State-of-the-art on CDMA2000 Security Support

Luuk Weltevreden
Department of Computer Science
Faculty of Electrical Engineering, Mathematics and Computer Science
University of Twente
l.weltevreden@student.utwente.nl

ABSTRACT
As mobile technology and the services it provides are evolving, security cannot stay behind. In the past, ensuring that the right subscriber gets the bill from his or hers phone calls seemed to be enough. However, fraudulent use of the mobile services proved otherwise. Additionally, people desire better security in the form of privacy and the coming of new services, such as m-commerce, require a secure environment to work with.

For all of these reasons, a solid security architecture is required for the latest mobile technologies. Third generation technologies (or 3G in short) are the latest technologies used by mobile service providers around the world. CDMA2000 is one of these third generation digital mobile technologies. In this paper the security architecture of CDMA2000 will be discussed and evaluated. The standardized protocols and algorithms will be discussed and compared with possible alternatives. Finally a conclusion will be given on the state of the security architecture and recommendations will be given based on this comparison.

Keywords
CDMA2000, 3GPP2, Cellular Networks, Security

1. INTRODUCTION
These days security in mobile networks is becoming increasingly more important. According to [CN05] there are already 1.5 billion Global System for Mobile communications (GSM) subscriptions worldwide. The Universal Mobile Telecommunications System (UMTS) Forum states that the total worldwide subscriptions for 3G mobile networks have now passed the 50 million mark and are increasingly growing, see [UF05]. Fraudulent use of network resources and the desire for user privacy are becoming more apparent. For applications such as m-commerce to be successful, a secure environment is obligatory.

This desire for better security has been acknowledged and since the earliest form of mobile technology, security has been improved significantly. Security in analogue mobile networks was nearly nonexistent and has been left up primarily to system manufacturers and service providers. With the advent of digital technology a step in the right direction was taken. Security mechanisms were developed to protect service providers against fraudulent use and to provide end-users with better privacy. However, the second generation (2G) security architecture is still far from perfect and leaves a lot to be desired. With third generation (3G) technology another attempt is made to improve the security of mobile networks.

3G technologies are an answer to the IMT-2000 specification developed by the International Telecommunications Union. The original goal was to develop a single, world-wide standard for mobile networks, but in reality many different standards have emerged. Two major standards rise from the crowd. They are UMTS developed by the third generation partnership project (3GPP) and Code Division Multiple Access 2000 (CDMA2000) by 3GPP2.

This paper will focus on the security architecture of CDMA2000. First, an introduction to the predecessors of 3G will be given. This will be followed by an overview of 3G networks. Next the security architecture of CDMA2000 will be discussed. This will be compared with possible alternatives in the last chapter.

4th Twente Student Conference on IT, Enschede 30 January, 2006

2. EVOLUTION OF MOBILE TECHNOLOGIES
A lot can be learned from security problems in previous generation technologies. After all, the threats and problems that appeared with 1G and 2G technologies will most likely still apply to 3G technologies as well. Therefore, in this chapter a quick overview will be given of the mobile technologies that predate 3G.

2.1 Analog technologies
There are two generations of mobile technologies that are considered analog. These technologies are referred to as 0G and 1G. Unlike one might think the term 1G does not refer to the first mobile technologies. It actually refers to the first *cellular* mobile technologies, while 0G refers to the *pre*-cellular mobile technologies. The terminals used in 0G could hardly be called mobile however. The first models were extremely large and therefore usually mounted in a car's trunk. At a later date handhelds became available, but by then 0G was already outdated by its successor, 1G.

The key aspect that differentiates 0G from 1G is that 1G technologies use cellular networks. A cellular network is a network made up by a number of cells. Each of these cells is served by a fixed transmitter, commonly referred to as a base station. There are in fact also examples of cellular networks used in 0G technologies, but what makes 1G different from 0G is that it supports *seamless* moving between cells. This simply means that if you move out of reach of a base station while making a phone call, using a 0G technology you would get disconnected, while using a 1G technology you wouldn't notice anything at all.

Other aspects that differentiate 0G from 1G are that 0G technologies are often half-duplex (meaning that receiving and transmitting voice could not be done at the same time) and that operators had to manually switch through calls, though eventually full duplex phones and fully automated network services were introduced. An example of a commercially used 0G standard is Autoradiopuhelin (ARP) introduced in 1971 in Finland. It was considered a great success with a maximum of 35560 users in 1986, see [Wik05].

In the 1970s the networks used by 0G where becoming too heavily congested. A new analog standard was introduced, now referred to as 1G. Like 0G, 1G uses the Ultra High Frequency (UHF) radio bands. Voice transmission was done unencrypted over the radio air interface. This means that anyone with a simple scanner could listen in on phone calls. Attempts by the government to block certain frequencies from scanners obviously didn't make an end to this problem.

Aside from the privacy aspect, this vulnerability also introduced another problem. Because the transmission data is sent unencrypted, the (meager) security mechanisms and any data exchanged for it, is being exposed to hackers. An example of how this can (and has been) misused is the Advanced Mobile Phone System (AMPS) introduced in the Americas during the early 1980s, see [Raj00].

Like most other 1G technologies it had but one form of security, a very simple authentication procedure. This procedure consisted of a verification of two numbers: the Mobile Identification Number (MIN) and the Electronic Subscriber Number (ESN). This verification takes place when a mobile device *roams*[1] into a system. First a blacklist is checked to see if the mobile device should be blocked. This could be for any reason, such as a stolen phone. Next, a message is sent to the Home Location Register (HLR) to validate the combination of the MIN and ESN. Both of these numbers where transmitted unencrypted over the radio air interface. Hackers where able to eaves-drop on such transmissions and where able to use the numbers to create illegal clones with which they could successfully authenticate themselves as another subscriber. To make matters worse, many providers did not even perform the authentication of the mobile phone due to lack of standardization and performance reasons. This resulted in an incredible amount of fraudulent use of mobile networks, costing as high as 1 billion US dollars a year in the United States or 25% of the total revenues in the Far East, see [BBM02], [Raj00].

2.2 Digital Technologies

The next milestone in the development of mobile technologies was the introduction of digital data processing. With this it is possible to achieve higher voice quality as digital information is not subject to distortions. Additionally, the spectrum can now be used more efficiently by applying techniques for multiplexing. Because analog technologies use Frequency Division Multiple Access (FDMA), only one user could use a particular frequency at any time in any given cell. With 2G this inefficiency was solved by using Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA). These techniques allow several users to share the same frequency. Further information on these technologies can be found in [NK01].

More relevant to this paper are the changes to the security architecture. As many organizations have developed 2G standards, not all security architectures will be discussed in this paper. Instead a generalization will be made on a number of security aspects, based on the security architecture of two major 2G standards: GSM developed by the GSM group (Groupe Spéciale Mobile) in Europe and cdmaOne developed by Qualcomm in the United States of America.

Both standards use a challenge-response mechanism to identify users. Simply said, this means that when making a phone call, the mobile device has to compute a response to a random challenge sent by the network. This response is computed using a unique secret key stored on the device. The response can then be validated by the network, as it possesses the same key as the mobile device. This key can in turn also be used to establish encryption on transmission over the radio air interface, see [RK04].

Looking back at the problems with the analog generation, it can be concluded that at least in theory these problems are handled. The transmissions are now being encrypted for privacy and confidentiality and a better method for authentication is used. In practice however, a number of problems were identified. First, the standards trusted, to a certain extend, on the obscurity of its used algorithms. As time passed and the secrets of the algorithms leaked, it was quickly proven that the used algorithms where too weak. Second, the standards have several protocol deficiencies which can be misused to illegally authenticate a pirated phone. An important flaw is the lack of integrity protection. As only the mobile device is authenticated, but not the network, a false base station can be used by pirates to obtain authentication data from an unknowing subscriber. [BBM02]

3. 3G – AN OVERVIEW

Like with 2G technologies, several competing organizations came up with standards for 3G. Therefore, in the mid 1980s the International Telecommunications Union (ITU) in Geneva challenged the industry to create a single, worldwide standard. It came up with the concept International Mobile Telecommunications 2000 (IMT-2000). 10 years later a unanimous approval of technical specifications has been given under the brand IMT-2000. This meant that for the first time it was possible to achieve full interoperability and interworking between mobile systems. In 1999, ITU approved five radio interfaces to be used for IMT-2000. These are:

- *IMT-DS (Direct Spread)*, also known as Wideband –CDMA (W-CDMA) or Universal Terrestrial Radio Access – Frequency Division Duplex (UTRA-FDD) and used in UMTS.
- *IMT-MC (Multi Carrier)*, also known as CDMA2000.
- *IMT-TD (Time Division)*, this includes TD-CDMA and TD-SCDMA and are both standardized for use in UMTS.
- *IMT-SC (Single Carrier)*, also known as Universal Wireless Communications 136 (UWC-136) or Enhanced Data rate for GSM Evolution (EDGE).
- *IMT-FT (Frequency Time)*, also known as Digital Enhanced Cordless Telecommunications (DECT), see [ITU05].

From these five, IMT-DS (or better known as UMTS) and IMT-MC (or better known as CDMA2000) are considered the major standards. UMTS is developed in Europe and is the successor to

[1] When *roaming*, a user enters the network of a given service provider. This happens when a mobile device is turned on in the coverage area of the given service provider, but also when leaving the coverage area of one service provider and entering another.

GSM. The development is done by [3GPP]. CDMA2000 is the successor to cdmaOne and is developed in the United States of America by [3GPP2].

3.1 Security Objectives

The key security objectives stated by ITU are very simple. Any 3G standard must, at the very least, meet the following two requirements:

- 3G security must be equivalent to the fixed/ISDN network security, and
- user privacy must be maintained while roaming.

The first requirement implies that there are differences between securing wireless networks and securing fixed networks. This can be motivated by a number of reasons. First, fixed networks enjoy a physical barrier. To intercept a transmission you will need physical access to the network, while with a wireless network you only need to be in range of the network. More important, there are constraints to the nodes in a wireless network that are less apparent in fixed networks. [NK01] gives four fundamental differences:

- available *bandwith*;
- allowable *error rates*;
- *latency* and *variability*, and
- *power* constraints.

Due to these differences, the protocols and algorithms used for fixed networks often have too much overhead for effective usage in wireless networks. This creates a big challenge in the design of the security architecture. The second requirement can be motivated by the fact that when roaming, no secure connection between the network and the user is in effect yet. Subscriber data will have to be sent over this unsecured connection, which might be used to violate the users' privacy. The design is even more complicated, due to backwards compatibility with older wireless technologies.

To meet these requirements, the security models of 3GPP and 3GPP2 have been improved and this has led to the following objectives:

- Enhance the 2G security architecture in:
  - subscriber authentication;
  - radio interface encryption;
  - subscriber identity confidentiality;
  - the use of (removable) subscriber identity modules;
  - the creation of a secure application layer between the mobile phone and home network;
  - transparency of security features.
- Ensure an adequate level of protection offered to:
  - users;
  - all information generated by users;
  - resources and services provided by serving networks.
- Ensure the existence of at least one ciphering algorithm that can be used on a worldwide basis.
- Ensure an adequate standardization of security features.
- Ensure the possibility to extend security features and mechanisms, see [BBM02], [33.120].

3.2 Security Threats

In chapter 2 an overview was given of already identified problems with mobile networks. This past experience has led to a number of threats which are explicitly addressed by the framework. These are:

- *Unauthorized access to services:* In this case an intruder manages to get unauthorized access due to masquerading or misuse of access rights. An example of this would be a rogue shell attack.
- *Eavesdropping:* In this case an intruder manages to intercept a transmission. This can for instance be during voice transfer, but also during the authentication process. So this can cause privacy problems, but the data obtained might also be used to perform other attacks.
- *Manipulation of messages:* In this case an intruder manages to manipulate a transmission between two parties.
- *Disturbing or misusing network services:* In this case an intruder attacks the network services, which may lead to denial of service (DoS) or reduced availability of the service.
- *Man-in-the-middle attacks:* In this case an intruder places himself between the two parties involved in a transmission. Both parties are not aware of the intruder and think they are actually talking to each other, while the intruder talks with both parties, see [BBM02], [33.120].

As these threats are the basis for which the 3G security architecture is developed, they will be used in this paper as a basis for comparison with algorithms and protocols not standardized by 3GPP2.

4. SECURITY ARCHITECTURE

In the following figure a very basic overview of the entities involved in a typical setup of a CDMA2000 network is given.

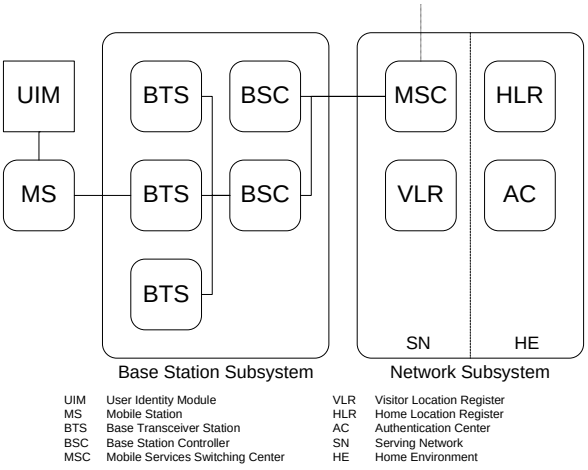

Figure 1. CDMA2000 network architecture

Left in the picture is the mobile phone. For the purpose of analysis, in this paper a difference is made between the User Identity Module (UIM) and the Mobile Station (MS). In reality, the UIM is inside the MS. The UIM can either be removable, in which case it is referred to as R-UIM, or not removable, in which case it is referred to as UIM.

The *Base Station Subsystem* is not relevant for this paper, but it is an essential part of the network. It provides a link between the MS and the Network Subsystem.

The Network Subsystem consists of a *Serving Network* (SN) and the *Home Environment* (HE). The HE is the network of a user's personal service operator. It mainly contains the home location register (HLR) and the authentication center (AC). The HLR holds a database containing subscriber data. The AC uses this data to validate users. Often the AC and HLR are coupled. The *serving network* (SN) holds a database consisting of every user currently authenticated on the network. A user can only be in one VLR at any time as he can only visit one SN at a time. The SN will communicate with the HE to exchange data used to authenticate visiting users.

4.1 Authentication

Authentication in CDMA2000 is done between the SN and the MS. Unlike with 2G technologies, the authentication is now mutual. This means that both the MS will be authenticated to the SN and the SN will be authenticated to the MS. This procedure for establishing the security association between the MS and the SN is called authentication and key agreement (AKA). Both UMTS and CDMA2000 have adopted similar protocols and procedures. The common core procedures and algorithms used by both standards are also referred to as 3G-AKA and can be found in [33.102]. CDMA2000 differs from UMTS by having an enhancement to the protocol. This enhancement helps against *rogue shell*[2] attacks. Figure 2 gives an overview of the 3G-AKA protocol with the CDMA2000 enhancement.

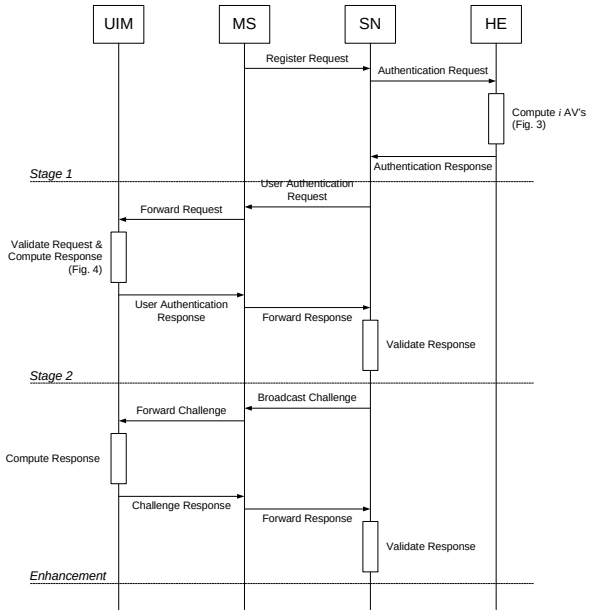

Figure 2. 3G-AKA Protocol with CDMA2000 Enhancement

4.1.1 Provisioning the Authentication Key

Before the AKA procedure can be explained we will first discuss the provisioning of the Authentication Key (further referred to as the A-Key). The A-Key is used as input for almost all of the functions in the AKA procedure. These functions are used in the AC and the UIM and therefore should be provided to both. Depending on whether an R-UIM or a UIM is used the provisioning of the A-Key can be rather complex.

In case of an R-UIM the provisioning is very similar to the process used in GSM. The manufacturing line or the service operator can program the A-Key into the R-UIM and communicate it to the AC in batches. As the R-UIM is removable it can be replaced by another when a new subscription is used for the same MS. With an irremovable UIM this is a bigger problem. It is undesirable to have a fixed combination of a UIM and a MS as this would mean you need a new phone for every new subscription you make. Therefore there has to be a way to provide the A-Key *after* production of the phone. The reason that not every UIM is removable can be for several reasons. An important reason is theft. A stolen phone with an irremovable UIM is of significantly less value than one with a R-UIM.

There are two standardized methods to provide the A-Key to the UIM and the AC. One is to have the customer or salesperson enter a special code at the time of purchase. This code is provided by the home service provider, thus the AC is already aware of it. The second, more preferred method used is over-the-air service provisioning (OTASP), see [C.S0016], [N.S0011]. In this case the customer or the salesperson calls to a special phone number at the time of purchase. During this call a Diffie-Hellman key establishment algorithm is performed between the UIM and the AC. This algorithm allows two parties to agree on a secret key over an insecure channel. For more information on the Diffie-Hellman algorithm see [RFC2631]. The Diffie-Hellman algorithm is vulnerable to a *man-in-the-middle* attack, but this can only cause an invalid authentication key. The customer will only have to repeat the OTASP. The procedure can be repeated at any time in the future. This means that any UIM can acquire a new authentication key if desired. This should only be necessary when switching service operators however.

4.1.2 Stage 1: Authentication Request

The first stage involves an authentication request by the SN to the HE. When the MS roams into the SN it will make an attempt to register itself. The SN needs an Authentication Vector (AV) to perform the mutual authentication between the MS and the SN. This AV can be requested from the HE. The SN can request more than one AV's at once, so it is possible that the SN will not have to perform this request every time. Figure 3 displays the parameters that are used to form an AV.

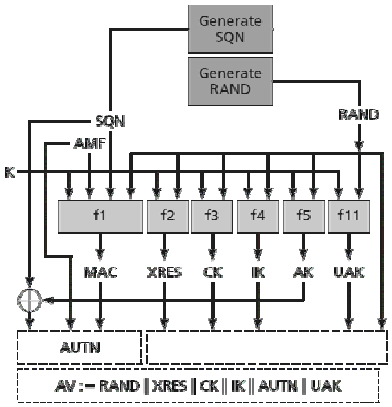

Figure 3. Authentication vector [BMM02]

[2] In the case of a *rogue shell* attack, the MS contains an illegal ciphering key (CK) and integrity key (IK). This can either be done by transmitting the CK and IK to another MS or when the MS stores the CK and IK even after removal of the R-UIM. This can in some occasions be used to illegally authenticate the MS.

This figure introduces a number of functions and acronyms used in the CDMA2000 security architecture. The tables below can be used as a reference for the acronyms and functions used in this figure. The acronyms and functions are ordered in the same sequence as they are displayed in the figure.

Table 1. Acronyms used in the AKA procedure

Term	Description
SQN	*Sequence Number*. This is an ever increasing unique number. This number is used to prevent replay attacks, where the attacker attempts to authenticate himself by replaying a recorded authentication attempt. For more information on how the SQN is computed and how it is validated see [33.102].
RAND	*Random Number*. This is a (pseudo) random number generated by f0.
AMF	*Authentication and key Management Field.* The purpose of this parameter is left up to the service provider. An example for its purpose is to allow multiple authentication algorithms and keys. For more examples on how this parameter can be used see [33.102].
A-Key (K)	*Authentication Key*. This is a key used as input for almost all functions used by the AKA procedure. The A-Key is secret and known only to the AC and the UIM. This is the key also mentioned in paragraph 4.1.1.
MAC	*Message Authentication Code.* This value is used to protect and validate the integrity and authenticity of the SQN and AMF. MAC is computed by f1.
XRES	*Expected User Response.* During stage 2 of the AKA procedure the MS will send a response back to the SN. This parameter is the expected response and will be used by the SN to validate the MS. XRES is computed by f2.
CK	*Ciphering Key*. This is a 128 bit session key that can be used to encrypt user and signaling data after the AKA procedure has finished. This key is established during the AKA procedure. CK is computed by f3. Also see paragraph 4.2 on confidentiality protection.
IK	*Integrity Key*. This is a 128 bit session key that can be used to generate a MAC for signaling messages. This key is established during the AKA procedure. IK is computed by f4. Also see paragraph 4.4 on integrity protection.
AK	*Anonymity Key.* This key is optionally used to conceal the SQN. This concealment provides a higher level of privacy protection as the SQN can be used to expose the identity and location of a user. The key is generated by f5. In the case the AK is not used f5 is 0 and the AK it generates is also 0.
UAK	*UIM Authentication Key.* This key is part of the enhancement from CDMA2000 to the 3G-AKA protocol. For every message generated by MS, the UAK function is used to compute a secure hash value (UMAC) of the MAC. Because the UAK is only stored on the UIM, this ensures that the UIM is always present and therefore solves a possible rogue shell attack. It is up to the service operator to support this feature.
AUTN	*Authentication Token.* This token is, like the AV, a concatenation of parameters. It contains all parameters the MS needs to authenticate itself with the SN. `AUTN := SQN [⊕ AK] || AMF || MAC` The concealment of SQN with AK is optional (also see the description of AK).

AV	*Authentication Vector*. This is a vector made up by a number of concatenated parameters containing data used to authenticate a roaming user.
	`AV := RAND || XRES || CK || IK ||`
	`AUTN || UAK`
	These parameters are computed by the HE and sent to the SN. The SN can use them to perform a mutual authentication with the MS.

AUTS	*Re-synchronization Token.* In case the SQN is rejected, the MS will send AUTS to the HE for a re-synchronization event. The HE can validate this parameter and use it to compute a new fresh SQN. It is concatenated from the following parameters: $AUTS := SQN_{MS} [\oplus AK] \| MAC\text{-}S$ Note the use of SQN_{MS} and MAC-S. SQN_{MS} means that the SQN known by the MS is used, while MAC-S designates the MAC generated by f1*.

The following table gives an overview of the functions used by the AKA procedure. With the exception of f0, each of these functions is located on both the AC and the UIM (f0 is not necessary on the UIM and only located on the AC). This means that these functions do not need to be standardized and the service providers are free to choose their algorithms. In all cases it should be computationally infeasible to derive the A-Key from the knowledge of the output values. An example of an algorithm set can be found in [35.206].

Table 2. Functions used by the AKA procedure [33.105]

Fn.	Description
f0	*Random challenge generation function*. This function generates (pseudo) random numbers, which are used as input for the other functions. `f0(internal state) → RAND`
f1	*Network authentication function*. This function is used by the MS to compute XMAC and by the HE to compute MAC. Both values will be compared by the MS to validate the integrity of SQN and AMF. `f1(K,SQN,RAND,AMF) → MAC (or XMAC)`

f1*	*Re-sync message authentication function.* In the rare case the SQN is rejected by the MS a re-synchronization attempt will be done. This means that the MS will send a fresh SQN to the SN, which will in turn be forwarded to the HE for validation. f1* is used to protect the integrity of the SQN send by the MS.
	`f1*(K,SQN,RAND,AMF*) →`
	`MAC (or XMAC)`
	AMF* is the default value of AMF used in re-

	synchronization attempts.
f2	*User authentication function.* This function is used by the MS to compute RES and by the HE to compute XRES. Both values will be compared by the SN to validate the authentication. `f2(K,RAND) → RES (or XRES)`
f3	*Cipher key derivation function.* This function derivates the session key CK from the A-Key. The CK is never sent to the MS by the SN. It is computed by both the UIM and the HE with the same input parameters. `f3(K,RAND) → CK`
f4	*Integrity key derivation function.* This function derivates the session key IK from the A-Key. The IK is never sent to the MS by the SN. It is computed by both the UIM and the HE with the same input parameters. `f4(K,RAND) → IK`
f5	*Anonymity key derivation function (normal).* This function derivates AK from the A-Key optionally used to conceal the SQN. It is computed by both the UIM and the HE with the same input parameters. `f5(K,RAND) → AK`
f5*	*Anonymity key derivation function (re-sync).* This function derivates AK from the A-Key in the case of a re-synchronization. It can optionally be used to conceal the SQN send by the MS. `f5*(K,RAND) → AK`
f11	*UIM authentication key derivation function.* This function derives UAK from the A-Key. UAK is used as part of the enhancement to the AKA procedure. `f11(K,RAND) → UAK`

4.1.3 Stage 2: Mutual Authentication

By now the SN has a number of AV's which can be used for authenticating the roaming user. One AV can be used for exactly one authentication attempt. The mutual authentication process is as follows:

1. The SN sends AUTN and RAND to the MS.
2. The MS forwards them to the UIM.
3. The UIM validates the integrity of AUTN and checks the freshness of SQN.
4. If the integrity and freshness are validated the UIM generates CK, IK, UAK and RES.
5. The UIM returns CK, IK and RES to the MS.
6. The MS will forward RES to the SN.
7. The SN will validate the MS by comparing RES to XRES.

A flowchart of the authentication process is given in figure 4.

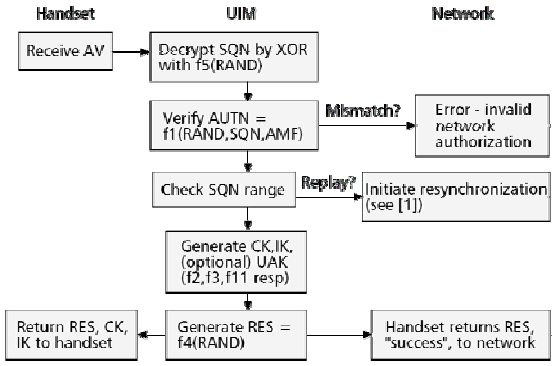

Figure 4. Authentication process in the UIM [RK04]

In three cases the authentication can fail. The first case is where the integrity of AUTN can not be validated by the MS. In this case the AKA procedure will be aborted. The same goes when RES does not match XRES. These errors will cause an invalid authorization failure.

A unique case is when the SQN cannot be validated for its freshness. This will cause a *re-synchronization failure*. In this case the MS will send the vector AUTS to the HE. AUTS can then be used by the HE to synchronize the SQN (also see the description of AUTS, f1* and f5*). This situation is very rare however and should not happen under normal circumstances.

4.1.4 Enhancement

An optional stage is part of the enhancement from CDMA2000 to the AKA procedure. In this case the UAK is used for creating a security relationship between the UIM and the SN. If the SN supports broadcast challenges it can periodically sent a challenge on which the UIM must respond. This process goes as follows:

1. The SN generates a random challenge called FRESH-A.
2. FRESH-A, along with COUNT-A (a random value used to generate the MAC) are sent to the HE and the MS.
3. The HE generates XUMAC (expected UIM-present MAC), based on FRESH-A, COUNT-A and the UAK.
4. The UIM generates UMAC, based on FRESH-A, COUNT-A and the UAK. Note that this must be done by the R-UIM, because UAK is never transferred out of the R-UIM.
5. The SN can then compare the XUMAC with the UMAC to validate in the R-UIM is still present.

Because the UAK is only stored on the R-UIM, the UIM cannot be removed from the phone while authenticating. This solves a possible rogue-shell attack.

4.2 Confidentiality

Confidentiality protection is applied to all user and signaling data between the MS and the radio network controller (RNC). This is done by using function f8 and session key CK. To facilitate worldwide roaming the ciphering function f8 is standardized and must be implemented by all MS's and SN's. For CDMA2000 the Advanced Encryption Standard (AES) is used. In [S.S0078] a reference can be found to ESP_AES, the mode of AES used in CDMA2000. Figure 5 displays how a data stream is encrypted and decrypted.

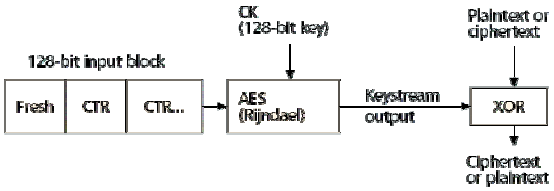

Figure 5. Confidentiality protection through AES

A "freshness" variable (Fresh) is constructed from the system time and the transmission direction. This value is concatenated with as many as 32 bit counters (CTR) as necessary to fill the block of 128 bits. The counter starts at 0 at the beginning of each frame, and is incremented for subsequent cipher blocks within that frame. This value and the CK are used as input for the block cipher operation. The output of this operation is used as a stream cipher to encrypt and decrypt the data. For more technical details on the block cipher algorithm, see [DR99] and [S.S0078].

4.3 Privacy

An already mentioned form of privacy protection is the use of anonymity keys to conceal the SQN during authentication. In some cases it is possible to expose the identity and location of a user if the SQN is known. In these cases the use of anonymity keys provides an excellent solution. It is up to the service operator to support this feature.

Privacy on user and data signaling is provided by the use of AES. While UMTS has a special mode of operation on the f8 function in order to improve privacy protection, no such feature is present in CDMA2000. The reasoning behind this is that due to the large block size and the considerable study of AES it is not necessary, see [RK04].

4.4 Integrity

User data in CDMA2000 is not integrity protected. The reasoning behind this is that if it would be important enough to protect it, it would be done at application level, see [RK04].

Signaling data on the other hand is always integrity protected. The enhancement to the 3G-AKA protocol has already been mentioned in paragraph 4.1.4. Depending on the importance of the message, it can be opted to either generate a normal MAC, based on IK, or an UMAC, based on both UAK and IK.

For the generation of MAC's, the HMAC-SHA-1 algorithm is standardized by the Internet Engineering Task Force (IETF). [RFC2404] The problem with this algorithm is that it is very inefficient for short signaling messages. Therefore CDMA2000 has adopted an optimized version of this algorithm called Enhanced Hash MAC (EHMAC), see [Pat02]. An algorithm for generating MAC's works as follows:

1. Alice wants to send message M to Bob and they want to be sure that M is not tampered with during transmission.
2. Alice and Bob agree on a secret key only known to them. In the case of CDMA2000 this key is IK (also see 4.1).
3. There are two functions in a MAC algorithm. One for signing (referred to as S) and one for validating (referred to as V).
4. Alice generates a MAC for M by using S(IK, M) and sends the MAC and M to Bob.
5. Bob validates M by computing V(IK, M, MAC).
6. Because it is computationally infeasible to find two messages M which share the same MAC, Bob can be sure that M is not tampered with if the function validates the MAC.

For more technical details on the EHMAC algorithm see [Pat02]. In case an UMAC is required instead of a MAC, the MAC is passed to the UIM for processing. The returned value is truncated to the same length as the original MAC and then returned as the UMAC value, see [RK04], [BBM02].

5. ALTERNATIVE SECURITY ALGORITHMS & PROTOCOLS

In this chapter an overview will be given of possible alternative algorithms and protocols to be used instead of the algorithms and protocols as specified by 3GPP2. Because there has already been considerable study on the AES and HMAC algorithms, the comparison will be limited to alternatives for the AKA procedure.

5.1 SAKA

As mentioned in 4.1.1, a Diffie-Hellman algorithm is used during OTASP for the provisioning of the A-Key to the R-UIM and the HE. Also mentioned was that this algorithm is vulnerable to a man-in-the-middle attack. Several algorithms have been proposed as alternative to the Diffie-Hellman algorithm. Many of these are based on the Simple Authenticated Key Agreement (SAKA) protocol proposed in [SS99].

SAKA is based on the Diffie-Hellman protocol, but it prevents a man-in-the-middle attack, while it generates an equal amount of data traffic. The difference with the Diffie-Hellman algorithm is that both parties compute two extra integers, based on a common shared password. These integers are applied to the messages sent during the algorithm, effectively preventing a man-in-the-middle attack. For more in-depth details on the algorithm see [SS99].

The problem with SAKA is that it does not provide forward secrecy. Through the use of a so-called reflection attack it is possible to compromise one of the mentioned integers. With this integer old session keys can be recovered, which can be used to decrypt previously recorded transmission. So while SAKA effectively protects the current transmission, it fails to protect past transmissions.

5.2 ECAKA

The Elliptic Curve Authenticated Key Agreement (ECAKA) protocol proposed in [SHY05] is based on SAKA and provides a solution to the man-in-the-middle-attack, but also has perfect forward secrecy. The proof of this claim goes beyond the scope of this paper, but it can be found in [SHY05].

The disadvantage of ECAKA is that, like SAKA, it requires a common shared password known in advance of the OTASP procedure. The goal of the OTASP procedure is to establish a common shared password. This means that ECAKA actually undermines the purpose of the OTASP procedure. However, as the compromise of the key required for ECAKA is much less dramatic than the compromise of the A-Key, and because the key *might* be reused in future OTASP requests, a small advantage can be gained.

A big advantage of the use of ECAKA over Diffie-Hellman or SAKA is that it uses Elliptic Curve Cryptography (ECC). This was introduced in 1985 by Neal Koblitz and Victor Miller. [Kob87], [Mil85] The advantage of ECC over conventional

Discrete Logarithm (DL) is that it provides a higher level of security with smaller parameters. For CDMA2000 this means that a 160 bit ECAKA key provides about twice the level of security as the specified 512 bit key. The smaller key size also means that it has better performance than the protocol specified by 3GPP2:

- ECAKA uses 1800 bits of bandwidth as opposed to 1700 bits used by 3GPP2. If however 3GPP2 uses a 1024 bit key to match the security level of ECAKA, it will require 3000 bits.
- ECAKA requires only 1900 bits of storage as opposed to 2400 bits for 3GPP2 and even 4000 bits for the 1024 bit key version.
- ECAKA requires 5 multiplication operations and 1 point addition operation as opposed to 2 exponentiation operations for 3GPP2. This results in a computational cost of one fourth of that of 3GPP2 when the 1024 bit key version is used.

5.3 AKA with Hash Chaining

As mentioned in 4.1.2, CDMA2000 uses a sequence number (SQN) approach to provide mutual authentication. [HH03] suggests using a combination of hash chaining and keyed HMAC techniques instead. A simplified description of the procedure is as follows:

1. During registration the MS generates a random integer seed *b*.
2. The MS then hashes *b* using a one way hash function *f* for *M* amount of times. This is the hash chain and is referred to as $f^M(b)$, where $f^M(b)$ equals $f(f(\ldots(f(b)\ldots)))$.
3. A MAC is generated from the hash chain with the HMAC algorithm and both the MAC and the hash chain are sent to the HE.
4. The HE can validate the hash chain, because the A-Key used to generate the MAC is available on both the MS and the HE.
5. The HE sends the hash chain along with the AV to the SN.
6. The SN generates a random integer seed *a*, just like the MS did in step 1 and creates a similar hash chain $g^M(a)$.
7. This hash chain is integrity protected using AK (now available from the AV) with HMAC and sent to the MS.
8. The MS can validate the hash chain, because it can generate the same AK as the SN.
9. The MS can now authenticate to SN by submitting $f^{M-1}(b)$. The SN can validate this by computing $f^M(f^{M-1}(b)) = f^M(b)$. Because an intruder can never know the previous value in the hash chain, only the legit user can authenticate. The SN can authenticate to MS in the same way by submitting $g^{M-1}(a)$.
10. For subsequent authentications both the MS and the VLR can simply remove one hash from the hash chains and repeat step 9.

The advantage of using these techniques is a reduced complexity of the protocol and its implementation, while still providing strong periodic mutual authentication and strong key agreement. It entirely removes the need for SQN's and synchronization between the HE and the MS. Additionally it provides an effective way for non-repudiation. In case of a billing dispute between a user and the SN, the hash chains can function as proof of previous visits from the user.

At the time of writing no performance analysis has been done on this protocol yet, so the performance cannot be directly compared to that of the AKA procedure used by CDMA2000.

5.4 X-AKA

Huang and Li feel that 3G-AKA has three weaknesses: bandwidth consumption between the SN and the HE, the amount of storage space required to store AV's and the need for SQN synchronization. In [HL05] they introduce the X-AKA protocol as a solution to these weaknesses.

The protocol uses a temporary key mechanism. Huang and Li suggest using f5 to generate these keys, but as the keys generated by f5 are only 48 bits long they suggest introducing a new function fx. This function should be capable of producing hash values of at least 128 bits. As CDMA2000 already has f11, which is capable of generating a 128 bits value, this may be sufficient. The use of a temporary key allows the SN, as opposed to the HE, to do the bulk of the computing. All that is required is to generate a temporary key during registration. This key is computed using fx(A-Key, RAND). Both the HE and MS can compute this key. The temporary key can then be used by the SN as A-Key, effectively reducing the need for subsequent communications with the HE.

This means that during registration of a user, less data will have to be transferred between the SN and the HE, because the SN will do the majority of the computations. In addition the SN will not have to store a number of AV's, because it can calculate them when needed. On the other hand this means that the SN will need the same algorithm set as the MS. This means that the functions f1, f2, f3, f4, f5 and f11 will need to be standardized.

5.5 AP-AKA

Zhang identifies two security risks in the 3G-AKA protocol. The first is the vulnerability to a redirection attack. In this case a false base station impersonates a SN. The false base station will then redirect all traffic to a SN of their choosing. The AKA procedure will normally succeed and the user will not notice being connected to another network. This can be abused to redirect data to an expensive SN, causing the user to get unusually high bills. Additionally it might be used to redirect traffic to networks with lower security, causing a wrong impression on the security level applied.

The second risk identified is that when a SN is corrupted, this can compromise the security of other SN's. If the attacker manages the send an authentication request through the corrupt SN, it can use the AV's to impersonate any SN. This means that one failing SN can compromise the entire system.

The Adaptive Protocol for Authentication and Key Agreement (AP-AKA) protocol proposed in [ZHA04] solves these problems by eliminating the need for SQN synchronization and providing a way for the MS to verify if AV's are indeed coming from the SN and are not used before. This is achieved by sending additional identity information from the user and the serving network along with the authentication data. The need for SQN synchronization is removed, because the user can validate the freshness of AV's directly. For further details on the implementation see [ZHA04].

6. CONCLUSION & FUTURE WORK

With the introduction of 3G the security of mobile networks has been greatly improved. Problems with previous generation technologies have been solved by providing strong mutual authentication, and through the use of time tested algorithms for

privacy, confidentiality and integrity protection. There still is room for improvement however.

When provisioning the A-Key to the R-UIM and the AC with OTASP, 3GPP2 specifies the use of a Diffie-Hellman algorithm. This algorithm is vulnerable to a man-in-the-middle attack. Sui et al suggest using their protocol named ECAKA instead. Apart from solving this vulnerability it provides a stronger security and higher performance.

Another problem is the need for SQN synchronization. Three different protocols suggest different approaches to solve this problem. Harn and Hsin introduce an enhancement to the AKA by using hash chains, resulting in reduced complexity of the protocol and its implementation. Additionally it introduces an effective method to help in billing disputes between subscribers and operators. Huang and Li propose their protocol named X-AKA, which uses temporary keys to transfer a large part of the authentication operations from the HE to the SN. Their protocol also results in decreased data traffic between the SN and the HE, and requires less storage on the SN, because it is no longer required to store authentication data. Zhang suggests using AP-AKA. This protocol effectively removes the possibility for a redirection attack and reduces the threat a corrupt SN can pose to the system.

Future work can involve a performance analysis of the mentioned protocols. Additionally research can be done on implementing CDMA2000 in the AAA model as specified by the IETF AAA Working Group, see [AAA].

REFERENCES

[33.102] The 3rd Generation Partnership Project, '3GPP TS 33.102 V6.4.0 Security Architecture', *http://www.3gpp.org*, Sep. 2005

[33.105] The 3rd Generation Partnership Project, '3GPP TS 33.105 V6.0.0 Cryptographic Algorithm Requirements', *http://www.3gpp.org*, Jun. 2004

[33.120] The 3rd Generation Partnership Project, '3GPP TS 33.120 V4.0.0 Security Principles and Objectives', *http://www.3gpp.org*, Mar. 2001

[35.206] The 3rd Generation Partnership Project, '3GPP TS 35.206 V6.0.0 An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*', *http://www.3gpp.org*, Dec. 2004

[3GPP] The 3rd Generation Partnership Project (3GPP), *http://www.3gpp.org*

[3GPP2] The 3rd Generation Partnership Project 2 (3GPP2), *http://www.3gpp2.org*

[AAA] The IETF AAA Working Group, *http://www.ietf.org/html.charters/aaa-charter.html*

[BMM02] Blumenthal, U., Marcovici, M., Mizikovsky, s., Patel, S., Sundaram, G. S., Wong, M. 'Wireless Network Security Architecture'. *Bell Labs Technical Journal*, Volume: 7, Issue: 2, Pages: 19-36, Dec. 2002

[C.S0016] The 3rd Generation Partnership Project, 'C.S0016-C v1.0: Over-the-Air Service Provisioning of Mobile Stations in Spread Spectrum Standards', *http://www.3gpp2.org*, Nov. 2004

[CN05] Cellular News, '1.5 Billion GSM Users'. *http://www.cellular-news.com/story/14014.php*, Sep. 2005

[DR99] Daemen, J., Rijmen, V. 'The Rijndael Block Cipher', *AES Proposal*, Sep. 1999

[ITU05] International Telecommunications Union, 'About mobile technology and IMT-2000'. *http://www.itu.int/osg/spu/imt-2000/technology.html*, Jan. 2005

[HH03] Harn, L., Hsin, W. J. 'On the security of wireless network access with enhancements'. *Proceedings of the 2003 ACM workshop on Wireless security*, Pages: 88-95, 2003

[HL05] Huang, C. M., Li, J. W. 'Authentication and Key Agreement Protocol for UMTS with Low Bandwidth Consumption', *Proceedings of the 19th International Conference of Advanced Information Networking and Applications*. Volume: 1, Issue: 28-30, Pages: 392-397, Mar. 2005

[Kob87] Koblitz, N. 'Elleptic Curve Cryptosystems', *Mathematics of Computation.* Volume: 48, Issue: 177, Pages 203-209, 1987

[Mil85] Miller, V. S. 'Use of Elliptic Curves in Cryptography', *Advances in Cryptology Crypto 85, Lecture Notes in Computer Science.* Springer-Verlag, Volume: 128, Pages: 417-426, 1985

[N.S0011] The 3rd Generation Partnership Project 2 (3GPP2), 'N.S0011-0 v1.0: OTASP and OTAPA', *http://www.3gpp2.org*, Jan. 1999

[NK01] Nichols, R. K., Lekkas, P. C. 'Wireless security: models, threats and solutions', *McGraw-Hill*, Dec. 2001

[Pat02] Patel, S. 'An Efficient MAC for Short Messages'. *Selected Areas in Communications, http://citeseer.ist.psu.edu/article/patel01efficient.html*, 2002

[Raj00] Rajagopalan, S. 'A Study of Security Problems Associated with the Telephone Network'. *Oregon State University, Department of Electrical and Computer Engineering*, 2000

[RFC2404] Madson, C., Glenn, R. 'The Use of HMAC-SHA-1-96 within ESP and AH' RFC2404, Nov. 1998

[RFC2631] Rescorla, E. 'Diffie-Hellman Key Agreement Method'. RFC2631, Jun. 1999

[RK04] Rose, G., Koien, G. M. 'Access security in CDMA2000, including a comparison with UMTS access security'. *Wireless Communications, IEEE*, Volume: 11, Issue: 1, Pages: 19-25, Feb. 2004

[S.S0078] The 3rd Generation Partnership Project, 'S.S0078-A v3.0: Common Security Algorithms', *http://www.3gpp2.org*, Oct. 2005

[SHY05] Sui, A. F., Hui, L. C. K., Yiu, S. M., Chow, K. P., Tsang, W. W., Chong, C. F., Pun, K. H., Chan, H. W. 'An improved authenticated key agreement protocol with perfect forward secrecy for wireless mobile communication'. *Wireless Communications and Networking Conference, IEEE*, Volume: 4, Pages: 2088-2093, Mar. 2005

[SS99] Seo, D., H., Sweeney, P. 'Simple Authenticated Key Agreement Algorithm'. *Electronic Letters*, Volume: 35, Issue: 13, Pages: 1073-1074, Jun. 1999

[UF05] UMTS Forum, 'Global 3G subscriptions hit 50 million, says UMTS Forum'. *http://www.umts-*

forum.org/servlet/dycon/ztumts/umts/Live/en/umts/News_PR_Article060905, Sep. 2005

[Wik05] Wikipedia, 'Autoradiopuhelin'. *http://en.wikipedia.org/wiki/Autoradiopuhelin*, Aug. 2005

[ZHA04] Zhang, M. 'Adaptive Protocol for Entity Authentication and Key Agreement in Mobile Networks', *Lecture Notes in Computer Science*, Volume: 2971, Pages: 166-183, 2004