TABLE OF CONTENTS

1 INTRODUCTION

1.1 Motivation

Nowadays the spread of wireless utilisation and the variability of applications cause fundamental changes in our everyday life. However the mobility involves several security problems that cannot be ignored while designing a telecommunication system. That is the reason why I chose this topic as my research field. I study in the Mobile Communications and Computing Laboratory of the Department of Telecommunications since 2003 September.

After an overview of security in existing wireless systems I will focus on a very actual topic: interworking between 3G and Wireless LAN systems. My objective is to give a detailed proposition of how security challenges could be solved in the proposed interworking model, and then to analyze it from a security point of view. Protection against different type of attacks will also be evaluated.

I will also present an OMNeT++ simulation of these security message exchanges, particularly concerning authentication, authorization and accounting problems.

At the end of my thesis I will describe some alternative solutions to general problems of service authorization.

1.2 History

Cryptography has a long and fascinating history. The most complete non-technical account of the subject is Kahn's The Codebreakers [3]. This book traces cryptography from its initial and limited use by the Egyptians some 4000 years ago, to the twentieth century where it played a crucial role in the outcome of both world wars. Completed in 1963, Kahn's book covers those aspects of the history which were most significant (up to that time) to the development of the subject. The predominant practitioners of the art were those associated with the military, the diplomatic service

and government in general. Cryptography was used as a tool to protect national secrets and strategies.

The proliferation of computers and communications systems in the 1960s brought with it a demand from the private sector for means to protect information in digital form and to provide security services. Beginning with the work of Feistel at IBM in the early 1970s and culminating in 1977 with the adoption as a U.S. Federal Information Processing Standard for encrypting unclassified information, DES, the Data Encryption Standard, is the most well-known cryptographic mechanism in history [3]. It remains the standard means for securing electronic commerce for many financial institutions around the world.

The most striking development in the history of cryptography came in 1976 when Diffie and Hellman published *New Directions in Cryptography* [4]. This paper introduced the revolutionary concept of public-key cryptography and also provided a new and ingenious method for key exchange, the security of which is based on the intractability of the discrete logarithm problem. Although the authors had no practical realization of a public-key encryption scheme at the time, the idea was clear and it generated extensive interest and activity in the cryptographic community. In 1978 Rivest, Shamir, and Adleman discovered the first practical public-key encryption and signature scheme, now referred to as RSA [5]. The RSA scheme is based on another hard mathematical problem, the intractability of factoring large integers. This application of a hard mathematical problem to cryptography revitalised efforts to find more efficient methods to factor. The 1980s saw major advances in this area but none which rendered the RSA system insecure. Another class of powerful and practical public-key schemes was found by ElGamal in 1985 [6]. These are also based on the discrete logarithm problem. One of the most significant contributions provided by public-key cryptography is the digital signature. In 1991 the first international standard for digital signatures (ISO/IEC 9796) was adopted. It is based on the RSA public-key scheme.

The search for new public-key schemes, improvements to existing cryptographic mechanisms, and proofs of security continues at a rapid pace. Various standards and infrastructures involving cryptography are being put in place. Security

products are being developed to address the security needs of an information intensive society.

1.3 Structure of My Work

In Section 2 I give an overview of existing mobile systems in order to have a reference point about security requirements. In Section 3 the basic 3G-WLAN interworking principles are detailed that I elaborated after several meetings with Balázs Bertényi. He is a Senior System Specification Engineer at NOKIA and works actually as the responsible of the 3G-WLAN Interworking Group. This Section is partly based on elaborated specifications and partly on my own ideas. In Section 4 the so-called EAP is detailed, on which the 3G-WLAN authentication scheme is based. In Section 5 a detailed protocol description of how EAP-AKA authentication could be applied in 3G-WLAN interworking environment is detailed. I based this work on the EAP-AKA standard. In Section 6 I enumerate several types of attack and observe the protection of the proposed architecture against these attacks. In Section 7 I present my 3G-WLAN interworking model that I created in OMNeT++ and the purpose of which is the simulation of the behavior of my proposed model. In Section 8 some alternative solutions are presented, that are applicable in any kind of service authorization procedure.

2 MOBILE SECURITY

2.1 New Challenges

The huge influx of mobile, handheld devices into the daily life necessitated the use of wireless techniques for communication between either two mobile devices or a stationary and a mobile device. The advantages of being mobile come in package with the disadvantages of high susceptibility to security hacks and high unreliability of transmission medium compared to the wired network. Several mechanisms have been proposed in literature to combat these problems. Data flow has to be protected against eavesdroppers, the privacy of the subscriber has to be assured, etc. Another factor that has to be considered is the limited computing capacity of the mobile devices, so the applied security algorithms cannot have a high complexity level.

In this section three mobile standards are described from a security point of view.

2.2 GSM

The security methods standardised for the GSM [1] made it one of the most secure cellular telecommunication standards currently available. Although the data confidentiality is only guaranteed on the radio channel, this is a major step in achieving end-to-end security. The subscriber's anonymity is ensured through the use of temporary identification numbers. The confidentiality of the communication itself on the radio link is performed by the application of encryption algorithms and frequency hopping which could only be realised using digital systems and signaling.

GSM uses three different security algorithms called A3, A5, and A8. In practice, A3 and A8 are generally implemented together (known as A3/A8). A3/A8 algorithm is implemented in Subscriber Identity Module (SIM) cards and in GSM network Authentication Centers. It is used to authenticate the customer and generate a key for

encrypting voice and data traffic. Development of A3 and A8 algorithms is considered a matter for individual GSM network operators, although example implementations are available. An A5 encryption algorithm scrambles the user's voice and data traffic between the handset and the base station to provide privacy. A5 algorithm is implemented in both the handset and the base station subsystem.

2.3 UMTS

UMTS [2] revolutionized telecommunications technology by offering mobile user content rich services, wireless broadband access to Internet, and worldwide roaming. The user is able to enjoy Voice-over-IP, multimedia messaging, and video conferencing services with up to 2 Mbps data rate. However mobile users and providers must be assured of the correct identity of the communicating party: user and signalling data must be protected with confidentiality and integrity mechanisms.

GSM security is based on the assumption that the core network is adequately secure and trustworthy for the transmission of confidential information in the clear from the home environment to the serving network. For future systems like UMTS this assumption may not be adequate anymore.

New attacks in the access network that were not possible when GSM was designed are feasible now, because intruders have more computational capabilities or new equipment has become available to attackers. Therefore additional security features have to be offered by the mechanisms utilised in UMTS to protect the access network. These additional features include enhancements in user identity confidentiality mechanisms, enhancements in the authentication and key agreement mechanisms to assure the freshness of the agreed keys (used e.g. to provide confidentiality or integrity) also to the user, or to assure the integrity of certain signalling messages to prevent sophisticated attacks [8].

The main security elements that are from GSM:

- Authentication of subscribers
- Subscriber identity confidentially

- Subscriber Identity Module (SIM) to be removable from terminal hardware
- Radio interface encryption

Additional UMTS security features:

- Security against using false base stations with mutual authentication
- Encryption extended from air interface to include Node B to radio network controller connection
- Security data in the network is protected in data storage and while transmitting ciphering keys and authentication data in the system
- Mechanism for upgrading security features

2.4 Wireless Local Area Networks

Wireless Local Area Networks (WLAN) are widely used in our every day life. Users are adopting the technology to save time and cost of running wires in providing high speed network access. The IEEE 802.11 is the most widely used WLAN standard, but it suffers from the weakness of its security protection.

The 802.11-1999 authentication mechanism works at the data link layer (MAC layer). Two authentication methods exist: open system authentication and shared key authentication [9]. Open system authentication is in principle a null authentication scheme and accepts anyone that requests authentication. Shared key authentication is a challenge-response authentication based on a shared secret. The user equipment (UE) sends an Authentication request to the Access Point (AP). The Access Point sends a chosen plaintext string to the station and the station responds with the WEP-encrypted string. If the string is correctly encrypted the AP sends an Authentication message to the UE to indicate that the authentication was successful. The standard allows for up to four keys in a cell but in practice all communication parties in the cell share the same key. The authentication is not mutual, only the UEs are authenticated. Shared key authentication is very weak. An attacker that listens to a successful authentication exchange will have all elements that are needed to successfully perform

an authentication of his/her own, even if the shared key is unknown. Today shared key authentication is not considered useful [19].

2.4.1 IEEE 802.1X and EAP

The 802.11i Task Group (TGi) within IEEE is working on enhancements to the 802.11 security [10]. It has been decided to use IEEE 802.1X as the authentication framework. IEEE 802.1X in turn uses the Extensible Authentication Protocol (detailed later) that allows for end-to-end mutual authentication between a UE and an Authentication Server.

Thus, even though 802.11i still performs access control on layer 2, the authentication message exchange is not restricted to the MAC layer but uses other IEEE standard as well as IETF standards. IEEE 802.1X is a standard for port-based access control. IEEE 802.1X can be described to lie between the MAC layer and higher layers and takes care of filtering off frames to/from non-authenticated stations. Before authentication is completed, only EAP-traffic is allowed to pass. This filtering procedure is shown in Figure 2.1. This allows an authentication exchange to cross the Access Point before general data is allowed to pass. When the 802.1X entity in the Access Point is informed that a UE has successfully authenticated, the AP starts to forward data packets to/from that station.

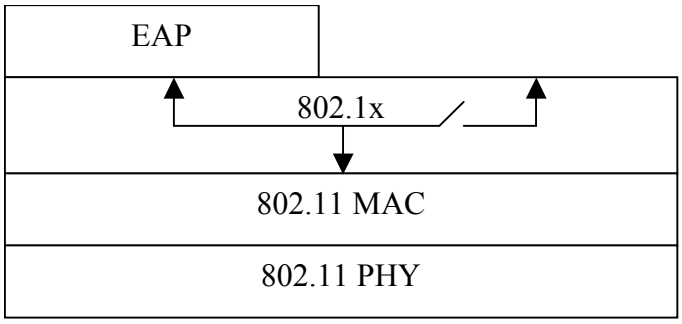

Figure 2.1. IEEE 802.1X in part of protocol stack in AP or mobile station.

EAP allows for end-to-end authentication between a UE and an Authentication Server (AS). In 802.1X a special frame format called EAP over WLAN (EAPOW) is

defined for sending EAP messages over 802 links. This allows EAP messages to be sent over the WLAN before higher layer protocols are initiated.

2.4.2 Remote Authentication Dial In User Service (RADIUS)

The Remote Authentication Dial In User Service (RADIUS) protocol was originally defined to enable centralized authentication, authorization, and access control (AAA) for SLIP and PPP dial-up sessions [7]. Instead of requiring every Network Access Server to maintain a list of authorized usernames and passwords, RADIUS Access-Requests were forwarded to an Authentication Server, commonly referred to as an AAA Server (Authentication, Authorization, and Accounting). This architecture made it possible to create a central user database, consolidating decision-making at a single point, while allowing calls to be supported by a large, physically distributed set of access servers.

The actual EAP authentication takes place between the User Equipment (UE) and the Authentication Server (AS) and is in principle transparent to the Access Point. The AP only has to forward EAP messages: EAPOW-encapsulated on the wireless side and e.g. RADIUS-encapsulated on the wired side (See Figure 2.2). If authentication is successful, the AS sends a RADIUS-Access Accept message to the AP (in the case RADIUS is used as AAA protocol). The AP then knows that the UE has been authenticated and can start forwarding traffic to/from the UE. After reception of the Access-Accept message from the AS, the AP sends an EAP-Success message to the UE.

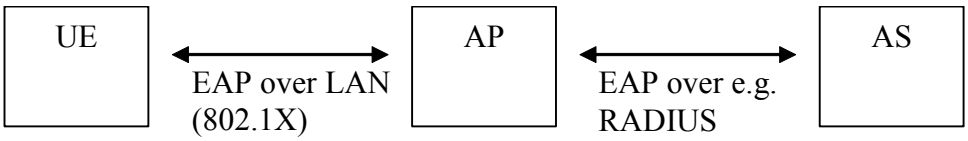

Figure 2.2. End-to-End authentication using EAP

It is out of the scope of 802.11i to specify a certain AAA protocol. IEEE 802.11i can in principle also be used without AAA protocol if the EAP method is implemented in the AP.

2.4.3 Diameter

Diameter protocol is another framework for carrying authentication, authorization and accounting information between the network access server and the AAA Server [11]. Nowadays the application of RADIUS protocol is most widely used than Diameter, but for compatibility reasons the capacity of transition from one protocol to the other is indispensable.

3 3G-WLAN INTERWORKING

3.1 Reasons of Interworking

Third-generation (3G) systems will improve data capacity and offer data rates up to 2 Mb/s and above, but it is unlikely that the user will be offered the full theoretical capacity since deployment cost is high. This is in contrast to WLAN systems, which provide affordable services and bit rates that easily exceed 3G bit rates. Ideally, we would want the subscription management, roaming, and security facilities of a 3G system and the hot spot capacity and low investment cost of WLAN systems. Integration of the two systems therefore aims to combine them such that their best features are kept intact and their weaknesses mediated by the companion system. An important challenge is to reconcile and consolidate the security architectures of the systems.

3.2 Interworking Requirements and Principles

First of all, some basic principles have to be discussed in order to have an idea of the realization of 3GPP-WLAN interworking. In this section these requirements are discussed. I based this part on the latest version of the technical specification of the ETSI 3GPP group [12]:

- Legacy WLAN terminals should be supported. However software upgrades may be required for security reasons.
- Minimal impact on the user equipment, i.e. client software.
- Minimal impact on existing WLAN networks.
- The need for operators to administer and maintain end user software shall be minimized.
- Existing SIM and USIM shall be supported.

- Authentication shall rely on (U)SIM based authentication mechanisms.
- Changes in the HSS/HLR/AuC shall be minimized.
- Methods for key distribution to the WLAN access network shall be supported.
- WLAN Access Authorization shall occur upon the success of the authentication procedure. It shall take into account the user's subscription profile and optionally information about the WLAN AN (WLAN Access Network), such as WLAN AN operator name, WLAN AN location information (e.g., country, telephone area code, city), WLAN AN throughput (e.g., maximum and minimum bandwidth guarantees for both ingress and egress traffic). This information is used to enable use-case scenarios like location based authentication/authorization, location based billing / customer care, and location based service offerings.
- It shall be possible to indicate to the user of the results of authorization requests.
- Results of WLAN Access Authorization requests shall be indicated to the WLAN, so that the WLAN can take appropriate action.
- The WLAN Access Authorization mechanism shall be able to inform the user and WLAN immediately of any change in service provision.

3.3 Access Control Principles

- **End to End Authentication:** WLAN Authentication signaling is executed between WLAN UE and 3GPP AAA Server for the purpose of authenticating the end-user and authorizing the access to the WLAN and 3GPP network.
- **Transporting Authentication signalling over WLAN Radio Interface:** WLAN authentication signaling is carried between WLAN UE and WLAN AN by WLAN Access Technology specific protocols. To ensure multivendor interoperability these WLAN technology specific protocols shall conform to existing standards of the specific WLAN access technology.
- **Transporting Authentication signaling between WLAN AN and 3GPP network:** WLAN Authentication signalling shall be transported between any

WLAN AN and 3GPP network by a standard protocol, which is independent of the specific WLAN technology utilised within the WLAN Access network.

- **WLAN Access Authorization:** This defines the process(es) in 3GPP AAA Server verifying whether WLAN Access should be allowed to a subscriber and deciding what access rules/policy should be applied to a subscriber. It is the stage after access authentication, but before service authorisation and WLAN UE's local IP address allocation.
- After the authentication process succeeds, there could be additional conditions for the 3GPP AAA Server to decide whether the access is allowed and what access rules/policy should be applied. These conditions may be based on the subscriber's profile, the account status, local agreements or information about the WLAN AN.
- Access rules/policy decided by the 3GPP AAA Server may be deployed in the 3GPP AAA Server, or/and in other entities such as the WAG (WLAN Access Gateway) or the WLAN AN.
- Access rules/policy may include access scope limitation, time limitation, bandwidth control values, and/or user priority.
- WLAN Access rules/policy should be specified by the home and/or visited operator based on the subscriber's profile, the account status, O&M rules (e.g. blacklist, access limitation list), and local agreements. Factors such as access time and access location could also be considered in these rules.
- Different access priority or the range of priorities may be authorized for different subscribers, and/or for one subscriber based on different access time or location, etc.

3.4 Interworking Scenarios

In this chapter the possible 3GPP-WLAN interworking scenarios are described. Each scenario realises an additional step in integrating WLAN in the 3GPP service offering and naturally includes the level of integration of the previous scenario [13].

3.4.1 Scenario 1 – Common Billing and Costumer Care

This is the simplest scheme of 3GPP -WLAN interworking. The connection between the WLAN and the 3GPP system is that there is a single customer relationship. The customer receives one bill from the mobile operator for the usage of both 3GPP and WLAN services. Integrated Customer Care allows for a simplified service offering from both the operator and the subscriber's perspective. The security level of the two systems may be independent. This scenario does not pose any new requirements on 3GPP specifications.

Use case:

Mr XY is a 3GPP subscriber who would like to access the WLAN service provided by his home operator. XY wants the charges for the WLAN usage included on his 3GPP service bill. XY's home 3GPP operator provides him with a user name and password to access the WLAN. He has access to Internet services and resources from the WLAN but does not have access to 3GPP services or resources other than those he can normally access from the Internet.

3.4.2 Scenario 2 - 3GPP system based Access Control and Charging

This is the scenario where authentication, authorization and accounting are provided by the 3GPP system. The security level of these functions applied to WLAN is in line with that of the 3GPP system. This ensures that the user does not see significant difference in the way access is granted. This may also provide means for the operator to charge access in a consistent manner over the two platforms.

Reusing the 3GPP system access control principles allows for additional benefits seen from a user and 3GPP system operator standpoint. First, the 3GPP system operator may easily allow subscribers within his existing 3GPP system customer base to access the WLAN with a minimum effort both for the subscriber and the operator. In addition, the maintenance of the subscriber may also be simplified.

No requirements are put upon the set of services to be offered in the WLAN part beyond those inherently offered by being addressable in an IP network.

Use case:

Mr. XY is 3GPP subscriber who needs a more secure way of accessing the WLAN than user name and password. XY's home 3GPP operator modifies his 3GPP user profile to include WLAN access and XY purchases a WLAN NIC (Network Interface Card) equipped with a UICC (Universal Integrated Circuit Card) associated with his 3GPP account. XY is authenticated on the WLAN from the credentials on the UICC but does not have access to 3GPP services other than those he can normally access from the Internet.

Mr. ZW is a 3GPP subscriber and wants to access 3GPP packet switched services and WLAN service without having to swap NIC's in his laptop. ZW purchases a dual mode (3GPP/WLAN) NIC. ZW can access 3GPP and WLAN service using separate sessions without changing any hardware.

3.4.3 Scenario 3: Access to 3GPP system based services

The goal of this scenario is to allow the operator to extend 3GPP system based services to the WLAN. Even though this scenario allows access to all services, it is an implementation question whether only a subset of the services is actually provided. However, service continuity between the 3GPP system part and the WLAN part is not required.

Use case:

Mr. XY is a 3GPP subscriber and wants to access to his 3GPP packet switched services, e.g MMS that he cannot normally access through the Internet. XY has a dual mode NIC in his laptop and is able to receive his MMS through the WLAN or 3GPP system.

3.4.4 Scenario 4: Service Continuity

The goal of this scenario is to allow the services supported in Scenario 3 to survive a change of access between WLAN and 3GPP systems. The change of access may be noticeable to the user, but there will be no need for the user/UE to re-establish

the service. There may be a change in service quality as a consequence of the transition between systems due to the varying capabilities and characteristics of the access technologies and their associated networks. It is also possible that some services may not survive, as the continuing network may not support an equivalent service. Change in service quality may be a consequence of mobility between radio access technologies, due to varying capabilities and characteristics of radio access technologies.

Use case:

Mr. XY is a 3GPP subscriber who travels frequently and has a PDA (Personal Digital Assistant) equipped with a WLAN and 3GPP transceiver. XY would like to be able to move freely about airports and hotels without having to establish a 3GPP session when he moves out of range of the WLAN. XY's PDA can switch between 3GPP and WLAN as required based on the parameters (e.g. QoS) in his profile on the same session. However, XY may experience brief interruptions in data flow during the transitions between 3GPP and WLAN.

Table 3.1 describes the service and operational capabilities of each scenario.

Scenarios **Service and operational capabilities**	Scenario 1: Common Billing and Customer Care	Scenario 2: 3GPP system based Access Control and Charging	Scenario 3: Access to 3GPP system based services	Scenario 4: Service continuity
Common billing	X	X	X	X
Common customer care	X	X	X	X
3GPP system based Access Control		X	X	X
3GPP system based Access Charging		X	X	X
Access to 3GPP system based services from WLAN			X	X
Service Continuity				X

Table 3.1. Scenarios and their capabilities

3.5 Service and Operational Capabilities

3.5.1 Common billing

The user receives the bill for the services consumption on either platform in a coordinated way. However, it does not include any requirement to harmonize the tariff structure or level of services on the two platforms.

3.5.2 Common customer care

The user does not have to bother about which platform that might have caused his need to consult the customer care.

3.5.3 3GPP system based Access Control

The user faces control procedures (authentication and authorization) similar for WLAN as within the 3GPP domain.

3.5.4 3GPP system based Access Charging

This capability enables that the 3GPP charging mechanism can be reused for WLAN.

3.5.5 Access to 3GPP system based services from WLAN

The user is offered access to the same based services over WLAN as may be accessed via the 3GPP system.

3.5.6 Service continuity

Services will survive the process of change of access network technology between WLAN and a 3GPP system (not elaborated yet).

3.6 Ownership

Ownership of the WLAN to be interworked with a 3GPP network may be one or more of the following general classes [9]:

1) The WLAN owner is a 3GPP system operator.
2) The WLAN owner is a public network operator who is not a 3GPP system operator. This may include, for example, fixed network operators, operators of mobile networks other than 3GPP systems or public WLAN operators.
3) The WLAN owner is an entity providing WLAN access in a local area (i.e. building manager/owner or airport authority) but who is otherwise not a public network operator. In this class it may be considered that a primary purpose of the WLAN operations is to provide local services and internet access as well as WLAN interworking.
4) The WLAN owner is a business entity that may be providing a WLAN for its internal use that also wishes to allow interconnection, and possibly visitor use, for some or all of their WLANs. The entity may have more than one WLAN in

operation in a location of which some may be interworked to 3GPP systems and some may not be interworked. In this class it may be considered that the primary purpose of the WLAN operations is for its own business and WLAN interworking is a secondary consideration.

This is not intended to be a restrictive list, but rather, illustrative of possibilities. There are many other possible combinations.

3.7 Interworking Architecture

First of all two cases have to be differentiated: if the owner of the used WLAN service is the user's 3GPP operator, he does not need to use roaming protocol. Contrary, if the used WLAN network belongs to another 3GPP operator, the roaming model has to be considered [12]. The two schemes are shown in Figure 3.1 and Figure 3.2.

3.7.1 Non Roaming Interworking Architecture

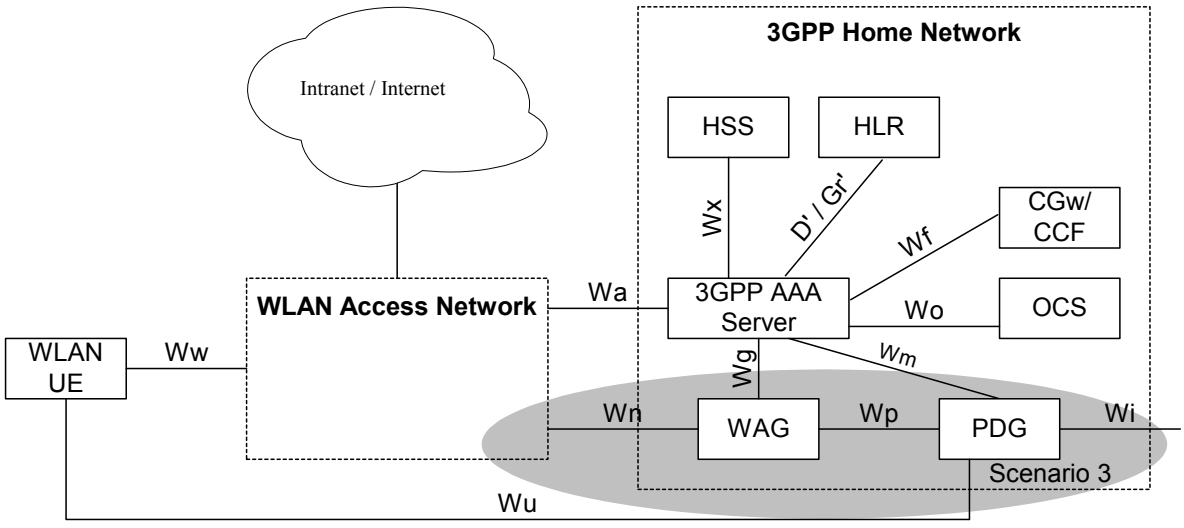

Figure 3.1. Non roaming interworking architecture

3.7.2 Roaming Interworking Architecture

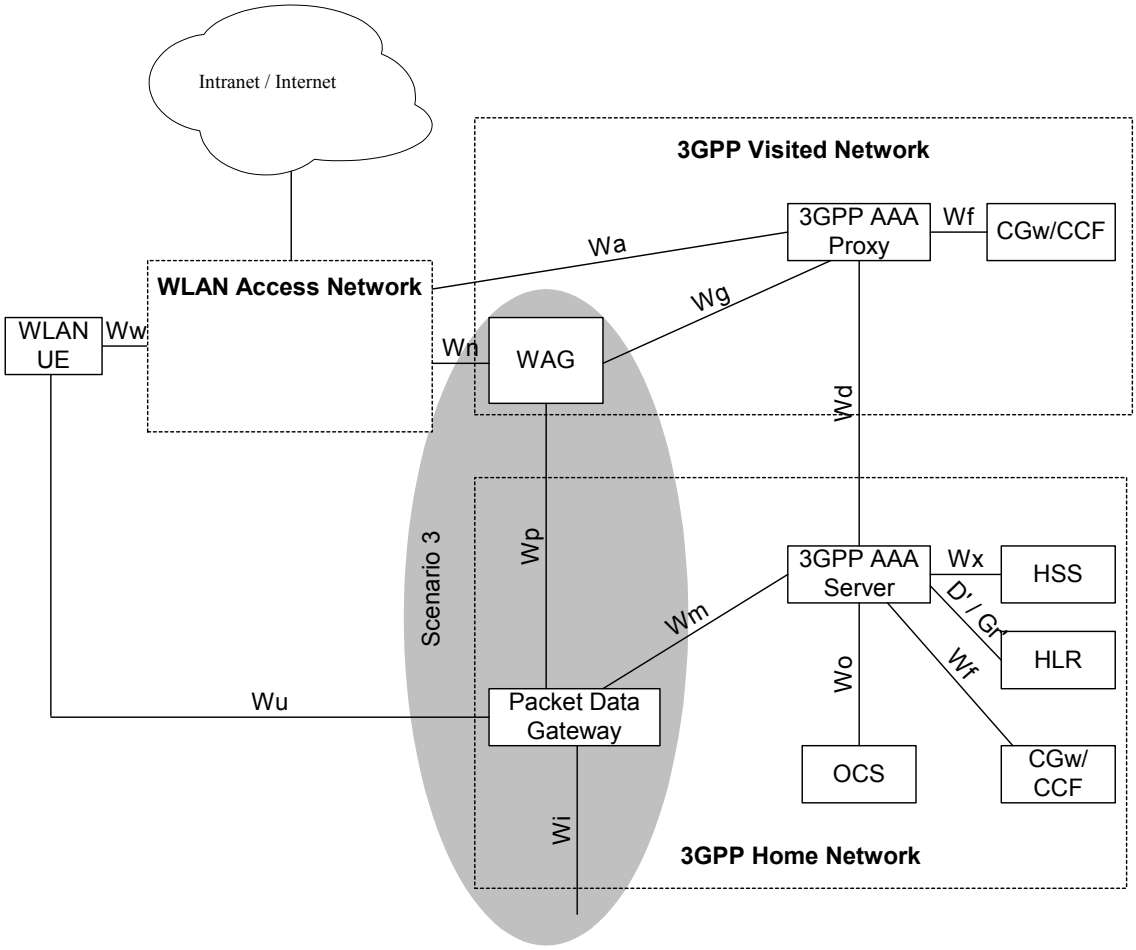

Figure 3.2. Roaming interworking architecture

3.8 Description of the Interworking Elements

On the left side of Figure 3.1 we can see the User Equipment which is in the area of WLAN coverage. The Internet access of the user is provided by the WLAN network according to scenario 2 and by the mobile operator according to scenario 3. The WLAN Access Network is connected to the mobile core network by the Wa

reference point, the 3GPP AAA server is a property of the mobile operator. The main element of the interworking scheme is the AAA server (Authentication, Authorization and Accounting), since it is responsible for access control, authorization, authentication and accounting. In the following the functionality of each element is listed [12].

3.8.1 WLAN UE

The WLAN UE (User Equipment) is a device used by a 3GPP subscriber to access the WLAN network for interworking purpose. In practice it would consist of a mobile phone (with a SIM/SMART card inside) and a notebook or PDA (with a WLAN card) which is somehow connected to the mobile phone.

The WLAN functions include:

- WLAN access authentication based on EAP methods (discussed later)
- Selection of a suitable VPLMN in the roaming case
- Building an appropriate NAI (discussed later)
- Obtain a local IP address
- Allowing users to select the type of network access, i.e. between direct access to external IP networks from the WLAN AN and network access through PLMN

3.8.2 3GPP AAA Proxy

The 3GPP AAA Proxy represents a proxying and filtering function that resides in the Visited 3GPP Network. The 3GPP AAA Proxy functions include:

- Relaying the AAA information between WLAN and the 3GPP AAA Server
- Enforcing policies derived from roaming agreements between 3GPP operators and between WLAN operator and 3GPP operator

- Reporting per-user charging/accounting information to the VPLMN CCF/CGw for roaming users

3.8.3 3GPP AAA Server

The 3GPP AAA server is the key element in the authorization of a subscriber. It is located within the 3GPP network. The 3GPP AAA Server:

- Retrieves authentication information and subscriber profile (including subscriber's authorization information) from the HLR/HSS of the 3GPP subscriber's home 3GPP network
- Authenticates the 3GPP subscriber based on the authentication information retrieved from HLR/HSS. (The authentication signaling may pass through AAA proxies.)
- Communicates authorization information to the WLAN potentially via AAA proxies.
- Registers the identity of each authenticated 3GPP subscriber
- Generates and reports per-user charging/accounting information to the HPLMN CCF/CGw

3.8.4 HLR/HSS

The HLR/HSS located within the 3GPP subscriber's home network is the entity containing authentication and subscription data required for the 3GPP subscriber to access the WLAN interworking service.

3.8.5 WLAN Access Gateway (WAG)

The WLAN Access Gateway is a gateway via which the data to/from the WLAN Access Network shall be routed via a PLMN to provide a WLAN UE with 3G based services. The WLAN Access Gateway shall reside in the VPLMN in the roaming case, and in the HPLMN in the non-roaming case. Since the WAG does not have a full trust relationship with the WLAN UE, it is not able to stop all messages. However, messages from an unknown IP address can easily be discarded. Additional types of message screening are left to the operators' control.

3.8.6 Packet data Gateway (PDG)

3GPP based services are accessed via a Packet Data Gateway. 3GPP based services may be accessed via a Packet Data Gateway in the user's Home Network or a PDG in the selected VPLMN. The process of authorisation and service selection and subscription checking determines whether a service shall be provided by the home network or by the visited network. The resolution of the IP address of the Packet Data Gateway providing access to the selected service will be performed in the PLMN functioning as the home network (in the VPLMN or HPLMN).

4 SECURITY ASPECTS OF THE INTERWORKING

4.1 General

There are number of different possible operating environments where interworking of the 3GPP system and the WLANs may be desired. The 3GPP operates universally in *Public*, *Corporate*, or *Residential* environments. WLANs may also be deployed in any of these environments and it would be advantageous if the standards for 3GPP-WLAN interworking could accommodate all of these environments. Such capability would further enhance the ease of use for the mobile system user and virtually extend the effective coverage areas of each system.

The different environments may involve different administrative domains and wide diversity of WLAN technical capabilities. As an example, the security capabilities and policies may differ between public, corporate and residential WLANs. These differences may lead to different interworking methods between 3GPP and WLANs. The environments and some of their characteristics may be summarised as follows [9]:

The ***"Public"*** environment includes all areas where there is unrestricted public presence, including outdoor areas, streets, transportation centres, retail stores, hotels, restaurants and public spaces and lobbies in major civic buildings. Here, for example, the WLAN operator is expecting general access and will likely have a system policies and equipment suitable for 3GPP - WLAN interworking.

The ***"Corporate"*** environment includes offices and factories where the users are restricted to employees of the business. Restricted visitor access may also be accommodated in this environment. The Corporate WLAN operator is providing service primarily for internal uses, and access to other networks may be screened (i.e. with a "firewall"). There may be several WLANs deployed within the corporation, not all of which are to be interworked with 3GPP. Thus, interworking between Corporate

WLAN and 3GPP may involve some different policies and techniques than for other environments.

The ***"Residential"*** environment includes individual homes and apartments where the users are typically the residents. Here, the WLAN owner and user are most likely the same. However, in a multi-tenant building, there may be a single WLAN (i.e. owned by the landlord) serving many users. The interworking of residential WLAN with 3GPP may involve some different policies that for other environments.

4.2 Extensible Authentication Protocol

The 3GPP-WLAN network authentication procedure is based on the use of Extensible Authentication Protocol (EAP). The EAP is a general protocol for Point to Point (PP) authentication which supports multiple authentication mechanisms [14]. EAP does not select a specific authentication mechanism at link control phase, but rather postpones this until the authentication phase. This allows the authenticator to request more information before determining the specific authentication mechanism. This also permits the use of a "back-end" server, which actually implements the various mechanisms while the PP authenticator merely passes through the authentication exchange. The procession of an EAP scheme is described in the following:

1. After the Link Establishment phase is complete, the authenticator sends one or more Requests to authenticate the peer. The Request has a type field to indicate what is being requested. Typically, the authenticator will send an initial Identity Request followed by one or more Requests for authentication information.
2. The peer sends a Response packet in reply to each Request. As with the Request packet, the Response packet contains a type field which corresponds to the type field of the Request.

3. The authenticator ends the authentication phase with a Success or Failure packet.

Depending on the core network of the mobile operator EAP-SIM or EAP-AKA authentication methods can be used. Though my work focuses on 3G-WLAN interworking, I think that it is important to mention the EAP-SIM method because of the incholate state of 3G network deployment in Hungary. I note that the bandwidth of WLAN provided internet connection can be high with either a 2G core network, since the data flow does not pass through the mobile radio network. In the following I will describe the functionality of the two EAP methods and show EAP-AKA authentication in details.

4.2.1 EAP-SIM Authentication

EAP-SIM authentication is based on a challenge-response mechanism [15]. The A3/A8 authentication and key derivation algorithms that run on the SIM can be given a 128-bit random number (RAND) as a challenge. The SIM runs operator-specific algorithms, which take the RAND and a secret key Ki stored on the SIM as input, and produce a 32-bit Signed Result (SRES) and a 64-bit long key Kc as output. The Kc key is originally intended to be used as an encryption key over the air interface, but in EAP-SIM protocol it is used for deriving keying material and not directly used.

The lack of mutual authentication is a weakness in GSM authentication. The 64 bit cipher key (Kc) that is derived is not strong enough for data networks. Hence in EAP-SIM, several RAND challenges are used for generating several 64-bit Kc keys, which are combined to constitute stronger keying material. In EAP-SIM the client issues a random number (NONCE_MT) to the network, in order to contribute to key derivation, and to prevent replays of EAP-SIM requests from previous exchanges. The NONCE_MT can be conceived as the client's challenge to the network. EAP-SIM also extends the combined RAND challenges and other messages with a message authentication code in order to provide message integrity protection along with mutual authentication.

EAP-SIM specifies optional support for protecting the privacy of subscriber identity using the same concept as GSM, which is using pseudonyms/temporary identifiers. It also specifies an optional fast re-authentication procedure (detailed later). In brief, EAP-SIM is in no sense weaker than the GSM mechanisms. In some cases EAP-SIM provides better security properties than the underlying GSM mechanisms. Many of the security features of EAP-SIM rely upon the secrecy of the Kc values in the SIM triplets, so protecting these values is crucial in the security of the EAP-SIM protocol.

In any case, if the GSM authentication mechanisms are considered to be sufficient for use on the cellular networks, I think that EAP-SIM is sufficiently secure for "WLAN-2G" interworking.

4.2.2 EAP-AKA Authentication

The 3rd Generation Partnership Project has specified an enhanced Authentication and Key Agreement (AKA) architecture for the Universal Mobile Telecommunications System (UMTS) [15]. The UMTS AKA mechanism includes mutual authentication, replay protection and derivation of longer session keys. EAP-AKA specifies an EAP method that is based on UMTS AKA. EAP-AKA, which is a more secure protocol, may be used instead of EAP-SIM, if USIMs and 3G network infrastructure are available.

The introduction of AKA inside EAP allows several new applications. These include the following:

- The use of the AKA also as a secure Point-to-Point authentication method in devices that already contain an USIM
- The use of the third generation mobile network authentication infrastructure in the context of wireless LANs
- Relying on AKA and the existing infrastructure in a seamless way with any other technology that can use EAP

The EAP-AKA authentication procedure uses the underlying UMTS AKA authentication, which works in the following manner:

1. The USIM and the home environment have agreed on a secret key beforehand
2. The actual authentication process starts by having the home environment produce an authentication vector, based on the secret key and a sequence number. The authentication vector contains a random part (RAND), an Authentication Token (AUTN) used for authenticating the network to the USIM, an expected result part (XRES), a session key for integrity check IK, and a session key for encryption CK.
3. The RAND and the AUTN are delivered to the USIM
4. The USIM verifies the AUTN, again based on the secret key and the sequence number. If this process is successful (the AUTN is valid and the sequence number used to generate AUTN is within the correct range), the USIM produces an authentication result, RES and sends this to the serving network.
5. The serving network verifies the result received from the USIM. If the result is correct, IK and CK can be used to protect further communications between the USIM and the serving network.

When verifying AUTN, the USIM may detect that the sequence number the network uses is not within the correct range. In this case, the USIM calculates a sequence number synchronization parameter (AUTS) and sends it to the network. AKA authentication may then be retried with a new authentication vector generated using the synchronized sequence number. In EAP-AKA, the EAP server node obtains the authentication vectors, compares RES and XRES, and uses CK and IK in key derivation.

In the third generation mobile networks, AKA is used both for radio network authentication and IP multimedia service authentication purposes. Different user identities and formats are used for these: the radio network uses the International Mobile Subscriber Identifier (IMSI), whereas the IP multimedia service uses the Network Access Identifier (NAI).

5 USE OF EAP-AKA AUTHENTICATION IN 3G-WLAN INTERWORKING

In this section I make a proposition of how EAP-AKA procedure could be applied in 3G-WLAN interworking. My proposal is based on the EAP-AKA standard [16]. Though I use declarative sentences these are only propositions and different realizations are possible.

5.1 Authentication scheme

As we have seen in Section 2.4 3G-WLAN networks use EAP-AKA authentication procedure. These authentication message exchanges are encapsulated in RADIUS or Diameter packages between the access network and the EAP server. In this interworking case there is no sense of speaking of RADIUS server, since the authenticator is the AAA server itself. Hence the importance of RADIUS is merely a compatibility question and only the message formats have to be modified in order to support legacy networks. For this reason I will focus on the Extensible Authentication Protocol in the further part of my work.

Figure 5.1 shows the functioning of the EAP-AKA procedure in 3G-WLAN interworking. The User Equipment communicates with the 3GPP AAA server and messages pass through the WLAN access network, but it is not shown in the figure. Detailed message exchange will be described in the simulation section. At the minimum, EAP-AKA uses two roundtrips to authorize the user and generate session keys.

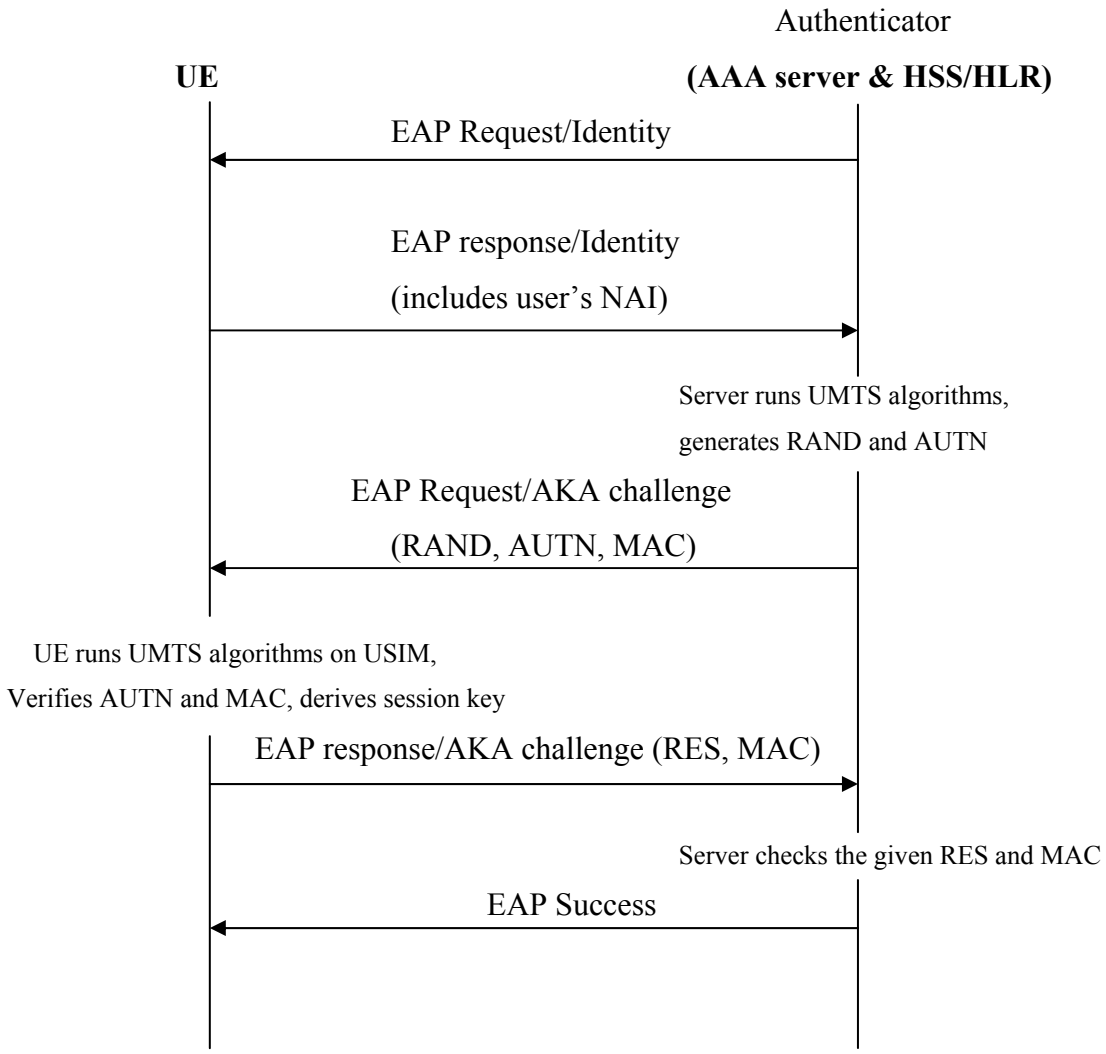

Figure 5.1. EAP-AKA authentication scheme

1. As in other EAP schemes, an identity request/response message pair is usually exchanged first. On full authentication, the UE's identity response includes either the user's International Mobile Subscriber Identity (IMSI), or a temporary identity (pseudonym).
2. After obtaining the subscriber identity, the AAA server obtains an authentication vector (RAND, AUTN, RES, CK, IK) for use in authenticating the subscriber. From the vector, the AAA server derives the keying material. The vector is obtained by contacting an Authentication Centre (AuC) on the

UMTS network; per UMTS specifications, several vectors may be obtained at a time. Vectors may be stored in the AAA server for use at a later time, but they may not be reused.

3. Next, the AAA server starts the actual AKA protocol by sending an EAP-Request/AKA-Challenge message. EAP-AKA packets encapsulate parameters in attributes, encoded in a type, length, value format. The EAP-Request/AKA-Challenge message contains a random number (RAND), a network authentication token (AUTN), and a message authentication code (MAC). The EAP-Request/AKA-Challenge message may optionally contain encrypted data, which is used for identity privacy and fast re-authentication support.
4. The UE runs the AKA algorithm (using a USIM) and verifies the AUTN. If this is successful, the UE is talking to a legitimate AAA server and proceeds to send the EAP-Response/AKA-Challenge. This message contains a result parameter that allows the AAA server to authenticate the UE, and the MAC attribute to integrity protect the EAP message.
5. The AAA server verifies that the RES and the MAC in the EAP-Response/AKA-Challenge packet are correct. Because protected success indications are not used in this example, the AAA server sends the EAP-Success packet, indicating that the authentication was successful. The AAA server may also include derived keying material in the message it sends to the authenticator. The UE has derived the same keying material, so the AAA server does not forward the keying material to the UE along with EAP-Success.

5.2 Detailed Authentication Scheme

In order to give a simplified idea of how 3G-WLAN authentication works, I did not show the initial EAP Over WLAN (EAPOW) messages between the UE and the WLAN access network. In order to reassure the reader that an attacker cannot personalize the access point, I show the detailed authentication scheme in Figure 5.2.

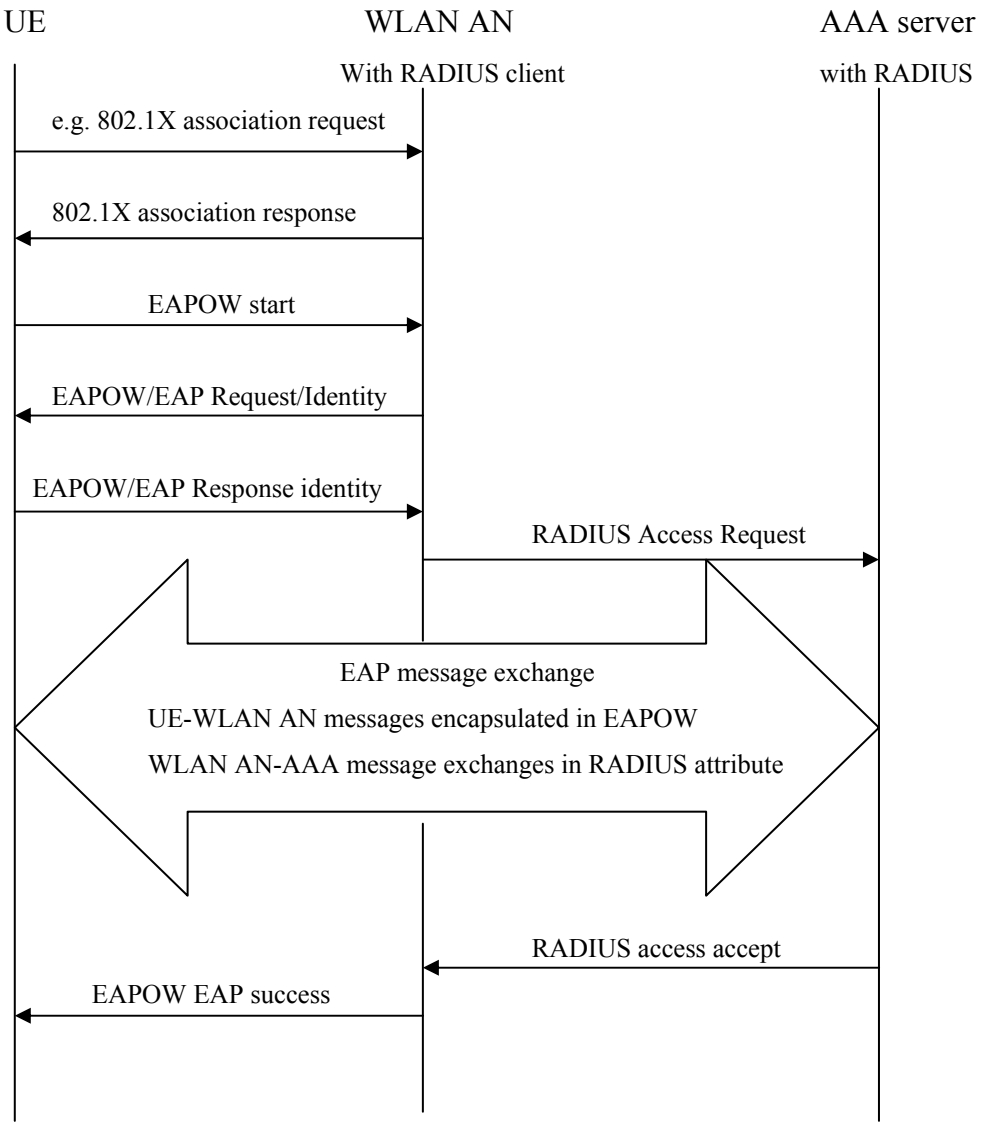

Figure 5.2. Detailed authentication scheme

As seen in Figure 5.2, after successful EAP Over WLAN (EAPOW) authentication the EAP authentication between the UE and the AAA server can really be considered as an end-to-end tunnel authentication. Hence in the further part of my work I will not consider the WLAN AN as a possible attack source.

5.3 User Identity Management

5.3.1 IMSI

At the beginning of EAP authentication, the AAA server usually issues the EAP-Request/Identity packet to the UE. The UE responds with EAP-Response/Identity, which contains the user's identity. UMTS subscribers are identified with the International Mobile Subscriber Identity (IMSI). The IMSI is composed of a three digit Mobile Country Code (MCC), a two or three digit Mobile Network Code (MNC) and a not more than 10 digit Mobile Subscriber Identification Number (MSIN). In other words, the IMSI is a string of not more than 15 digits. MCC and MNC uniquely identify the UMTS operator and help identify the AuC from which the authentication vectors need to be retrieved for this subscriber.

5.3.2 NAI Username

In order to facilitate the use of the existing cellular roaming infrastructure, the UE transmits the user's IMSI within the NAI format in the EAP Response/Identity packet. The NAI is of the format "0imsi@realm". In other words, the first character is the digit zero (ASCII 0x30), followed by the IMSI, followed by the “@” character and the realm. The IMSI is an ASCII string that consists of not more than 15 decimal digits (ASCII values between 0x30 and 0x39).

The use of a realm portion is not mandatory. If a realm is used, it may be chosen by the subscriber's home operator and it may have a configurable parameter in the EAP-AKA user implementation. In this case, the UE is typically configured with the NAI realm of the home operator. Operators may reserve a specific realm name for EAP-AKA users. This convention makes it easy to recognize that the NAI identifies a UMTS subscriber. Such reserved NAI realm may be useful as a hint as to the first authentication method to use during method negotiation. When the UE is using a pseudonym username instead of the permanent username, the UE selects the realm name portion similarly as it selects the realm portion when using the permanent username.

If no configured realm name is available, the UE may derive the realm name from the MCC and MNC portions of the IMSI.

5.4 Identity Privacy Support

5.4.1 General

EAP-AKA includes optional identity privacy (anonymity) support that can be used to hide the cleartext permanent identity and thereby to make the subscriber's EAP exchanges untraceable to eavesdroppers. Because the permanent identity never changes, revealing it would help observers to track the user. The permanent identity is usually based on the IMSI, which may further help the tracking, because the same identifier may be used in other contexts as well. Identity privacy is based on temporary identities, or pseudonyms, which are equivalent to but separate from the Temporary Mobile Subscriber Identities (TMSI) that are used on cellular networks.

There are three types of usernames in EAP-AKA UE identities: permanent usernames, pseudonym usernames and fast re-authentication usernames. The first two types of identities are only used on full authentication and the last one only on fast re-authentication (described later). When the optional identity privacy support is not used, the non-pseudonym permanent identity is used on full authentication.

5.4.2 UE Permanent Username

The non-pseudonym permanent username is derived from the IMSI. In this case, the permanent username must be of the format "0" | IMSI, where the character "|" denotes concatenation. In other words, the first character of the username is the digit zero (ASCII value 30 hexadecimal), followed by the IMSI (ASCII values between 30 and 39 hexadecimal).

For example, 0123456789098765@myoperator.com might be a valid permanent identity. In this example, 0123456789098765 is the permanent username.

5.4.3 Pseudonyms and Fast Re-authentication Identities

Pseudonym usernames and fast re-authentication identities are generated by the AAA server. They produce pseudonym usernames and fast re-authentication identities in an implementation-dependent manner. Only the AAA server needs to be able to map the pseudonym username to the permanent identity, or to recognize a fast re-authentication identity. Fast re-authentication will be detailed in section 5.5.

EAP-AKA includes no provisions to ensure that the same AAA server that generated a pseudonym username will be used on the authentication exchange when the pseudonym username is used. Hence, I think that it would be recommended that the AAA servers implement some centralized mechanism to allow all AAA servers of the home operator to map pseudonyms generated by other severs to the permanent identity. Another possible solution would be to transmit the generated pseudonyms to the HSS/HLR. If no such mechanism is available, the AAA server failing to understand a pseudonym issued by another server can request the UE to send the permanent identity, which is an event that should be minimized.

When issuing a fast re-authentication identity, the AAA server may include a realm name in the identity to make the fast re-authentication request be forwarded to the same AAA server.

When generating fast re-authentication identities, the server should choose a fresh new fast re-authentication identity that is different from the previous ones used within a same re-authentication context. The fast re-authentication identity should include a random component. The random component works as a full authentication context identifier. A context-specific fast re-authentication identity can help the server to detect whether its fast re-authentication state information matches the UE's fast re-authentication state information (in other words whether the state information is from the same full authentication exchange). The random component also makes the fast re-authentication identities unpredictable, so an attacker cannot initiate a fast re-authentication exchange to get the server's EAP-Request/AKA Re-authentication packet.

Regardless of construction method, the pseudonym username and the fast re-authentication identity must conform to the grammar specified for the username

portion of an NAI. The AAA servers that the subscribers of an operator can use must ensure that the pseudonym usernames and the username portions used in fast re-authentication identities they generate are unique.

In any case, it is necessary that permanent usernames, pseudonym usernames and fast re-authentication usernames are separate and recognizable from each other. It is also desirable that EAP-SIM and EAP-AKA user names be recognizable from each other as an aid for the server to which method to offer. In general, it is the task of the AAA server and the policies of its administrator to ensure sufficient separation in the usernames. For instance, when the usernames have been derived from the IMSI, the server could use different leading characters in the pseudonym usernames and fast re-authentication usernames. When mapping a fast re-authentication identity to a permanent identity, the server should only examine the username portion of the fast re-authentication identity and ignore the realm portion of the identity.

Because the UE may fail to save a pseudonym username sent in an EAP-Request/AKA-Challenge, for example due to malfunction, the AAA server should maintain at least the most recently used pseudonym username in addition to the most recently issued pseudonym username. If the authentication exchange is not completed successfully, then the server should not overwrite the pseudonym username that was issued during the most recent successful authentication exchange.

5.5 Fast Re-authentication

5.5.1 General

Depending on 3G-WLAN applications (public, residential or corporate), there might be need for several EAP authentications during the communication. Because the EAP-AKA full authentication procedure makes use of the UMTS AKA algorithms, and it therefore requires fresh authentication vectors from the Authentication Centre, the full authentication procedure may result in many network operations when used very frequently. Therefore, EAP-AKA includes a more inexpensive fast re-authentication procedure that does not make use of the UMTS AKA algorithms and does not need new vectors from the Authentication Centre.

Fast re-authentication is optional to implement for both the AAA server and UE. On each EAP authentication, either one of the entities may also fall back on full authentication if they do not want to use fast re-authentication.

Fast re-authentication is based on the keys derived on the preceding full authentication. The same K_aut and K_encr keys as in full authentication are used to protect EAP-AKA packets and attributes, and the original Master Key from full authentication is used to generate a fresh Master Session Key.

The fast re-authentication exchange makes use of an unsigned 16-bit counter, included in the AT_COUNTER attribute. The counter has three goals:

1) it can be used to limit the number of successive re-authentication exchanges without full-authentication
2) it contributes to the keying material, and
3) it protects the UE and the AAA server from replays.

On full authentication, both the server and the UE initialize the counter to one. The counter value of at least one is used on the first fast re-authentication. On subsequent fast re-authentications, the counter must be greater than on any of the previous fast re-authentications. For example, on the second fast re-authentication, counter value is two or greater etc. The AT_COUNTER attribute is encrypted.

Both the UE and the AAA server maintain a copy of the counter. The AAA server sends its counter value to the UE in the fast re-authentication request. The UE must verify that its counter value is less than or equal to the value sent by the AAA server. The server includes an encrypted server random nonce (AT_NONCE_S) in the fast re-authentication request. The AT_MAC attribute in the UE's response is calculated over NONCE_S to provide a challenge/response authentication scheme. The NONCE_S also contributes to the new Master Session Key.

Both the UE and the server should have an upper limit for the number of subsequent fast re-authentications allowed before a full authentication needs to be performed. Because a 16-bit counter is used in fast re-authentication, the theoretical maximum number of re-authentications is reached when the counter value reaches FFFF hexadecimal. In order to use fast re-authentication, the UE and the AAA server

need to store the following values: Master Key (MK), latest counter value and the next fast re-authentication identity. K_aut, K_encr may either be stored or derived again from MK. The server may also need to store the permanent identity of the user.

5.5.2 Fast re-authentication – UMTS-AKA Comparison

On order to understand the fast re-authentication exchange, I made a comparison with UMTS-AKA authentication procedure. The two procedures are very similar. The counter corresponds to the UMTS AKA sequence number (SQN), NONCE_S corresponds to RAND, AT_MAC in EAP-Request/AKA-Reauthentication corresponds to AUTN, the AT_MAC in EAP-Response/AKA-Re-authentication corresponds to RES, AT_COUNTER_TOO_SMALL corresponds to AUTS (Authentication Synchronization), and encrypting the counter corresponds to the usage of the AK (Anonymity Key). Also the key generation on fast re-authentication with regard to random or fresh material is similar to UMTS AKA: the server generates the NONCE_S and counter values, and the UE only verifies that the counter value is fresh. This duality is listed in Table 5.1.

Fast Re-authentication	USIM-AKA
AT_COUNTER	SQN
NONCE_S	RAND
AT_MAC in EAP request	AUTN
AT_MAC in EAP response	RES
AT_COUNTER_TOO_SMALL	AUTS

Table 5.1. Fast re-authentication – USIM-AKA comparison

5.5.3 Fast Re-authentication Identity

The fast re-authentication procedure makes use of separate re-authentication user identities. Pseudonyms and the permanent identity are reserved for full

authentication only. If a fast re-authentication identity is lost or the 3GPP network does not recognize it, the AAA server can fall back on full authentication. If the AAA server supports fast re-authentication, it may include the AT_NEXT_REAUTH_ID attribute in the encrypted data of EAP-Request/AKA-Challenge message. This attribute contains a new re-authentication identity for the next fast re-authentication. The attribute also works as a capability flag that indicates the fact that the server supports fast re-authentication, and that the server wants to continue using fast re-authentication within the current context. The UE may ignore this attribute, in which case it will use full authentication next time. If the UE wants to use fast re-authentication, it uses this fast re-authentication identity on next authentication. Even if the UE has a fast re-authentication identity, he may discard the re-authentication identity and use a pseudonym or the permanent identity instead, in which case full authentication must be performed. If the AAA server does not include the AT_NEXT_REAUTH_ID in the encrypted data of EAP-Request/AKA-Challenge or EAP-Request/AKA-Re-authentication, then the UE must discard its current fast re-authentication state information and perform a full authentication next time.

5.5.4 Fast Re-authentication Procedure

Figure 5.3. illustrates the entire fast re-authentication procedure [16]

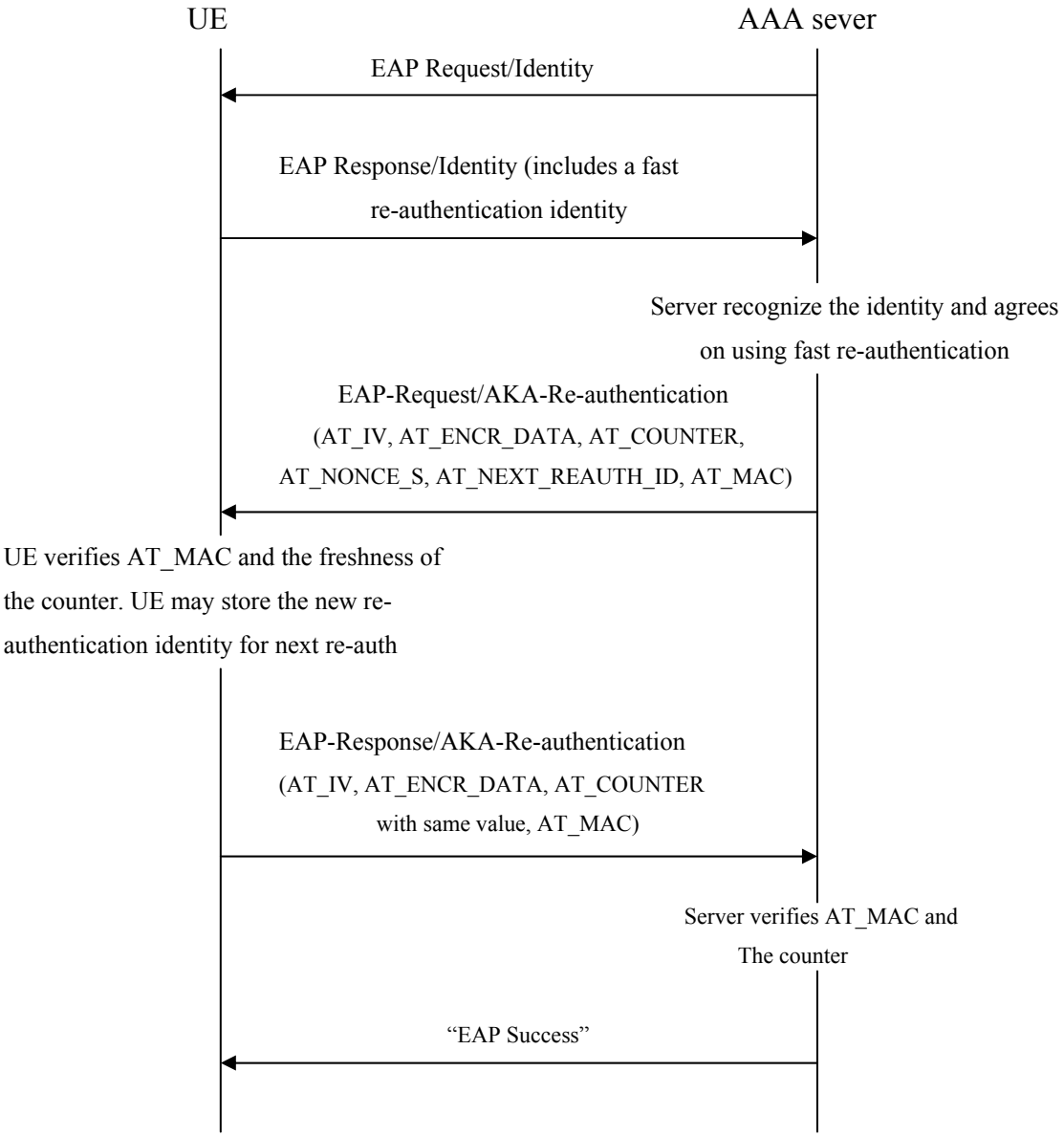

Figure 5.3. Fast re-authentication procedure

The UE uses its fast re-authentication identity in the EAP-Response/Identity packet. If the server recognizes the identity as a valid fast re-authentication identity,

and if the server agrees on using fast re-authentication, then the server sends the EAP-Request/AKA-Reauthentication packet to the UE. This packet must include the encrypted AT_COUNTER attribute, with a fresh counter value, the encrypted AT_NONCE_S attribute that contains a random number chosen by the server, the AT_ENCR_DATA and the AT_IV attributes used for encryption, and the AT_MAC attribute that contains a message authentication code over the packet. The packet may also include an encrypted AT_NEXT_REAUTH_ID attribute that contains the next fast re-authentication identity.

Fast re-authentication identities are one-time identities. If the UE does not receive a new fast re-authentication identity, it must use either the permanent identity or a pseudonym identity on the next authentication to initiate full authentication.
The UE verifies that AT_MAC is correct and that the counter value is fresh (greater than any previously used value). The UE may save the next fast re-authentication identity from the encrypted AT_NEXT_REAUTH_ID for next time. If all checks are successful, the UE responds with the EAP-Response/AKA-Re-authentication packet, including the AT_COUNTER attribute with the same counter value and the AT_MAC attribute.

The AAA server verifies the AT_MAC attribute and also verifies that the counter value is the same that it used in the EAP-Request/AKA-Re-authentication packet. If these checks are successful, the fast re-authentication has succeeded and the server sends the EAP-Success packet to the UE.

5.6 Table of Attributes

On order to have an oveview of the EAP-AKA messages, I listed the used message types and attributes (Table 5.2). Messages are denoted with numbers in parentheses as follows:

(1) EAP-Request/AKA-Identity
(2) EAP-Response/AKA-Identity
(3) EAP-Request/AKA-Challenge

(4) EAP-Response/AKA-Challenge
(5) EAP-Request/AKA-Notification
(6) EAP-Response/AKA-Notification
(7) EAP-Response/AKA-Client-Error
(8) EAP-Request/AKA-Re-authentication
(9) EAP-Response/AKA-Re-authentication
(10) EAP-Response/AKA-Authentication-Reject
(11) EAP-Response/AKA-Synchronization-Failure.

The column denoted with "E" indicates whether the attribute is a nested attribute that must be included within AT_ENCR_DATA. (Y: yes, N: no)

"0" indicates that the attribute must not be included in the message,
"1" indicates that the attribute must be included in the message,
"0-1" indicates that the attribute is sometimes included in the message, and
"0*" indicates that the attribute is not included in the message in cases specified, but may be included in the future versions of the protocol.

Attribute	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	E
AT_PERMANENT_ID_REQ	0-1	0	0	0	0	0	0	0	0	0	0	N
AT_ANY_ID_REQ	0-1	0	0	0	0	0	0	0	0	0	0	N
AT_FULLAUTH_ID_REQ	0-1	0	0	0	0	0	0	0	0	0	0	N
AT_IDENTITY	0	0-1	0	0	0	0	0	0	0	0	0	N
AT_RAND	0	0	1	0	0	0	0	0	0	0	0	N
AT_AUTN	0	0	1	0	0	0	0	0	0	0	0	N
AT_RES	0	0	0	1	0	0	0	0	0	0	0	N
AT_AUTS	0	0	0	0	0	0	0	0	0	0	1	N
AT_NEXT_PSEUDONYM	0	0	0-1	0	0	0	0	0	0	0	0	Y
AT_NEXT_REAUTH_ID	0	0	0-1	0	0	0	0	0-1	0	0	0	Y
AT_IV	0	0	0-1	0*	0-1	0-1	0	1	1	0	0	N
AT_ENCR_DATA	0	0	0-1	0*	0-1	0-1	0	1	1	0	0	N
AT_PADDING	0	0	0-1	0*	0-1	0-1	0	0-1	0-1	0	0	Y
AT_CHECKCODE	0	0	0-1	0-1	0	0	0	0-1	0-1	0	0	N
AT_RESULT_IND	0	0	0-1	0-1	0	0	0	0-1	0-1	0	0	N
AT_MAC	0	0	1	1	0-1	0-1	0	1	1	0	0	N
AT_COUNTER	0	0	0	0	0-1	0-1	0	1	1	0	0	Y
AT_COUNTER_TOO_SMALL	0	0	0	0	0	0	0	0	0-1	0	0	Y
AT_NONCE_S	0	0	0	0	0	0	0	1	0	0	0	Y
AT_NOTIFICATION	0	0	0	0	1	0	0	0	0	0	0	N
AT_CLIENT_ERROR_CODE	0	0	0	0	0	0	1	0	0	0	0	N

Table 5.2. EAP-AKA message types and attributes

6 PROTECTION AGAINST POSSIBLE ATTACKS

In this chapter I would like to give a (non exclusive) list of possible attacks against the 3G-WLAN interworking system. For some possible attacks I will analyze the level of protection of the 3G-WLAN interworking system. I based this protection analyze on the description of EAP-AKA procedure [16].

First of all it is useful to distinguish two classes of attacks: **passive attacks** in which an adversary eavesdrops on messages sent between honest users and **active attacks** (i.e., "man-in-the-middle" attacks) in which – in addition to eavesdropping – the adversary inserts, deletes, or arbitrarily modifies messages sent from one user to another. Passive attacks are well characterized (the adversary's choices are inherently limited) and techniques for achieving security against passive attacks are relatively well understood. On the other hand, active attacks are not well characterized and precise modeling is difficult. Few techniques exist for dealing with active attacks, and designing practical protocols secure against such attacks remains a challenge.

6.1 Identity Protection

EAP-AKA includes optional Identity privacy support that protects the privacy of the subscriber identity against passive eavesdropping. EAP-AKA authentication protocol only specifies a mechanism to deliver pseudonyms from the AAA server to the UE as part of an EAP-AKA exchange. Hence, a UE that has not yet performed any EAP-AKA exchanges does not typically have a pseudonym available. In this case the privacy mechanism cannot be used, but the permanent identity will have to be sent in the clear. I think that if the UE were able to store the pseudonym in a non-volatile memory, so that it can be maintained across reboots, it would considerably enhance the security level of the first authentication roundtrip. On the opposite case, an active attacker that impersonates the network may use the AT_PERMANENT_ID_REQ attribute to learn the subscriber's IMSI. In this case, there should be two different policies that the UE can employ with regard to AT_PERMANENT_ID_REQ: A

"conservative" UE assumes that the 3GPP network is able to maintain pseudonyms robustly. Therefore, if a conservative UE has a pseudonym username, it responds with EAP-Response/AKA-Client-Error to the EAP packet with AT_PERMANENT_ _ID_REQ, because he believes that the valid network is able to map the pseudonym identity to the user's permanent identity. (Alternatively, the conservative UE may accept AT_PERMANENT_ID_REQ in certain circumstances, for example if the pseudonym was received a long time ago.) The benefit of this policy is that it protects the UE against active attacks on anonymity. On the other hand, a "liberal" UE always accepts the AT_PERMANENT_ID_REQ and responds with the permanent identity. The benefit of this policy is that it works even if the valid 3GPP network sometimes loses pseudonyms and is not able to map them to the permanent identity. This type of attack can be effectively prevented by checking the AT_MAC attribute.

6.2 Flooding the Authentication Centre

The AAA server obtains authentication vectors from the Authentication Centre (AuC). It should be noted that a malicious user may generate a lot of protocol requests to mount a denial of service attack. The AAA server implementation should take this into account and should take steps to limit the traffic that it generates towards the AuC, preventing the attacker from flooding the AuC and from extending the denial of service attack from EAP-AKA to other users of the AuC.

6.3 Key Derivation

EAP-AKA supports key derivation with 128-bit effective key strength. The Transient EAP Keys used to protect EAP-AKA packets (K_encr, K_aut) and the Master Session Keys are cryptographically separate. An attacker cannot derive any non-trivial information from K_encr or K_aut based on the Master Session Key or vice versa. An attacker also cannot calculate the pre-shared secret from the UMTS AKA IK, UMTS AKA CK, EAP-AKA K_encr, EAP-AKA K_aut or from the Master Session Key.

6.4 Brute-Force and Dictionary Attacks

The effective strength of EAP-AKA values is 128 bits, and there are no known computationally feasible brute-force attacks. Because UMTS AKA is not a password protocol (the pre-shared secret must not be a weak password), EAP-AKA is not vulnerable to dictionary attacks.

6.5 Protection, Replay Protection and Confidentiality

AT_MAC and AT_COUNTER attributes are used to provide integrity, replay and confidentiality protection for EAP-AKA Requests and Responses [16]. Because keys are not available at the beginning of the EAP methods, the AT_MAC attribute cannot be used for protecting EAP/AKA-Identity messages. However, the AT_CHECKCODE [14] attribute can optionally be used to protect the integrity of the EAP/AKA-Identity roundtrip.

Confidentiality protection is applied only to a part of the protocol fields. The table of attributes (Table 5.2) summarizes which fields are confidentiality protected. On full authentication, replay protection of the EAP exchange is provided by RAND and AUTN values from the underlying UMTS AKA scheme. Protection against replays of EAP-AKA messages is also based on the fact that messages that can include AT_MAC can only be sent once with a certain EAP-AKA Subtype, and on the fact that a different K_aut key will be used for calculating AT_MAC in each full authentication exchange.

On fast re-authentication, a counter included in AT_COUNTER and a server random nonce is used to provide replay protection. The AT_COUNTER attribute is also included in EAP-AKA notifications, if they are used after successful authentication in order to provide replay protection between re-authentication exchanges.

The UE will only accept EAP-Success after successful authentication. Hence, the attacker cannot force the UE to believe successful authentication has occurred when mutual authentication failed or has not happened yet.

6.6 Negotiation Attacks

Because EAP-AKA does not protect the EAP method negotiation (e.g. EAP-SIM or EAP-AKA), EAP method downgrading attacks may be possible, especially if the user uses the same identity with EAP-AKA and other EAP methods.

EAP-AKA allows the protocol to be extended by defining new attribute types. When defining such attributes, it should be noted that any extra attributes included in EAP-Request/AKA-Identity or EAP-Response/AKA-Identity packets are not included in the MACs later on, and thus some other precautions must be taken to avoid modifications to them.

6.7 Protected Result Indications

EAP-AKA supports optional protected success/failure indications. If a failure occurs after successful authentication, then the EAP-AKA failure indication is integrity and replay protected.

Even if an EAP-Failure packet is lost when using EAP-AKA over an unreliable medium, the EAP-AKA failure indications will help to ensure that the UE and the AAA server will know the other parties of the authentication decision. If protected success indications are used, then the loss of Success packet will also be addressed by the acknowledged, integrity and replay protected EAP-AKA success indication. If the optional success indications are not used, then the UE may end up believing the server succeeded authentication when it actually failed. Since access will not be granted in this case protected result indications are not needed unless the client is not able to realize it does not have access for an extended period of time.

6.8 Man-in-the-middle Attacks

In order to avoid man-in-the-middle attacks and session hijacking, user data is integrity protected in EAP-AKA authentication. The EAP-AKA Master Session Key

or keys derived from it may be used as the integrity protection keys, or, if an external security mechanism is used, the link integrity protection keys may be derived by this external mechanism.

If EAP-AKA is used with a tunneling protocol or as part of a sequence of methods, there should be cryptographic binding provided between the protocols and EAP-AKA to prevent man-in-the-middle attacks through rogue authenticators being able to setup one-way authenticated tunnels. EAP-AKA Master Session Key may be used to provide the cryptographic binding.

7 SIMULATION OF AUTHENTICATION PROTOCOLS

In this section I will describe the realised simulations. These simulations will show the message exchanges of an initial authentication, of a fast re-authentication and also protection against some possible attacks.

7.1 About OMNeT++ Discrete Event Simulator

At the beginning of my work I had to acquire knowledge of OMNeT++ discrete event simulator and of the NED language, in which Zoltan Faigl helped me a lot.

OMNeT++ is an object-oriented modular discrete event simulator [17]. The name itself stands for Objective Modular Network Testbed in C++. OMNeT++ is a public-source, component-based, modular and open-architecture simulation environment. Its primary application area is the simulation of communication networks, but because of its generic and flexible architecture, it has been successfully used in other areas like the simulation of IT systems, queuing networks, hardware architectures and business processes as well.

The simulator can be used for:

- traffic modeling of telecommunication networks
- protocol modeling
- modeling queuing networks
- modeling multiprocessors and other distributed hardware systems
- validating hardware architectures
- evaluating performance aspects of complex software systems
- modeling any other system where the discrete event approach is suitable.

An OMNeT++ model consists of hierarchically nested modules. The depth of module nesting is not limited, which allows the user to reflect the logical structure of the actual system in the model structure. Modules communicate with message passing. Messages can contain arbitrarily complex data structures. Modules can send messages either directly to their destination or along a predefined path, through gates and connections.

Modules can have parameters which are used for three main purposes: to customize module behavior; to create flexible model topologies (where parameters can specify the number of modules, connection structure etc); and for module communication, as shared variables.

Modules at the lowest level of the module hierarchy are to be provided by the user, and they contain the algorithms in the model. During simulation execution, simple modules appear to run in parallel, since they are implemented as co-routines. To write simple modules, the user does not need to learn a new programming language, but he/she is assumed to have some knowledge of C++ programming.

OMNeT++ simulations can feature different user interfaces for different purposes: debugging, demonstration and batch execution. Advanced user interfaces make the inside of the model visible to the user, allow him/her to start/stop simulation execution and to intervene by changing variables/objects inside the model. This is very important in the development/debugging phase of the simulation project. User interfaces also facilitate demonstration of how a model works.

The simulator as well as user interfaces and tools are portable: they are known to work on Windows and on several Unix flavors, using various C++ compilers.

7.2 Simulation

First of all the interworking topology had to be designed in NED language. The NED language[18] supports modular description of a network. This means that a network description consists of a number of component descriptions (channels, simple/compound module types). The channels, simple modules and compound modules of one network description can be reused in another network description. As

a consequence, the NED language makes it possible for users to build their own module libraries.

Files containing network descriptions generally have a .ned suffix. NED files are not used directly: they are translated into C++ code by the NEDC compiler, then compiled by the C++ compiler and linked into the simulation executable.

My topology consists of the following modules: a user equipment (UE), a WLAN access network (WLANAN), an Authentication Authorization and Accounting server (AAA), a Home Subscriber Server (HSS), a Charging Gateway (CGW), an Online Charging System (OCS), a Packet Data Gateway (PDG) and a WLAN Access Gateway (WAG). This structure is shown in Figure 7.1.

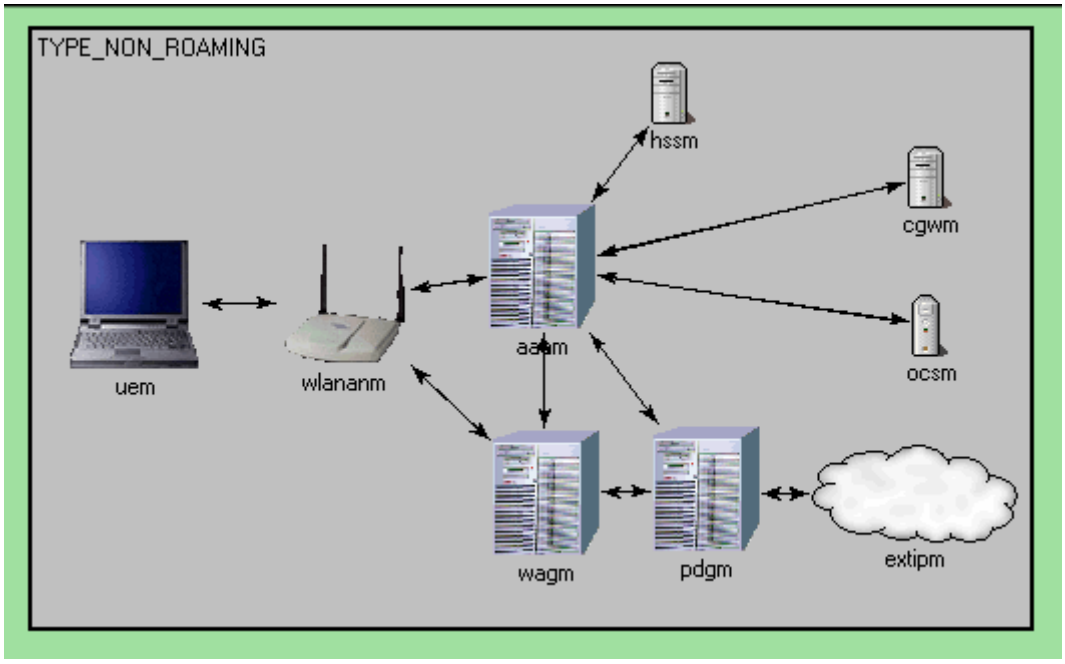

Figure 7.1. Simulation environment

Contrary to the authentication schemes above, in the simulation I showed messages between the WLAN AN and the UE as well.

In order to see the execution of different authentication methods I realized a full authentication and a fast re-authentication scheme. Afterwards the resistance to two external attacks can be seen.

7.2.1 Running the Simulation

The window arrangement that I used while running the simulations can be seen in Figure 7.2.

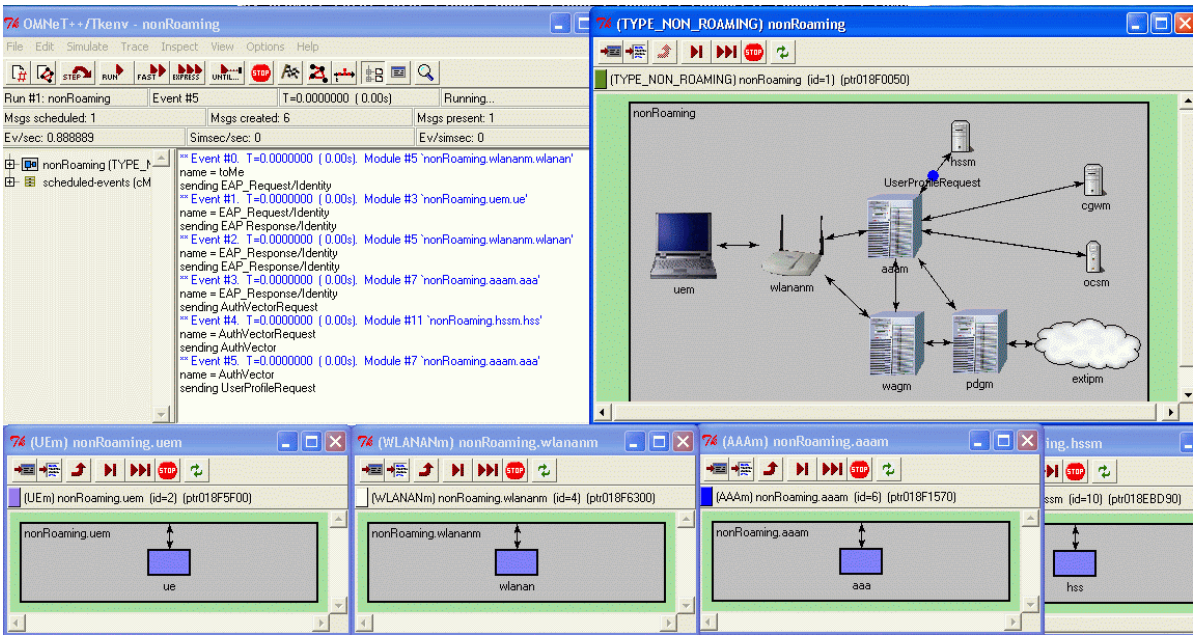

Figure 7.2. Simulation arrangement

The upper right case is used to display all the sent and received messages of each entity. In this case the results of an "if clause" can also be followed. For example when the AAA server receives a RES from the User Equipment there is a comparison with the SRES and the result of this can also be seen. This helped me to follow why an authentication process failed eventually.

In the below cases the entities can be seen that are involved in the authentication scheme. While running the simulation, the receipt of each message can be seen in these cases.

7.2.2 Full Authentication

Firstly I realized a full authentication scheme. This is an elementary part of the 3G-WLAN interworking, since each communication establishment begins with a full

authentication. In this simulation I realized the initial message exchanges between the User Equipment and the WLAN AN as well.

First the User Equipment sends an association request to the WLAN AN. The WLAN AN responds with an association response. The UE then sends an EAP start message which launches the EAP authentication between the UE and the WLAN AN. The WLAN AN sends an EAP Request to the UE and attaches its own identity. It sends an integrity protection value as well. On receipt, the UE checks the integrity value and if the result is positive (the value is greater then any of the precedent values), it responds with an EAP response and attaches its own identity. In my simulation the WLAN AN contains a RADIUS client so it can check the user's identity. If the user name is not present in the RADIUS client of the WLAN AN, it sends a request to the AAA server to get the missing username. In my simulation the usernames are entire numbers between 0 and 10, but this can be modified and real usernames can be introduced. If this part is successful, the WLAN AN continues the authentication method towards the AAA server. From that time the authentication procedure can be considered as an end-to-end authentication between the UE and the AAA server. For this reason I did not change the color of the message packets while passing through the WLAN AN. The AAA server receives the user identity in the NAI format. The reception of this message is shown in Figure 7.3.

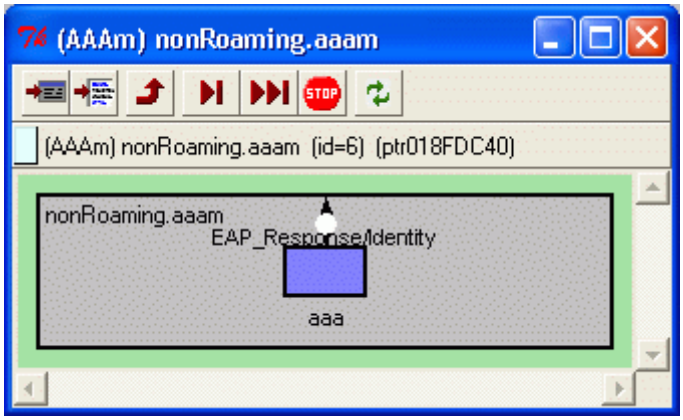

Figure 7.3. Receipt of user identity by the AAA server

On receipt, the AAA server checks if the identity is valid. The server also checks if it has an authentication vector available (RAND, AUTN, XRES, IK, CK) from previous authentication. In my simulated case it did not have any, so it retrieved

an authentication vector from the HSS. The AAA server also checks if the user's WLAN access profile is available. In my simulation it also had to be retrieved from the HSS. This is shown in Figure7.4 (blue point).

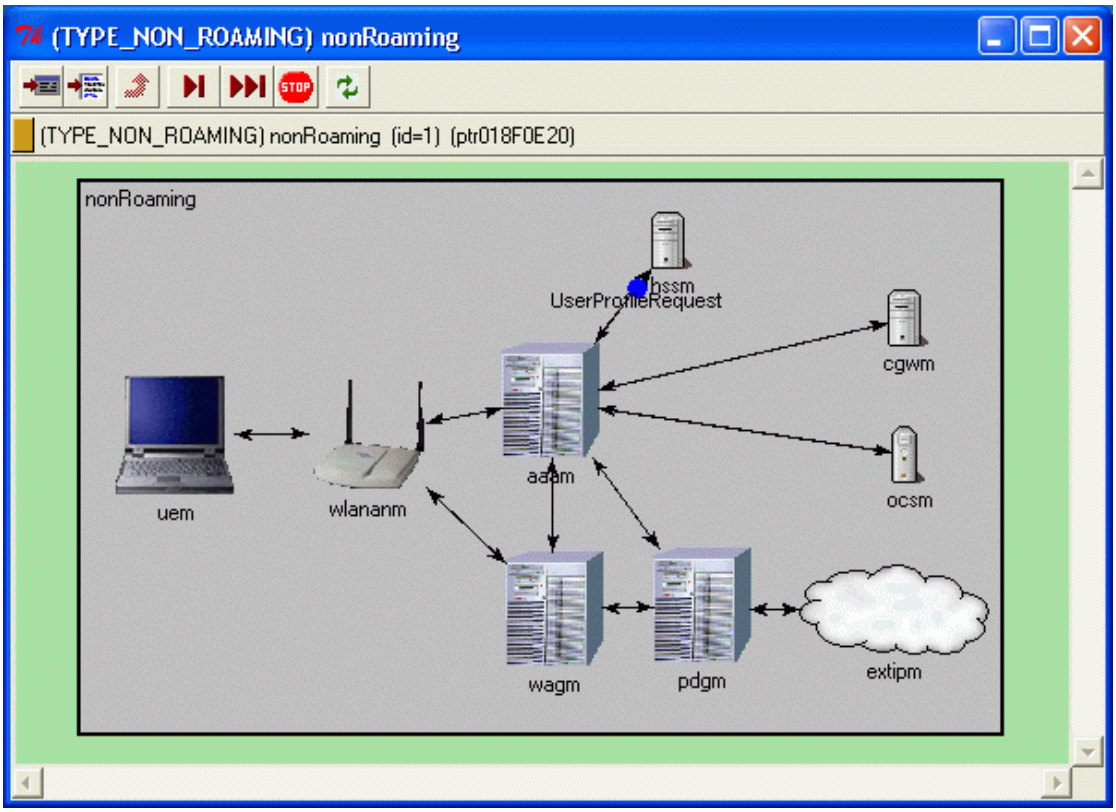

Figure 7.4. User profile retrieval from the HSS

Finally the AAA server derives new keying material from IK and CK. In my simulation this derivation was not realized, but gives the possibility to import real functions. Derived keying materials are used to encrypt and integrity protect the temporary identifier which is then transmitted to the UE. If the AAA server has all the necessary authentication material, it sends an AKA challenge to the User Equipment in form of an EAP-Request. This message package includes RAND, AUTN and the user identifier. The User Equipment runs the UMTS algorithm on its USIM card. It verifies if the AUTN originates indeed from the operator's AAA server. In my simulation I used a Sequence Number (SQN), the value of which was increasing permanently in the AAA server. This SQN is also present in the AUTN value, so the User Equipment can verify if this value is greater then any precedent SQN value. This

method prevents an eventual replay-attack. If AUTN is correct, the USIM computes RES, CK and IK values. Since I did not have the f2, f3 and f4 functions available I used fixed RES, IK and CK values. The perfection of this part is a further possibility.

Afterwards the UE sends EAP Response/AKA-Challenge to the AAA server. The AAA server compares the received RES value with XRES. In this simulated case the comparison was successful so the AAA server sends EAP Success message to the UE and the derived keying material to the WLAN AN. The WLAN AN will use this keying material to encrypt the communication with the authenticated User Equipment, but this encryption is out of scope of the authentication. The complete message flow with the comments can be seen in Figure 7.5.

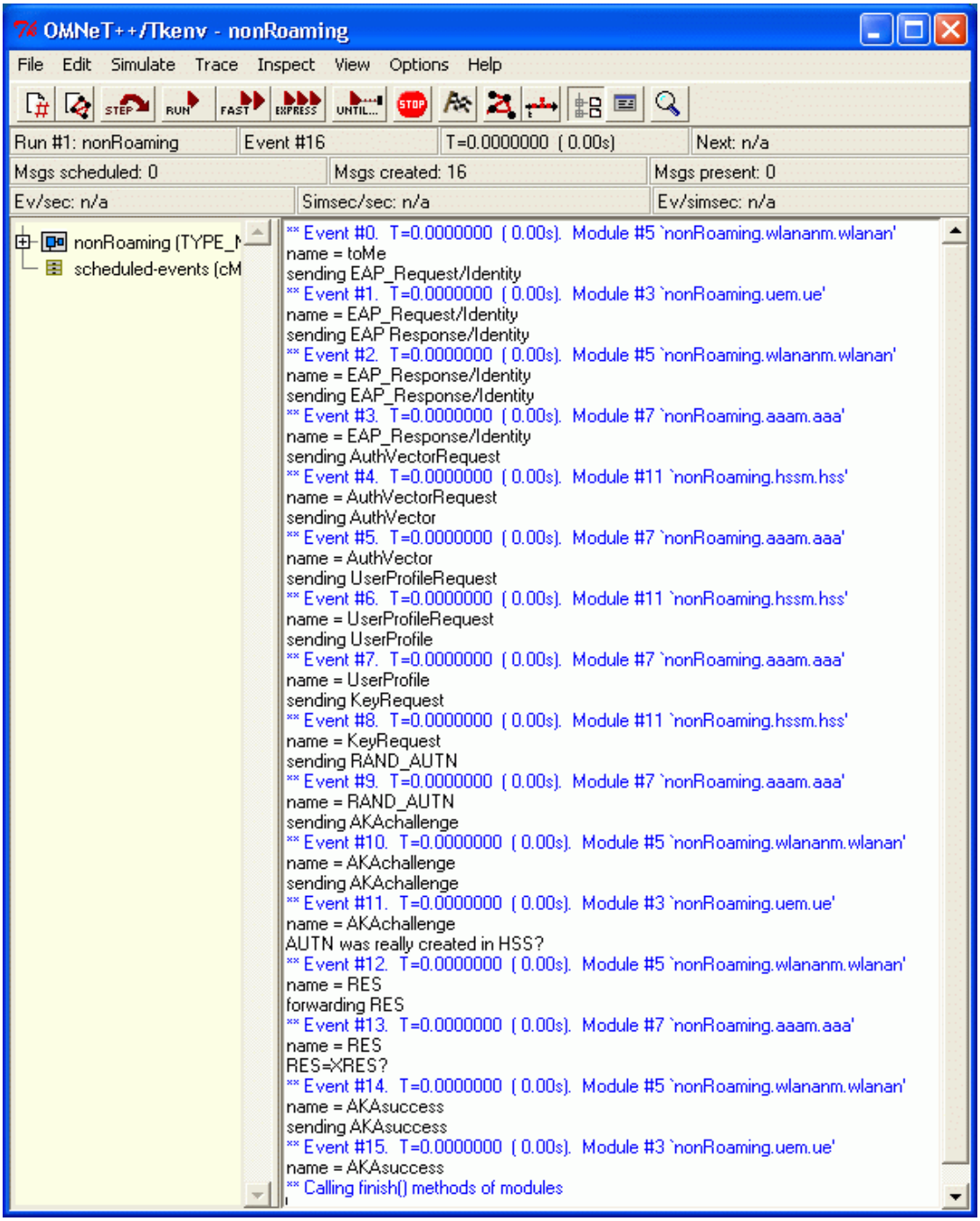

Figure 7.4. Displayed comments of a full authentication

7.2.3 Fast Re-authentication

As told in Section 5.5, the fast re-authentication is also an important process in order to avoid useless network operation between the AAA server and the

Authentication Center. I realized the fast re-authentication procedure as follows: when the AAA sends EAP request/AKA challenge to the UE, I used an AT_NEXT_REAUTH_ID attribute to communicate that the server supports fast re-authentication. On the EAP response packet the UE includes its fast re-authentication identity, instead of including the permanent identity. In my simulation I used a separate field in the identity to differentiate a permanent identity from a fast re-authentication identity. On receipt the AAA server checks this field and if the value is "1" it means that the UE accepted the fast re-authentication request and the used identity is a fast re-authentication identity. The AAA server checks if the received identity is present in its database. Since in my simulation there were only a few user identities in the AAA server, a one-by-one comparison was proceeded to find the received user name. However in case of many users, other methods should be applied. If the AAA sever does not have the received user identity in its database, it falls back on full re-authentication. In this case the AAA server continues with proceeding full authentication. (In my simulation the AAA sends a FastReauthFailed message to itself and after handling this message it falls back on full authentication.) If the AAA server recognizes the identity as a valid fast re-authentication identity, it proceeds the EAP-Request/Fast Re-authentication packet with the appropriate attributes to the UE. The realization of this packet was not difficult, since I only had to rename SQN to AT_COUNTER, RAND to NONCE_S and AUTN to AT_MAC. On receipt of these values the UE proceeds the same method as in a full authentication and sends back the result that is called AT_MAC. The AAA server verifies the AT_MAC value and sends EAP success or EAP failure message to the UE. An instance of the realized fast re-authentication is shown in Figure 7.5.

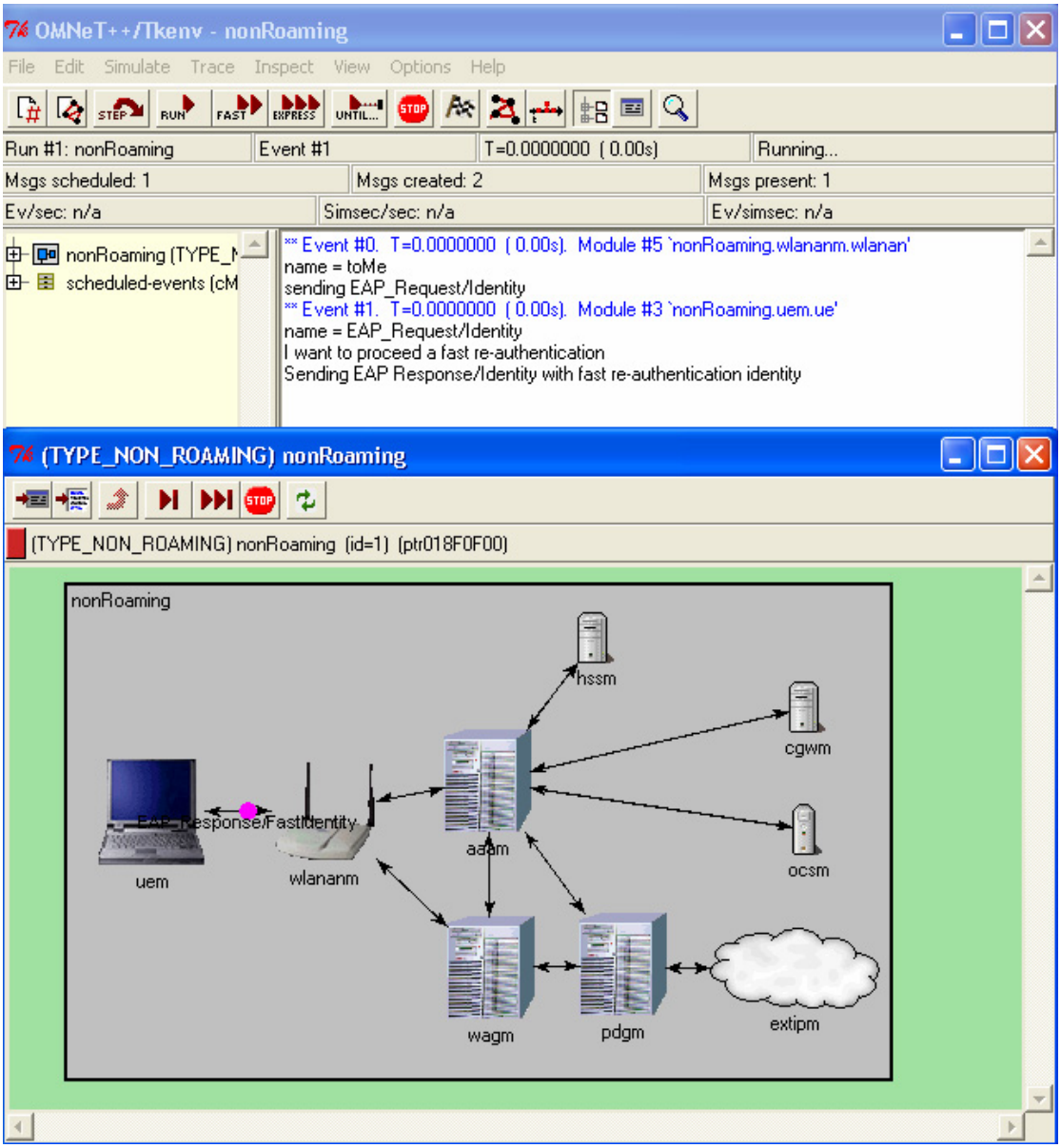

Figure 7.5. Sending user's fast re-authentication identity to the AAA server

7.2.4 Rogue AAA server

In my first attack scheme a rogue AAA server tries to impersonate the core network while proceeding a full authentication. In my example the rogue AAA server had recorded a full authentication between the UE and the real AAA server and tries to repeat the used values. It sends EAP Request/AKA challenge to the UE in order to have its permanent identity. On receipt of this message the UE verifies if the SQN part of the AUTN value is big enough. As soon as the UE realizes that the SQN value

is smaller or equal to the last full authentication SQN value it sends an error message to this false AAA server and the EAP message exchange terminates immediately.

7.2.5 Rogue User Equipment

In my second attack scheme a rogue UE tries to use a false identity in order to use illegally the WLAN network. I supposed that somehow he got to know the permanent identity of a legal user beforehand. On receipt of the EAP request/AKA challenge it tries to calculate the appropriate RES value, but since it does not have the correct K value, the sent RES will be incorrect. On receipt of the EAP response/AKA-challenge the AAA server compares the received RES with the XRES and terminates immediately the EAP exchange. An instance of this failed attack is illustrated in figure 7.6.

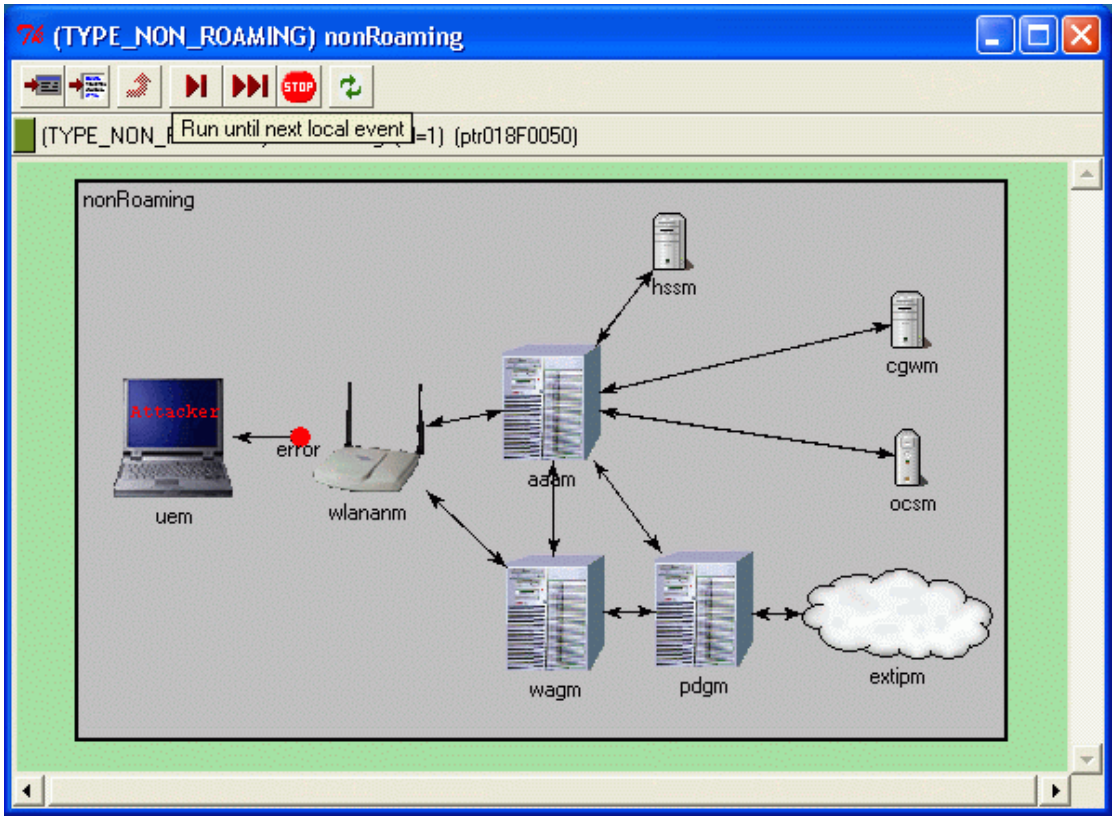

Figure 7.6. Failed attack of a rogue UE

7.2.6 Used Functions

In this section I give a list of the functions that I used several times in the simulation of 3G-WLAN interworking authentication. In order to show an example of each function I took out a part of the source code and gave a brief explication below each part.

```
#include <omnetpp.h>
```

This library has to be included in any OMNeT++ simulations on order to provide basic OMNeT functions

```
#include <string.h>
```

In the simulation I used the `strcmp()` function which is included in the `string.h` library.

```
//Parameter gyujto
const char * strt[]={
      "IMSI",    //0
      "TMSI",         //1
      "RAND",    //2
      "AUTN",
      "encrTMSI",//4
      "IK",
      "CK",       //6
      "XRES",
      "RES"       //8
};
```

Normally an OMNeT++ basic message can only contain characters and numbers. Since I wanted to transmit strings (like IMSI, TMSI, etc), I chose the following method: I made a string array with all the possible messages and during the simulation I only transmitted the index number of the given string. This solution also has the benefit of reducing the used memory size. In the *omnetpp.ini* these strings can be given concrete values that are used during the simulation in the following manner:
(This is a part of an omnetpp.ini file:)

```
 [Run 1]
*.res=3
```

This means that during “run 1” the RES value is 3.

```
res= par("RES");
```

With this command the value of RES can be achieved in the `handleMessage` function for example

```
      //Sajat uzenetformatum definialasa
      class sajatmsg : public cMessage{
      public:
            explicit sajatmsg(const char *name=NULL, int k=0, long
len=1, int pri=0, bool err=false) :
            cMessage(name,k,len,pri,err) {}
            int mezo1;
            int mezo2;
            int mezo3;
            int mezo4;
            int mezo5;
      };
```

In order to be able to put easily the above index numbers in the message field (see `mezoX`) I made an own message format for all message types. With this method, messages can be created as follows:

sajatmsg **msg*=new sajatmsg("*messageName*", colorCode);

msg->`mezo1` = messageIndex;

messageName

(Where `sajatmsg*` is my own message format)

```
      //UE
      class UE : public cSimpleModule {
          Module_Class_Members(UE,cSimpleModule,0)

            virtual void handleMessage(cMessage *msg);

      }; Define_Module( UE );
```

This is the usual way of declaring modules in OMNeT++. In this case the so called UE simple module is created. It contains only the `handleMessage` function. The handleMessage function contains the operations which describe the behavior of the module.

In OMNeT++ two different programming concepts exist. The first one is realized with activity functions. In this case all the processes are running simultaneously in the modules. In the second one an event-based model is used where an event is launched by received messages.

In the following the handleMessage function of the User Equipment (UE module) can be seen:

```
      void UE::handleMessage(cMessage *msg)
      {
            ev << "name = " << msg->name() << "\n";
```

When UE receives a message I used the ev function to make the message name appear on the display panel. The received message can be referred as *msg* and afterwards when we would like to process the message, this variable name has to be used. At this moment come the `if` clauses on order to handle different messages in an appropriate manner.

```
            if (!strcmp(msg->name(),"EAP_Request/Identity"))
            {
                  ev << "sending EAP Response/Identity\n";
                  sajatmsg *eapResponseId=new
sajatmsg("EAP_Response/Identity", 2);
                  eapResponseId->mezo1 = 0;
                  send(eapResponseId, "out");

            }
```

In the upper clause we can see the processing of an EAP Request/Identity message. It creates a message which is called EAP Response/Identity. It contains a field (mezo1) with zero value which means IMSI. (See strt [] array.)

```
            if (!strcmp(msg->name(),"AKAchallenge"))
            {
                  ev << "AUTN was really created in HSS?\n";
                  sajatmsg *RES=new sajatmsg("RES", 2);
                  RES->mezo1 = 8;
                  send(RES, "out");

            }
```

In this clause we can see the processing of an AKA challenge. It creates a message which is called RES. It contains a field (mezo1) with value 8, which means RES (See strt [] array.)

```


            delete msg;
      }
```

After handling the message (`msg`) it has to be deleted in order to free the allocated memory.

8 FURTHER POSSIBILITIES

The identification and authentication of users are essential in any kind of trusted transaction. The existing authentication and authorization methods of mobile networks have proven sufficiently secure and operable. It seems to be very appropriate to use the identity authentication and authorization services of GSM/UMTS networks not only for traditional voice and data services, but also in other services. In this section I will show examples of WLAN authorization methods, but it should be noted that these security solutions would be appropriate for any kind of service authorization.

Up to now the subscriber used the underlying UMTS AKA method to get access to the service that he/she wanted to use. In our case this service was the access to a WLAN network. The simplified scheme of this authorization method can be seen in Figure 8.1.

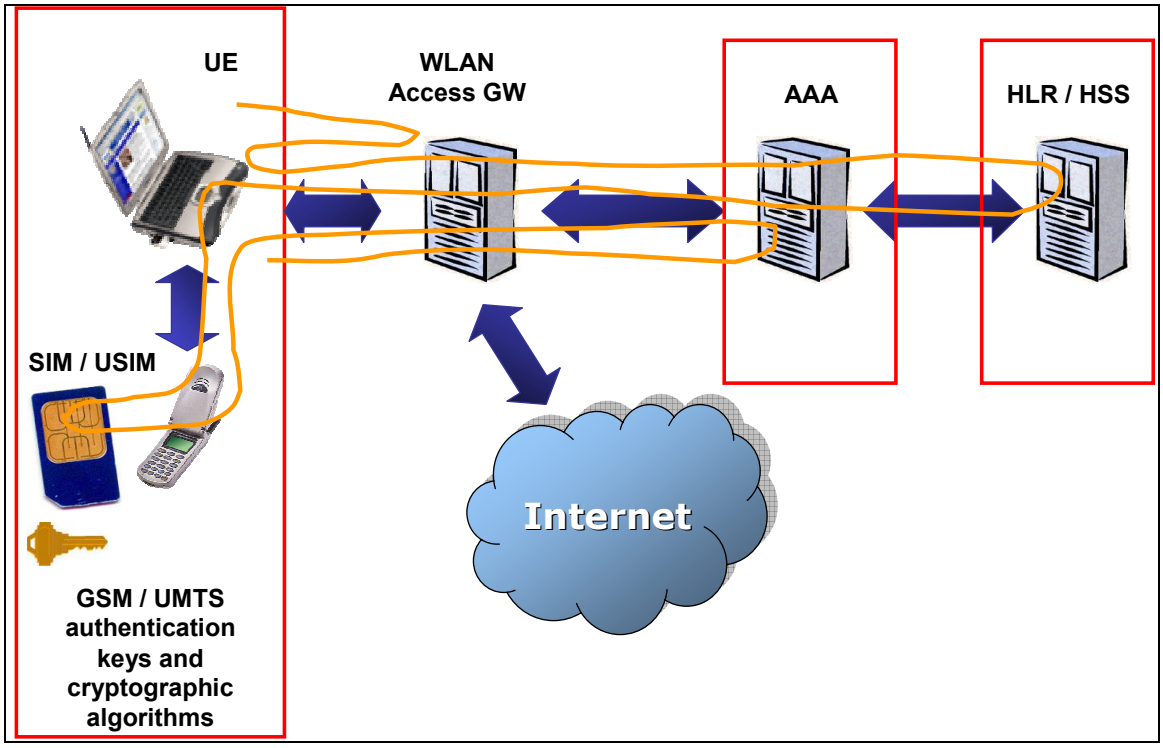

Figure 8.1. Simplified 3G-WLAN authorization scheme

The authorization process can be followed on the orange line. In this case the WLAN Access Network, the 3GPPP AAA server, the HLR/HSS and the (U)SIM card are involved in the authorization process. However other methods can be applied which are appropriate for other applications as well. In this section a list of these possibilities is presented.

8.1 One Time Password Distribution

In this scheme the authorization decision is not taken by the 3GPP AAA server, but with the help of a One Time Password (OTP). The authorization scheme can be seen in Figure 8.2. An aggregation point via which the mobile user can be accessed is also used which is called Acquirer in this scheme.

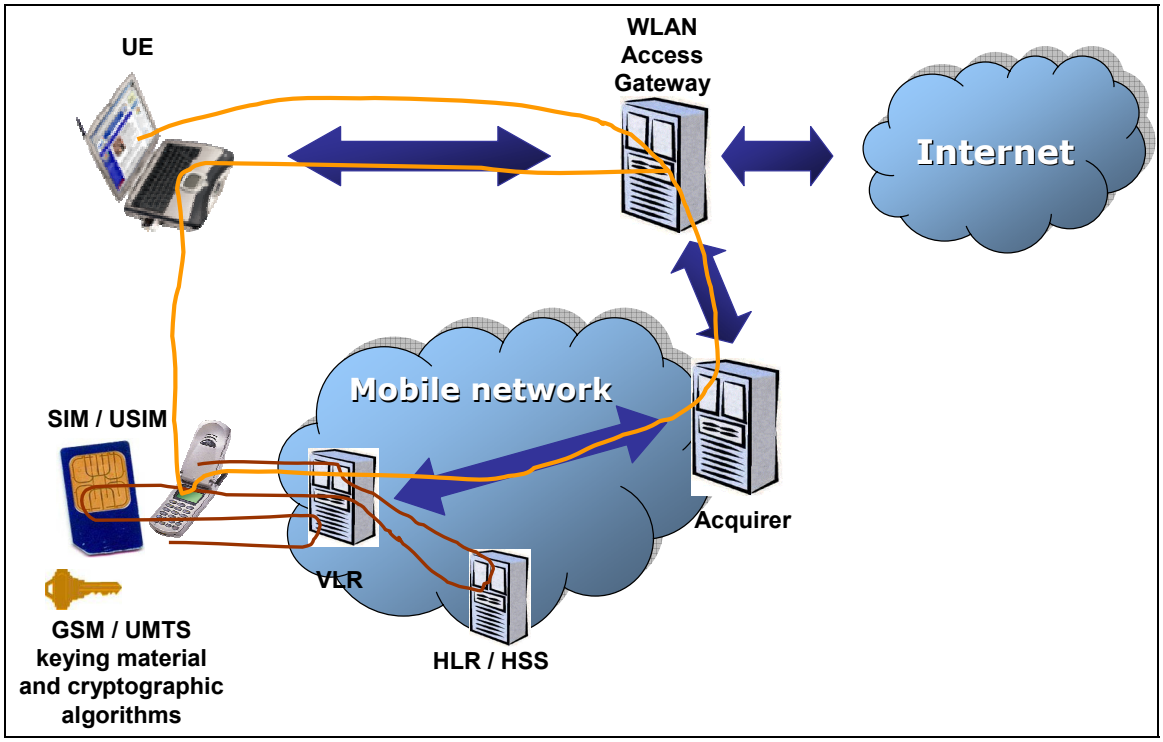

Figure 8.2. One Time Password

The red line represents the initial login of the 3GPP subscriber. This is independent from the WLAN interworking and can be considered trustworthy. Afterwards the access to WLAN service processes as follows:

1. The User Equipment sends an access request to the WLAN Access Gateway (WAG) and attaches the mobile phone number of the user
2. The WAG creates a OTP which will be used while logging in the WLAN network
3. The OTP is forwarded to the Acquirer which sends the received password to the appropriate VLR
4. The VLR sends the OTP to the user's mobile phone in a short message
5. The user types the received OTP in his/her User Equipment, that logs in the WLAN access network

The advantage of this solution is that the mobile operators do not have to create an AAA server and also that the user's mobile phone is needed in the network access, so frauds can be effectively prevented.

8.2 Authorization with User Side Signature

In this solution we take advantage of the fact that the mobile (U)SIM card is able to effectuate trustworthy signature. In this scheme a Mobile Signature Service Provider is used which is able to send messages to the USIM card in an appropriate format and to verify the validity of a signed message received from the USIM card [20]. The authorization scheme can be seen in Figure 8.3.

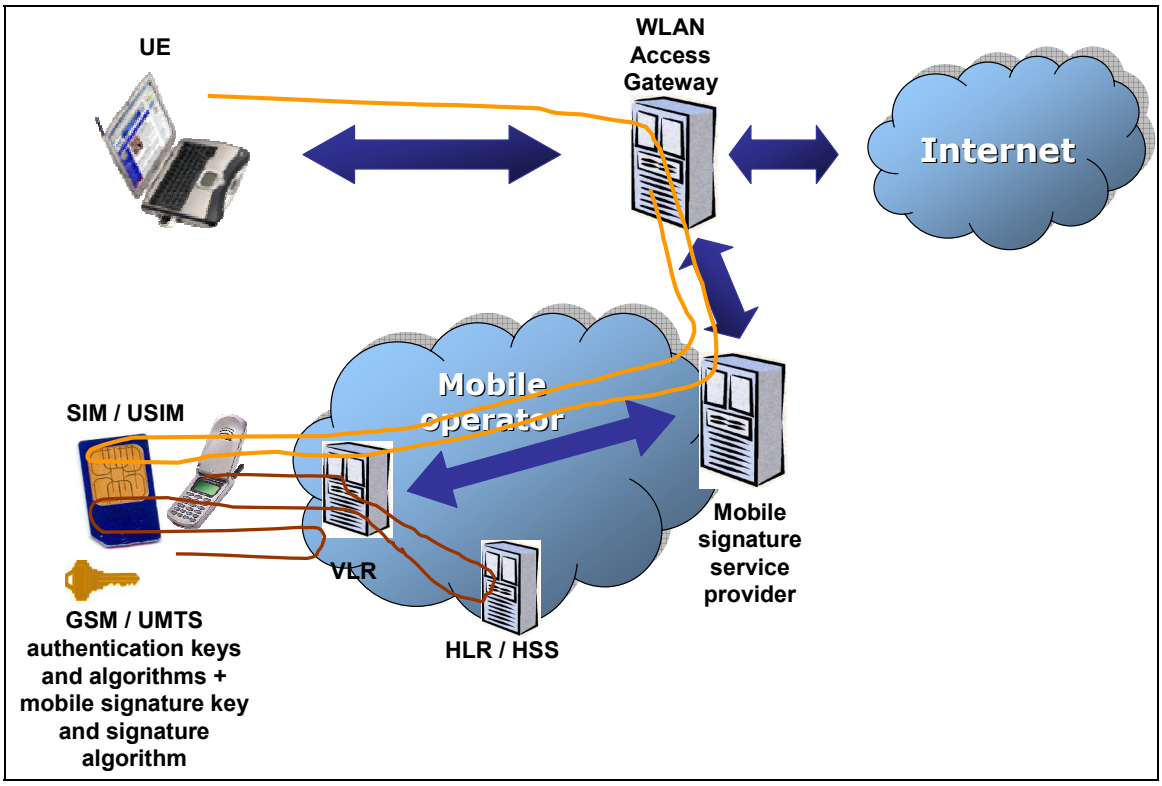

Figure 8.3. User authorization with user side signature

After successful mobile subscriber login, the WLAN service authorization is effectuated as follows:

1. The User Equipment sends an access request/mobile phone number to the WLAN Access Gateway
2. The WAG sends a question "Do you really want to access this service" to the Mobile Signature Service Provider
3. This question is forwarded to the user
4. If the user really wants to access the service, he/she allows the USIM card to create the signature, and then sends back to the Signature Service Provider.
5. The mobile Signature Service Provider checks the validity and the contents of the response and informs the WAG of the decision
6. If authorization was successful, the WAG allows the UE to access the service

The disadvantage of this solution is that the SIM card has to be modified, but in exchange a much more secure authorization is provided.

8.3 Authorization with Server Side Signature

In the previous solution we used the USIM card to sign the access approval. However the ability of providing signature is an advanced USIM feature and most of the subscribers do not have such possibilities. Hence a signing server could also be able to effectuate a trustworthy signature. The realization of this solution can be seen on Figure 8.4.

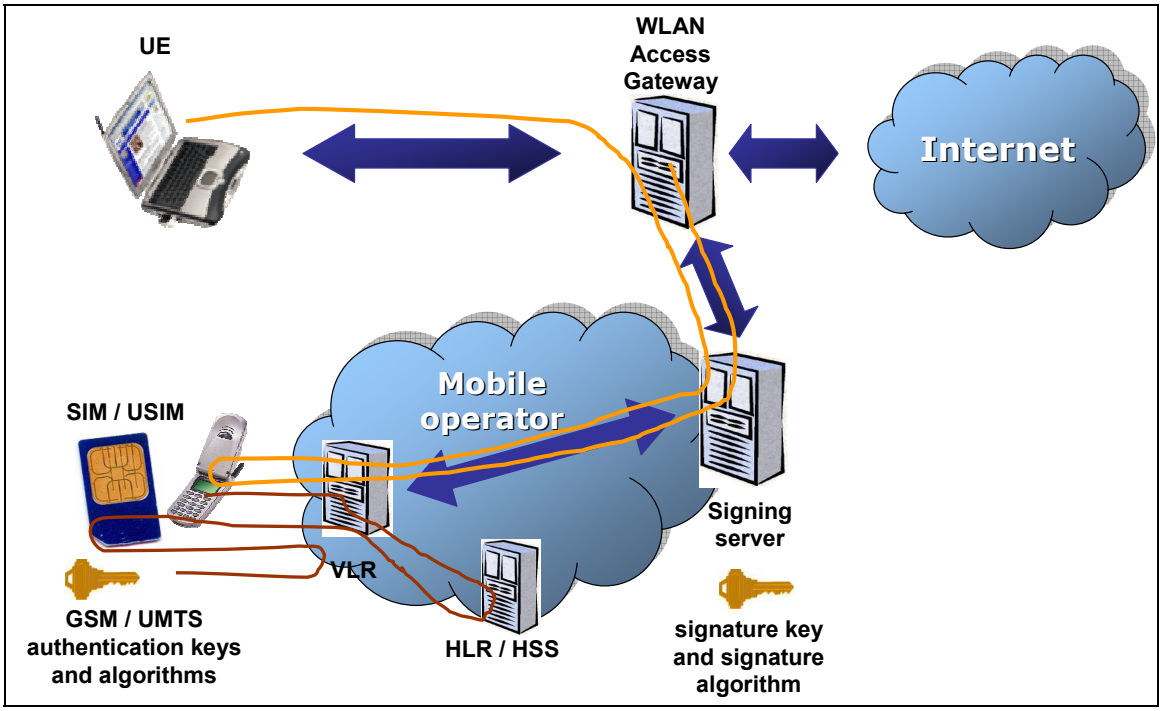

Figure 8.4. Authorization with sever side signature

1. The UE sends an access request/mobile phone number to the WAG
2. The WAG sends a question “Do you really want to access this service” to the Signing Server
3. The Signing Sever forwards the question to the mobile phone of the user
4. The user sends back an access approval
5. The Signing Server signs the response and forwards signed access request to the WAG
6. In case of valid signature the WAG gives service access authorization to the UE

As we can see, in this case the WAG has to contain the certificate of all the subscribers in order to decide whether the signed response is valid or not. This solution can be implemented much faster, since the SIM card does not need to be modified.

8.4 Identity Roaming

This is a global solution of how any service access can be authorized. As the precedent solutions this scheme also gives the possibility to be re-used for other applications that need authorization, not only for WLAN access.

The Identity Roaming means that any attribute of an identity can be transferred regardless to the physical position of the user and to the actual service provider type [21] [22].

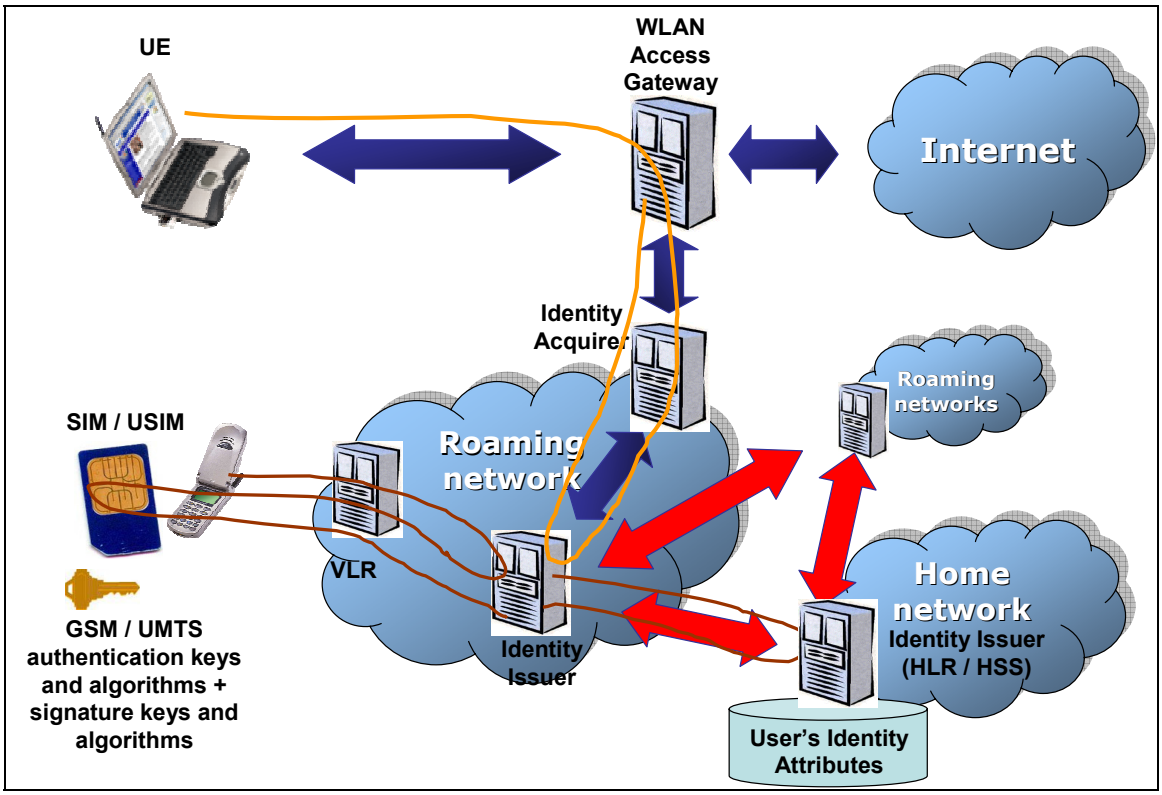

Figure 8.5. Identity roaming

In this scheme a so called Identity Issuer is used. The Identity Issuer can be located in the operator's HLR/HSS and its function is to provide the Identity Acquirer with the user profile attributes. In Figure 8.5 I showed a case where the Identity Issuer had to acquire the subscriber's user attributes from user's home Identity Issuer. The authorization processes as follows:

1. The user is located abroad and would like to access a local WLAN network. He/she sends an access request to the WAG
2. The WAG wants to retrieve the credential of the user, so forwards the access request to the Identity Acquirer
3. The Identity Acquirer sends the request to the Identity Issuer of the actually used mobile operator
4. Since in this case the Identity Issuer does not dispose the necessary user attributes, it sends a request to the user's home Identity Issuer in order to acquire the attributes
5. When the Identity Issuer of the visited network has the necessary attributes, it sends them to the Identity Acquirer
6. The Identity Acquirer forwards the needed user attributes to the WAG.
7. Based on this credential the WAG accepts or refuses the access.

It should be noted that the user profile does not only contain WLAN authorization attribute, but any other attribute that can be used while proceeding a service access request. This is a global solution the study of which is a very actual topic and could be the subject of another thesis.

9 CONCLUSION

Because of the increasing amount of mobile applications, security enhancements become crucial while designing a mobile system. The identification and authentication of users are essential in any kind of trusted transaction. The existing mobile AKA methods of mobile networks have proven sufficiently secure and operable. It seems to be very appropriate to use this authentication procedure to other services.

In this work I gave an overview of the applied security solutions in existing mobile systems and afterward I described the elaborated parts of 3G-WLAN interworking security and gave propositions to the manner how EAP-AKA security could be applied. Since at the present time this work is under elaboration at ETSI, modifications in the standard are possible. Then I observed the proposed interworking solution from a security point of view. Afterwards the simulation of these authorization message exchanges was presented in OMNeT++. The program that I realized can be a convenient utility to simulate the authentication and authorization message exchanges of any kind of service access. A further work would be to make my simulation program be able to apply real f1, f2, f3, f4, f5 functions while computing authentication keys. Finally I listed some other possible solutions to realize any kind of service access authorization.

Naturally the exact 3G-WLAN architecture will depend on the operator that deploys the system, but in order to assure a high compatibility level, it is recommended to make the standard as strict as possible.

10 ACKNOWLEDGEMENTS

It was a honor to me to work with Zoltán Faigl, with Maté Szalay and with Győző Gódor (MCCL laboratory). I would also like to thank for the useful contribution of Balázs Bertényi (Senior System Specification Engineer at NOKIA).

11 ABBREVIATIONS

3GPP	Third Generation Partnership Project
AAA	Authentication, Authorization and Accounting
AKA	Authentication and Key Agreement
AP	Access Point
APN	Access Point Name
AS	Authentication Server
AuC	Authentication Centre
AUTN	Authentication Token
AUTS	Authentication Synchronisation
BTS	Base Transceiver Station
CGW	Charging Gateway
CK	Cipher Key
DES	Data Encryption Standard
EAP	Extensible Authentication Protocol
EAPOW	Extensible Authentication Protocol Over WLAN
EMSK	Extended Master Session Key
GSM	Global System for Mobile
HLR	Home Location Register
HPLMN	Home Visited Public Land Mobile Network
HSS	Home Subscriber Server
IETF	Internet Engineering Task Force
IK	Integrity Key
LAN	Local Area Network
MAC	Message Authentication Code OR Medium Access Control
MCC	Mobile Country Code
MK	Master Key
MNC	Mobile Network Code
MSIN	Mobile Subscriber Identification Number

MSK	Master Session Key
NAI	Network Access Identifier
NAI	Network Access Identifier
NIC	Network Interface Card
O&M	Operations and Maintenance
OCS	Online Charging System
PDA	Personal Digital Assistant
PDG	Packet Data Gateway
PLMN	Public Land Mobile Network
PPP	Point to point Protocol
PRF	Pseudo-Random number Function
QoS	Quality of Sevice
RADIUS	Remote Authentication Dial In User Service
RAND	Random Number
SIM	Subscriber identity Module
SLIP	Serial Line Internet Protocol
SRES	Signed Result
TEK	Transient EAP Key
TMSI	Temporary Mobile Subscriber Identities
UE	User Equipment
UICC	Universal Integrated Circuit Card
UICC	Universal Integrated Circuit Card
UMTS	Universal Mobile Telecommunication System
USIM	Universal Subscriber Identity Module
VPLMN	Visited Public Land Mobile Network
WAG	WLAN Access Gateway
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network
WLAN AN	WLAN Access Network
XRES	Expected User Response

12 REFERENCES

[1]	GSM World from the GSM association http://www.gsmworld.com/index.shtml
[2]	Useca: UMTS Security Architecture, Heckmanns, 1999.
[3]	David Kahn: The Codebreakers ISBN 0684831309
[4]	Whitfield Diffile and Martin E. Hellman: New Directions in Cryptography, IEEE Transactions on Information Theory, IT-22, November 1976, pp 644-654
[5]	R.L. Rivest, A. Shamir, L. Adleman: Public Key Cryptography, CACM 21, 120-126, 1978
[6]	ElGamal T.: A Public Key Cryptosystem and Signature Scheme Based on Discrete Logarithms, IEEE Transactions on Information Theory, vol IT-31 no 4, pp. 469-472, 1985
[7]	Lisa Phifer: 802.1X Port Access Control for WLANs http://www.wi-fiplanet.com/tutorials/article.php/3073201
[8]	Heikki Kaaranen at al.: UMTS Networks – Architecture mobility and Services, Wiley 2001
[9]	3GPP TS 23.234; 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP System to Wireless Local Area Network Interworking
[10]	3GPP TS 33.234 (2004-03) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP System to Wireless Local Area Network Interworking
[11]	Industrial Technology Research Institute: RADIUS and Diameter http://www.csie.nctu.edu.tw/~sltsao/RADIUS-DIAMETER.pdf
[12]	3GPP TS 23.234 V6.0.0 (2004-03) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP System to Wireless Local Area Network Interworking
[13]	3GPP TR 22.934 V6.2.0 (2003-09) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP System to Wireless Local Area Network Interworking

[14]	L. Blunk, J. Vollbrecht (Merit Network Inc): Extensible Authentication Protocol http://ietf.levkowetz.com/drafts/eap/rfc2284bis/draft-ietf-eap-rfc2284bis-08.o-from-08.n.diff.html
[15]	H. Haverinen, J. Salowey: Extensible Authentication Protocol Method for GSM Subscriber Identity Modules (EAP-SIM)
[16]	J. Arkko (Ericsson), H. Haverinen (Nokia): Extensible Authentication Protocol Method for UMTS Authentication and Key Agreement (EAP-AKA)
[17]	OMNeT++ Descrete Event Simulator System Community Site http://www.omnetpp.org/external/doc/html/usman.php
[18]	OMNeT++ Descrete Event Simulator System Community Site – NED description http://www.omnetpp.org/external/doc/html/usman.php#sec127
[19]	Rice Compoter Science http://www.cs.rice.edu/~astubble/wep/wep_attack.html
[20]	ETSI TR 102 203: “Mobile Commerce (M-COMM); Mobile Signatures; Business and Functional Requirements”
[21]	Radicchio: “A universally recognized and accepted identity scheme that leverages the mobile infrastructure.” WORKING DOCUMENT Best Practice Working Group 26th September 2002