_3, and two S-boxes, S_1 and S_2. The output z is a sequence of elements over GF(2^{32}). i.e., each output of the algorithms is a 32-bit data block.

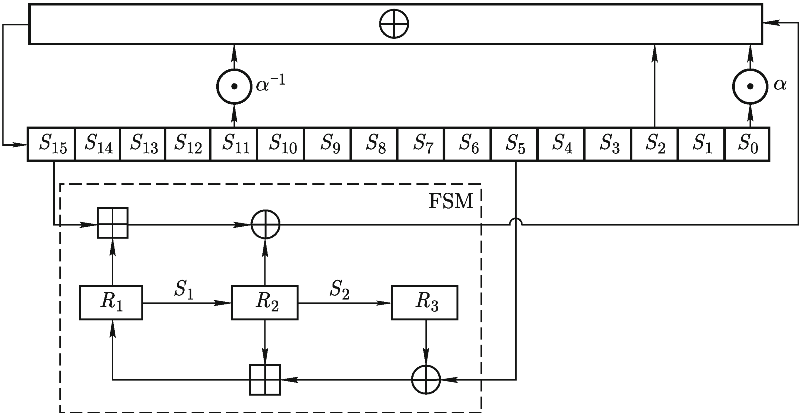
α^{-1}
α
S_{15} S_{14} S_{13} S_{12} S_{11} S_{10} S_9 S_8 S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_0
FSM
R_1
S_1
R_2
S_2
R_3

Fig. 2.9 The structure of SNOW-3G.

2.5.2 A new stream cipher ZUC

The name of the stream cipher ZUC is after a famous Chinese mathematician in the history named as Zu Chongzhi. The structure of ZUC, as shown in Fig. 2.10, looks to have some similarities with that of SNOW-3G. In fact, they are similar in a few phases. First, both of the algorithms have two components: an LFSR and a finite state machine. Second, they both use 128-bit seed key and output 32-bit key streams. Third, they both employ a mixture of different operations, e.g. addition and multiplication over a finite field, exclusive-OR, and addition modular an integer. However, it is also easy to find some substantial differences. First, ZUC used an LFSR defined over a prime finite field GF$(2^{31}-1)$ which seems to have more complicated algebraic structure when being viewed over the binary field GF(2). Second, ZUC has a bit-reorganization operation which breaks the algebraic structure of the contents from the LFSR cells. Third, the finite state machine component in ZUC seems to be more complicated than that in SNOW-3G. And fourth, ZUC has more operations than those used in SNOW-3G. It is not surprising that the performance of ZUC is slightly degraded compared with that of SNOW-3G. On the other hand, the structural differences of ZUC compared with SNOW-3G make the two algorithms not likely to stand or fall together, as has been pointed by the two review reports[19,20].

It should be pointed out that the security functions f_8 and f_9 as in LTE networks are not bounded by the use of SNOW-3G or ZUC. SNOW-3G is a standard algorithm and ZUC is in the process of becoming a standard algorithm. With the development of mobile networks, there may be more cryptographic algorithms introduced in the future, while some of the existing algorithms may be abandoned.

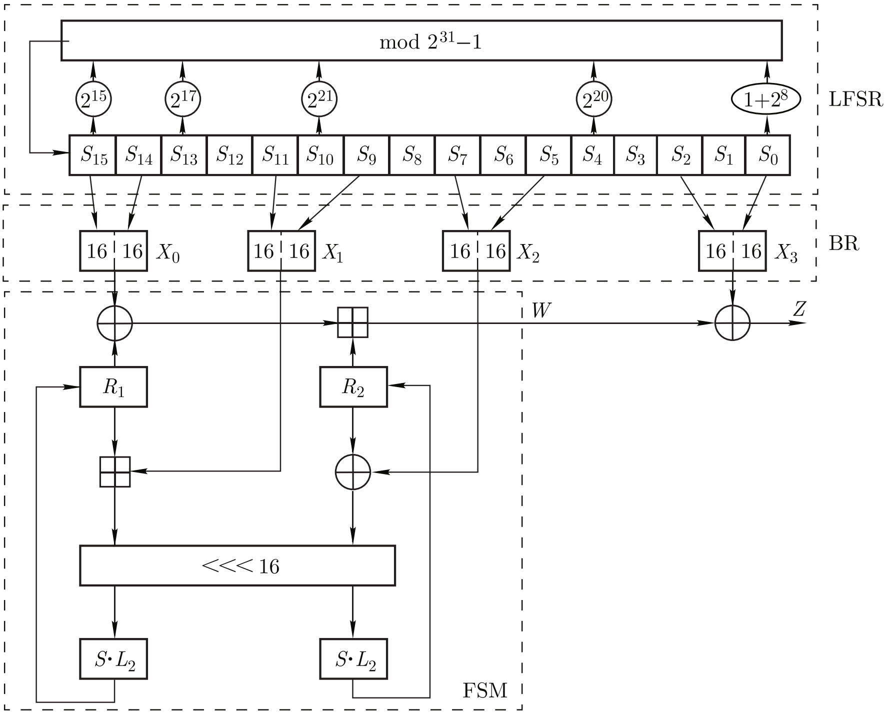

Fig. 2.10 The structure of ZUC.

2.5.3 The confidentiality/integrity algorithm set 128-EEA3/128-EIA3

Within the security architecture of the LTE system, there are standardized algorithms for confidentiality and integrity. Two sets of algorithms 128-EEA1/128-EIA1 and 128-EEA2/128-EIA2 have been specified as standard[9,10]. The third set of algorithms 128-EEA3/128-EIA3 is based on the stream cipher ZUC as described above.

The confidentiality algorithm 128-EEA3 is a stream cipher that is used to encrypt/decrypt blocks of data using a confidentiality key CK. The block of data may be between 1 and 20 000 bits long. The integrity algorithm 128-EIA3 computes a 32-bit Message Authentication Code (MAC) of a given input message using an integrity key IK. Since ZUC is an algorithm needing an initialization vector (IV) as well as an initial key for the initialization, there will be an IV involved both in the confidentiality algorithm 128-EEA3 and the integrity algorithm 128-EIA3[14,15].

The inputs to the algorithms 128-EEA3/128-EIA3 are given in Table 2.1. The encryption key for data confidentiality is CK and that for data integrity is IK. Both are a string of 128 bits.

Table 2.1 The inputs to 128-EIA3/128-EIA3

Parameter	Size in bits	Meaning
COUNT	32	The counter
BEARER	5	The bearer identity
DIRECTION	1	The direction of transmission
CK/IK	128	The integrity key
LENGTH	32	The bits of the input message
M	LENGTH	The input message

Let

$$\text{COUNT}=\text{COUNT}[0]\|\text{COUNT}[1]\|\text{COUNT}[2]\|\text{COUNT}[3]$$

be the 32-bit counter, where COUNT[i] ($0 \leqslant i \leqslant 3$) are bytes. The 128-bit initialization vector for 128-EEA3 is set as

$$\text{IV} = \text{IV}[0]\|\text{IV}[1]\|\text{IV}[2]\|\ldots\|\text{IV}[15],$$

where IV[i] ($0 \leqslant i \leqslant 15$) are bytes, defined by

$$\begin{aligned}
&\text{IV}[0] = \text{COUNT}[0], \quad \text{IV}[1] = \text{COUNT}[1],\\
&\text{IV}[2] = \text{COUNT}[2], \quad \text{IV}[3] = \text{COUNT}[3],\\
&\text{IV}[4] = \text{BEARER}\|\text{DIRECTION}\|00,\\
&\text{IV}[5] = \text{IV}[6] = \text{IV}[7] = 00000000,\\
&\text{IV}[8] = \text{IV}[0], \quad \text{IV}[9] = \text{IV}[1],\\
&\text{IV}[10] = \text{IV}[2], \quad \text{IV}[11] = \text{IV}[3],\\
&\text{IV}[12] = \text{IV}[4], \quad \text{IV}[13] = \text{IV}[5],\\
&\text{IV}[14] = \text{IV}[6], \quad \text{IV}[15] = \text{IV}[7].
\end{aligned}$$

The initialization vector used for 128-EIA3, IV=IV[0]‖IV[1]‖IV[2]‖…‖ IV[15], is defined by

$$\begin{aligned}
&\text{IV}[0] = \text{COUNT}[0], \quad \text{IV}[1] = \text{COUNT}[1],\\
&\text{IV}[2] = \text{COUNT}[2], \quad \text{IV}[3] = \text{COUNT}[3],\\
&\text{IV}[4] = \text{BEARER}\|(000)_2, \quad \text{IV}[5] = (00000000)_2,\\
&\text{IV}[6] = (00000000)_2, \quad \text{IV}[7] = (00000000)_2,\\
&\text{IV}[8] = \text{IV}[0] \oplus (\text{DIRECTION} \ll 7), \quad \text{IV}[9] = \text{IV}[1],\\
&\text{IV}[10] = \text{IV}[2], \quad \text{IV}[11] = \text{IV}[3],\\
&\text{IV}[12] = \text{IV}[4], \quad \text{IV}[13] = \text{IV}[5],\\
&\text{IV}[14] = \text{IV}[6] \oplus (\text{DIRECTION} \ll 7), \quad \text{IV}[15] = \text{IV}[7].
\end{aligned}$$

There is not much to say about the 128-EEA3 algorithm, because it is basically to run the ZUC algorithm to encrypt the message using the encryption key CK and the initialization vector IV. It only needs to note that the key is used only to encrypt a message of up to 20 000 bits.

The 128-EIA3 algorithm, however, has something more than the ZUC. More precisely, Let $N = \text{LENGTH} + 64$ and $L = \lceil N/32 \rceil$. Let ZUC generate L 32-bit key words $z[0]$, $z[1]$, $\cdots$, $z[L-1]$ with the initial key IK and the initialization vector IV is defined as above, where $z[0]$ is the first key word generated by ZUC, $z[1]$ is the next, and so on. Let $k[0]$, $k[1]$, $\cdots$, $k[31]$, $k[32]$, $\cdots$, $k[N-1]$ be the key bit stream corresponding to the above key words $z[1]$, $\cdots$, $z[L-1]$. Then $N = 32 * L$.

For each $i = 0, 1, 2, \cdots, N-32$, let $k_i = k[i] \| k[i+1] \| \ldots \| k[i+31]$. Then each k_i is a 32-bit word. LET T be a 32-bit word. Set $T = 0$, and for each $i = 0, 1, 2, \cdots$, LENGTH-1, if $M[i] = 1$, then set $T = T \oplus k_i$. At last let $T = T \oplus k_{\text{LENGTH}}$. Finally we take $T \oplus k_{N-32}$ as the output MAC, i.e. $\text{MAC} = T \oplus k_{N-32}$.

2.5.4 The security flaws and improvements of ZUC

The ZUC algorithm has an initialization process before actual key stream can be produced. The initialization can be depicted in Fig. 2.11.

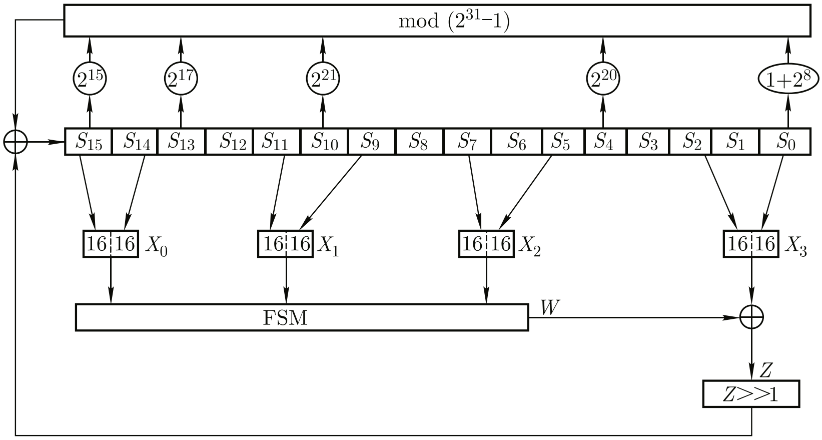

Fig. 2.11 The initialization of ZUC version 1.0.

Certain effort has been made on evaluating the security of the ZUC algorithm. First of all, the designers have made a comprehensive security analysis and evaluation. Then some professional evaluation groups gave their evaluation reports[19,20]. Surprisingly none of these evaluations has found obvious security flaws of ZUC.

It was realized that the ZUC algorithm has some security problems not long after the publication of the algorithm specification. The first flaw was notified by both the designers as well as some external members, and has been reported at the first international workshop on ZUC conducted in December 2010. It was found that the initialization process does not keep key entropy due to that $\boldsymbol{z}$ is involved in updating the LFSR feedback. In fact, the very first version of ZUC was to use $\boldsymbol{w}$ to be involved in updating the LFSR, and the change was to base on the assumption that $\boldsymbol{z}$ is more likely to be balanced than $\boldsymbol{w}$. In fact, a more serious problem was found by Wu[21] that, when $\boldsymbol{z}$ or $\boldsymbol{w}$ was truncated from a 32-bit string into a 31-bit string, it can cause problems. Denote that as $\boldsymbol{z}$, let the feedback of the LFSR be $\boldsymbol{x}$, then when $\boldsymbol{z} = \boldsymbol{x}$ and $\boldsymbol{z} = 2^{32} - \boldsymbol{x}$, these two values will all result in $\boldsymbol{z} \oplus \boldsymbol{x} = \mathbf{0}$, and hence cause attacks.

Notifying these security flaws, the version 1.5 of ZUC released on 4th January 2011[23] has some changes on the initialization of ZUC, which can be depicted in Fig. 2.12.

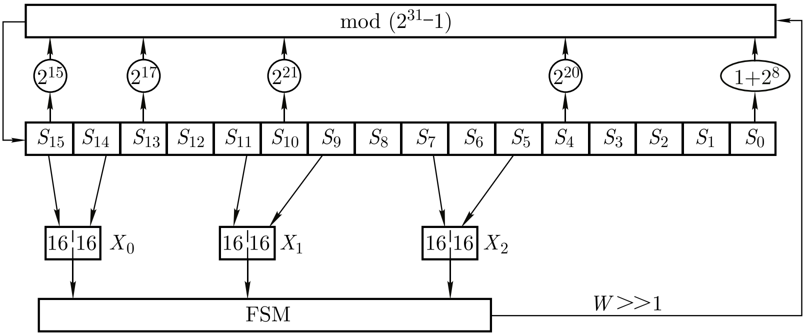

Fig. 2.12 The initialization of ZUC version 1.5.

It is noted that there are two changes in the ZUC version 1.5 over the earlier version. One of the changes is that $\boldsymbol{w}$ instead of $\boldsymbol{z}$ is involved in updating the content of s_{15} in the LFSR, and another change is that the operation on $\boldsymbol{w} \gg 1$ with the feedback of the LFSR is addition modulo $2^{31} - 1$ instead of XOR which was found to cause problems. These two simple changes seem to have solved the security flaws found so far.

2.5.5 The security flaws and an improvement of 128-EIA3

Regardless which version of the ZUC algorithm is used, the 128-EIA3 in its earlier versions were found to be insecure[22]. The process of 128-EIA3 can be depicted in Fig. 2.13.

The attack found by Fuhr et al.[22] on 128-EIA3 before the version 1.5 is as follows: let message M produces a MAC using a key IK and an initialization vector IV. Let $M' = 1||M$ be the concatenation of 1 and the message M.

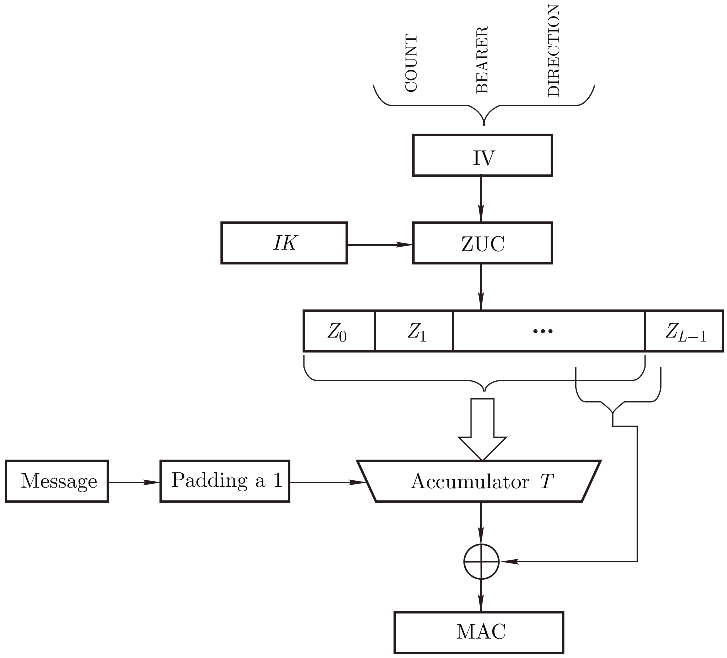

Fig. 2.13 The process of the EIA3 MAC computation.

Then it is easy to verify that the new authentication code MAC′ for M' under the same set of key/IV will have 31 bits in common with MAC, so one can guess the other bit of MAC′ with a probability of 50% to succeed. This high probability of successfully forging an authentication code is not acceptable.

It is noted that the cause of forgery authentication code to be possible was mainly from the observation that the last "mask" to finally producing a MAC was not really random for different messages, particularly for the two messages in the form above. Therefore, a simple change can amend the security flaw. In the version 1.5 of 128-EIA3, the following change has been made:

Let ZUC generate a key stream of $L = \lceil \text{LENGTH}/32 \rceil + 2$ words, each word is 32 bits. Denote the generated bit string by $z[0], z[1], \cdots, z[32 \times (L-1)]$, where $z[0]$ is the most significant bit of the first output word of ZUC and $z[31]$ is the least significant bit.

For each $i = 0, 1, 2, \cdots, 32 \times (L-1)$, let $z_i = z[i] \| z[i+1] \| \ldots \| z[i+31]$. Then each z_i is a 32-bit word. Let T a block of 32 bits word. Set $T = 0$.

```
For each $i = 0, 1, 2, \cdots,$ LENGTH$-1$, if $M[i] = 1$, then
    $T = T \oplus z_i$.
Set
    $T = T \oplus z_{\text{LENGTH}}$.
```

Finally we take $T \oplus z_{32\times(L-1)}$ as the output of MAC, i.e.

MAC= $T \oplus z_{32\times(L-1)}$.

In brief, the change of the EIA3 was to use the next word generated by ZUC as the mask, as depicted in Fig. 2.14, where in the earlier version a 32-bit block immediately following the computation of T was used.

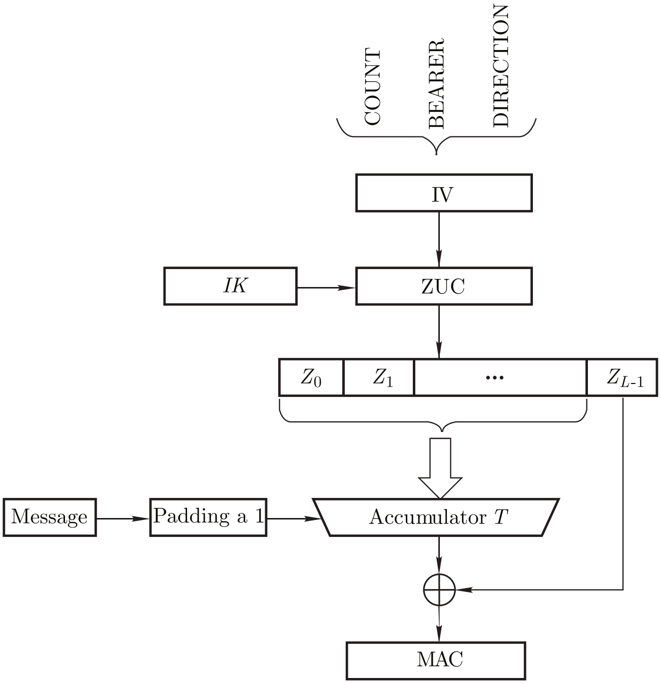

Fig. 2.14 The process of the EIA3-version1.5 MAC computation.

2.5.6 The security limitation of the authentication algorithm in LTE

It is noted that in LTE systems, the authentication code of any message is defined to be in 32 bits. This remains the same in EIA1, EIA2, and EIA3. Because with a birthday attack, a collision can be found with computation effort of 2^{16}, or a forgery authentication code can be found for any given message with computation effort of 2^{32}. This limitation however is an inherent problem in LTE and cannot be changed, unless the industry standard is to be changed.

2.6 Security issues in femtocell

With the increased demand on the network bandwidth of mobile communications, signal frequency becomes higher and higher, and the coverage of each base station also becomes smaller. This means that when being viewed as cellular networks, the cells become smaller. Moreover, for indoor mobile devices, good signal coverage needs more base stations to be installed. This also incurs high cost both of installation and of maintenance.

A new system called "femtocell" was proposed[25] to solve the problem of signal coverage limitations for wideband mobile networks. The femtocell introduces a kind of micro-station, called Home NodeB (HNB) as in 3G networks, or Home eNodeB (HeNB) as in LTE networks, is a good solution to compensate the limitation of indoor signal coverage. The network structure of femtocells can be depicted in Fig. 2.15.

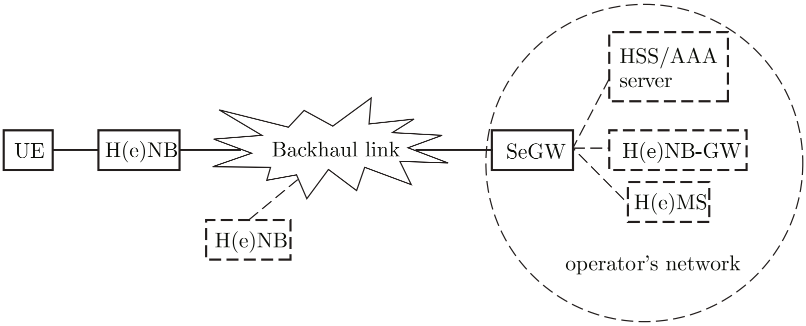

Fig. 2.15 The network structure of femtocell.

From Fig. 2.15 it is seen that the Home (e)NodeB (H(e)NB) is a micro station serving for mobile users (e.g. UE), where the H(e)NB is connected to an operator's network via the Internet (LAN or ADSL), which is not trusted by the network operator.

There are different sources of security threats to the femtocell system. First, the environment where the H(e)NB's are installed is not trusted by the operator, this is because the H(e)NB's are often installed in places out of control by the operator, such as private homes, private hotels, or buildings, where public access is limited, so that the access to the H(e)NB's by the operators becomes difficult. Second, the connection from the H(e)NB's to the operator's core networks is not controlled and hence trusted by the operator, because the connection is often through public networks such as the Internet. Third, the number of H(e)NB's can become quite large compared with the traditional base stations, which may induce unpredictable security threats, such as colluding attack from the H(e)NB's on the network (the public network or the operator's network).

When an H(e)NB is installed in an untrusted environment, users may install illegally manufactured H(e)NB's, or to manipulate some originally legal H(e)NB's for perhaps malicious purposes. The illegal modification/installation of H(e)NB's may abuse some victim mobile users when being serviced, or provide wrong information to the operator which will confuse the accounting process. Due to the H(e)NB's are connected to the operator's core network via public networks, some attacks to the public networks may be modified to attack the femtocell system, either the end H(e)NB's or the operator's core networks. The security threats due to a large number of H(e)NB's can come from the process of manufacture control, since H(e)NB's need to have a key (a shared symmetric key or a private key corresponding to a public key certificate) burned when they are being manufactured.

There are many standard requirements for the manufacturing and installation of H(e)NB's, but from the point of view of attackers, there is no point to follow those requirements. Technical requirements and rules do not provide good solutions to practical security threats.

Security threats should taken into consideration the situation when an H(e)NB is also integrated as a security gateway of a home-based wireless network, this is a specific scenario in the concept of Internet of Things (IoT), and is not discussed in depth here.

2.7 Privacy issues in cellular networks

With the increased number of users in cellular networks, and with the increased demand on services provided by the cellular and mobile networks, user privacy becomes a notable issue. Many kinds of information that were previous not treated as private now need to be classified as private. For example the information about the location of a mobile user is such an instance. When the number of users was small, it was not a big concern. But now the location information may be illegally used to trace a user and becomes a kind of private information and hence needs to be protected. The privacy issue is particularly important in some of the services such as mobile device assisted medical/therapy systems.

From cryptographic approach, there are techniques closely related to privacy issues. For example, blind signature, anonymous signature, and zero knowledge proof in some sense. However, those techniques need to be applied to practical application environment to solve practical privacy requirements. There seem to have some confliction in the privacy problem. A natural understanding on the privacy issue is the incapability to link an identity to some other information, e.g. medical record. However, when a particular medical record is needed, there must be a valid identity provided for the purpose of authentication. On one hand, how to use an identity information for the authentication purpose and on the other hand to protect the identity information against unauthorized access is a big challenge. Very often the problem becomes simpler if a trusted third party (TTP) is involved. The limitation is communication and computation bottleneck of the TTP. So far technical solutions on the privacy issues are far from being satisfactory in many practical applications, this is an area needing to be studied further and it has enormous practical applications.

2.8 Security issues of mobile devices

The security of cellular networks and communication systems tend to provide better security services, including the security protocols and security algorithms. However, there are security threats on the mobile equipments not caused by communication security protocols or the security algorithms. These kinds of security threats include the operating system malfunctioning and the loss of mobile devices.

The advanced mobile devices today are not just cellphones, they include laptop computers. Even considering only the cellphones, many of the cellphones have many functionalities same as in a computer, including an operating system and many applications. Most popular operating systems for smart cellphones include Symbian, Windows CE, Windows Mobile, and palm OS. Different operating systems may have different behavior in different aspects, for example with respect to the memory management, response time, and energy management. They all suffer security threats caused by mobile worms, viruses, and unauthorized access. Since the mobile operating system cannot have sophisticated antivirus software which would consume much of the operation resource, mobile operating systems are more fragile against security threats than many other operating systems for computers. Therefore, lightweight and efficient antivirus software for cellphones will be in a high demand.

Another security threats for mobile communication is the loss of mobile devices, in particular the missing, stolen, damage of cellphones. In this case the users will lose all the important information stored in the cellphones. To reduce the loss caused by mobile device loss, secure and timely data backup services are important. In the case of a cellphone being stolen, it may risk the privacy of the cellphone owner. The proper protection of the data stored in cellphones is important. Therefore, with respect to the data security and protection of mobile devices, on one hand the data stored in mobile devices needs to be securely protected, and on the other hand there should be a good way for secure data backup and recovery mechanism.

2.9 Concluding remarks

The technology of wireless and mobile communication evolves, and the concept of cellular networks also evolves. Today with the concept of ubiquitous computing becoming popular and the technology becoming mature, the concept of ubiquitous networks also emerges, which seems to cover the traditional concept as well as many advanced techniques of cellular networks. The new concept also means more services and inevitably more security threats and challenges as well.

This chapter tries to give a general picture about different aspects of security problems in cellular networks and communications. It was realized that there are so many techniques with respect to the information security problems in cellular networks and communications, and it ended up with a very brief introduction and has limited coverage. It was hoped that this introduction could help some readers to know the security challenges and motivate their interest to find good solutions.

References

[1] Farley T (2007) The Cell-Phone Revolution. American Heritage of Invention & Technology, 22(3): 8 – 19.

[2] Goldberg I, Wagner D, Green L (1999) The (Real-Time) Cryptanalysis of A5/2. Rump session of Crypto'99.

[3] Barkan E, Biham E, Keller N (2003) Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication. Proceedings of Crypto 2003, LNCS 2729: 600 – 616. Springer-Verlag, Berlin.

[4] Ekdahl P, Johansson T (2003) Another attack on A5/1. IEEE Transactions on Information Theory, 49(1): 284 – 289.

[5] Barkan E, Biham E (2006) Conditional Estimators: An Effective Attack on A5/1. Selected Areas in Cryptography 2005, LNCS 3897: 1 – 19. Springer-Verlag, Berlin.

[6] Matsui M, Tokita T (2000) MISTY, KASUMI and Camellia Cipher Algorithm Development. Mitsibishi Electric Advance (Mitsibishi Electric corp.) 100: 2 – 8.

[7] Dunkelman O, Keller N, Shamir A (2010) A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony. CRYPTO 2010, LNCS 6223: 393 – 410, Springer-Verlag.

[8] Dunkelman O, Keller N, Shamir A (2010) A Practical-Time Attack on the A5/3 Cryptosystem Used in Third Generation GSM Telephony. Cryptology ePrint Archive: Report 2010/013.

[9] Specification of the 3GPP Confidentiality and Integrity Algorithms; Document 1: f_8 and f_9 specifications (3GPP TS35.201 Release 6). Available at http://www.3gpp.org/ftp/Specs/html-info/35201.htm. Accessed 10 November, 2011.

[10] 3GPP System Architecture Evolution (SAE); Security architecture (3GPP TS33.401 Release 9). Available at http://www.3gpp.org/ftp/Specs/html-info/33401.htm. Accessed 10 November, 2011.

[11] Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Document 1: UEA2 and UIA2 specifications. Available at http://cryptome.org/uea2-uia2/uea2-uia2.htm. Accessed 10 November, 2011.

[12] Specification of the 3GPP Confidentiality and Integrity Algorithms; Document 2: KASUMI Specification (3GPP TS35.202). Available at http://www.3gpp.org/ftp/Specs/html-info/35202.htm. Accessed 10 November, 2011.

[13] Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Document 2: SNOW 3G specification. Available at http://www.3gpp.org/ftp/Specs/html-info/35216.htm. Accessed 10 November, 2011.

[14] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms EEA3 & EIA3. Document 1: EEA3 and EIA3 Specification; Version: 1.0; Date: 18th June, 2010. http://www.gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm. Accessed 10 November, 2011.

[15] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms EEA3 & EIA3. Document 2: ZUC Specification; Version: 1.0; Date: 18th June, 2010. http://www.gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm. Accessed 10 November, 2011.

[16] ETSI/SAGE Specification. Specification of the MILENAGE-2G Algorithms: an Example Algorithm Set for the GSM Authentication and Key Generation Functions A3 and A8. Version 1.0. May, 2002. http://www.gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm. Accessed 10 November, 2011.

[17] Dai Watanabe, Alex Biryukov, Christophe De Cannière (2004) A distinguishing attack of SNOW 2.0 with linear masking method. In Selected Areas in Cryptography 2003. LNCS 3006: 222–233, Springer-Verlag, Berlin.

[18] Nicolas Courtois and Willi Meier (2003) Algebraic Attacks on Stream Ciphers with Linear Feedback, In Advances in Cryptology-EUROCRYPT 2003, LNCS 2656: 346–359, Springer-Verlag, Berlin.

[19] Knudsen L R, Preneel B, and Rijmen V (2010) Evaluation of ZUC, ABT Crypto, Version 1.1, May, 2010.

[20] Cid C, Murphy S, Piper F, and Dodd M (2010) ZUC Algorithm Evaluation Report, Codes & Ciphers Ltd., 7 May, 2010.

[21] Hongjun Wu, et.al. (2010) Cryptanalysis of Stream Cipher ZUC in the 3GPP Confidentiality & Integrity Algorithms 128-EEA3 & 128-EIA3, presented at the Rump session of Asiacrypt 2010, Singapore.

[22] Fuhr T, Gilbert H, Reinhard J R, and Videau M (2010) A forgery attack on the candidate LTE integrity algorithm 128-EIA3 (updated version), Cryptology ePrint Archive, Report 2010/618, 2010, Available at http://eprint.iacr.org/. Accessed 10 November, 2011.

[23] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 1: 128-EEA3 & 128-EIA3 Specification; Version: 1.5, 4th January, 2011. Available at http://www.gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm. Accessed 10 November, 2011.

[24] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3, Document 2: ZUC Specifica-tion, Version: 1.5, 4th January, 2011. Available at http://www.gsmworld.com/our-work/programmes-and-initiatives/fraud-and-security/gsm_security_algorithms.htm. Accessed 10 November, 2011.

[25] Femtocell forum, http://www.femtoforum.com/femto/.