Supporting Secure Scalable End-To-End QoS In 4G Mobile Wireless Networks

Odhiambo Marcel O, Muchenje Best
ohangmo@unisa.ac.za, 46730133@mylife.unisa.ac.za
Department of Electrical and Mining Engineering, University of South Africa (UNISA),
P.O. Box 392, UNISA - 0003, South Africa.

Abstract: *With the convergence of the Internet and wireless communications, mobile wireless networks and data services are undergoing tremendous evolutionary growth that has seen the development of fourth generation (4G) mobile wireless access technologies based on an all-IP platform. However, major challenges in the development of such heterogeneous network infrastructure such as quality of service (QoS) provisioning and network security services for mobile users' communication flows, among others still exists. In this paper an integrated architectural view and methodology for QoS and security support in 4G mobile wireless networks, which integrates QoS signaling with secure enhanced evolved packet system authentication and key agreement (SE-EPS AKA protocol) is presented. The success of 4G mobile wireless networks depends on the prudent deployment of homogeneously designed, high-speed, secure, multiservice IP-centric integrated multimedia, voice and data networks.*

Keywords: Terms- 4G mobile wireless networks, Security enhanced Evolved Packet System Authentication and Key Agreement (SE-EPS AKA), quality of service (QoS)

1. Introduction

The introduction of fourth generation (4G) mobile wireless networking has brought about a number of interesting but also scaring challenges, chief among them is the integration of quality of service (QoS) and network security in an environment now heavily proliferated with computing devices with diverse computing capabilities, which poses a great risk when in the wrong hands. This is further compounded by business models pursued by different telecom services. The use of IPv6 protocol as a convergence layer has immensely eased the support of seamless mobility and QoS across heterogeneous networking environments, provision of content-rich multimedia and value-added services in such multi-provider heterogeneous network environments demands a common signalling framework for session negotiation, network resources reservation, session and QoS negotiation, and most importantly, integrating QoS and network security services in the signalling framework.

This paper focuses on developing a seamless integration of QoS and network security services for heterogeneous 4G mobile wireless networks. And as such the QoS sub-system conceptualised here is based on the use of QoS brokers (the mobility management entities, MME) that manage network resources and performance admission control for user equipment (UE) data flows. The proposed architecture give rise to three scenarios for session setup and (re)negotiation, differing on the entity that issues requests to QoS broker, namely (a) user equipment (EU) itself, (b) services proxies within the framework and (c) modules in the network access routers, that are able to do application signalling parsing and modification.

2. Overview of the QoS Services Architecture

Figure 1 is an outline of a 4G mobile wireless network. It illustrates the architecture of the evolved packet core (EPC). The radio-access network (RAN) and the evolved packet core (EPC) are also referred as the evolved packet system (EPS). Detailed explanation for the functionalities of the various entities of this network architecture is given in the literature [1].

The main design aim of 4G mobile wireless networks is the support of seamless UE mobility under a unified heterogeneous architecture that accommodates scalable and incremental development of new advanced applications and services. Thus, the IPv6 protocol, used as the convergence layer of 4G systems, is used natively to support mobility. The IPv6 creates an abstraction layer that conceals technology-specific application environments. Extension enhancements added to IPv6 in 4G mobile environments completely provides seamless mobility with fast handoffs. The correlation between mobility and QoS is outlined in [2].

In the proposed 4G network model several network domains, each with a host of access networks supporting disparate wireless technologies, are interconnected to each other via a core network, thus allowing different network operators to internetwork in a common environment. Special arrangements amongst operators have to be in place to allow integration of services and applications across different network domains. Figure 2 illustrates the proposed network architecture.

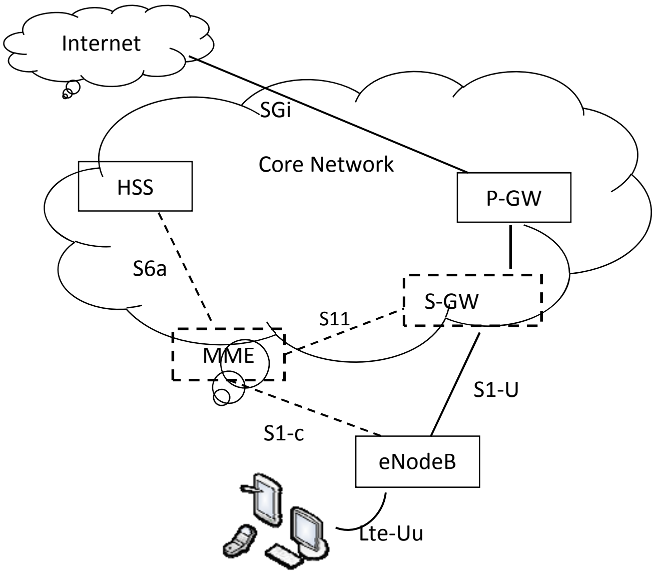

Figure 1: Core Network Architecture [1]

The MMEs in the access network perform admission control for data flows and inter- and intra-domain handoffs, and manage network resources, configuring the access routers, in policy decision or enforcement relationships. In addition, the MMEs help optimise network resources by performing load balancing for the users and sessions among the available networks through the use of network initiated handoffs. QoS support in the network core is based on DiffServ for scalability reasons, thus enabling aggregated inter-domain network segments.

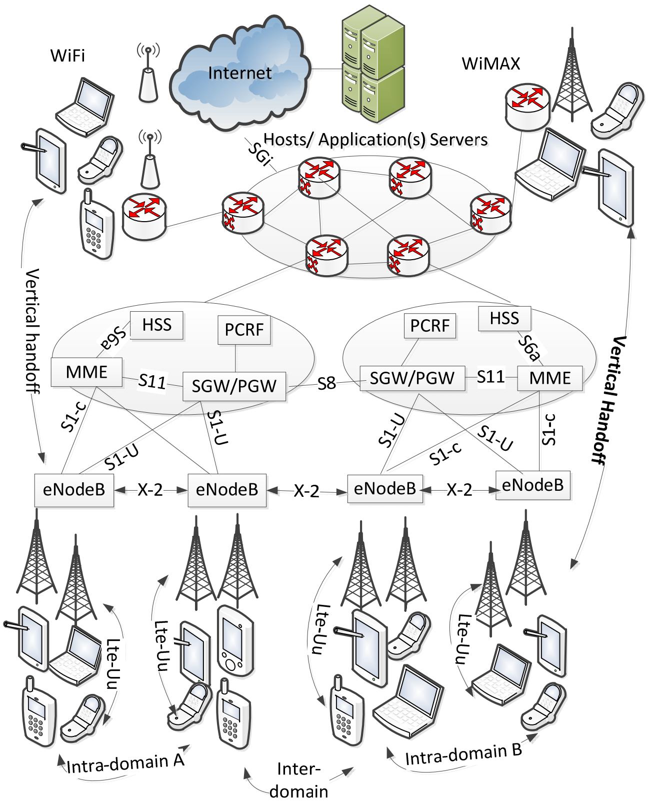

Figure 2: Network architecture [2]

The aggregated information is propagated to the access network MME where it is used for admission control in order to achieve end-to-end QoS for data flow. The integration of InterServ and DiffServ allows per-flow and per-aggregate processing of data in two-layer hierarchy architecture of which the end result is providing fine-grained QoS control while keeping the scalability properties of per-aggregated core resource management decoupled from per-session signalling. The service provision platform (SPP), within the network core, enable the running of the services and applications, through the multimedia service platform (MMSP), which consists of a broker, and proxy servers responsible for the provision and control of multimedia services and is also capable of mapping application level QoS configurations to network resource requirements and performing QoS requests for data flows. This architecture has a large degree of flexibility in QoS signalling, enabling the use of diversity of QoS access signalling scenarios that will fulfil the needs for different applications and different business cases for a diverse range of network operators and access services providers.

Unification of the scenarios is achieved by centralisation of the admission and handoff control at the access network MMEs. The SPP contains a core network MME which is responsible for resource management in the network core. Policies for resource management are defined by the policy-based network management system (PBNMS) and are sent to the core network MME where they are cached in a local database for use. The central monitoring system (CMS) collects statistics and other network usage data from network monitoring entities and feeds the PBNMS and MMEs with this information for proper network resources management [2].

During the user registration on the network the access MME retrieves a subset of the user's profile from the Authentication, Authorisation, Accounting, Auditing and Charging (AAAAC) system, which is part of the HSS and PCRF, in the EPC. This is meant to improve efficiency and scalability. This subset, called the network view of the user profile, contains information on the set of network services such classes of services, bandwidth parameters etc. as outlined in the service level agreement specifications for each user. Similarly, a service view of the user profile, containing information on the higher level services available to the user such as voice calls, video telephony, and the respective codecs, that is retrieved by the MMSP to control multimedia services [2].

3. QoS Signalling Scenarios Testing and Validation

This section outlines different QoS signalling scenarios during multimedia call initiation between two UEs. According to this proposal the MMSP, ARM and the UE are able to issue QoS requests (several signalling protocols such as SIP can be used). Figure 3 illustrates a simplified scenario in which UE1 initiates a multimedia session with another terminal UE2, where the two UEs are in different network domains.

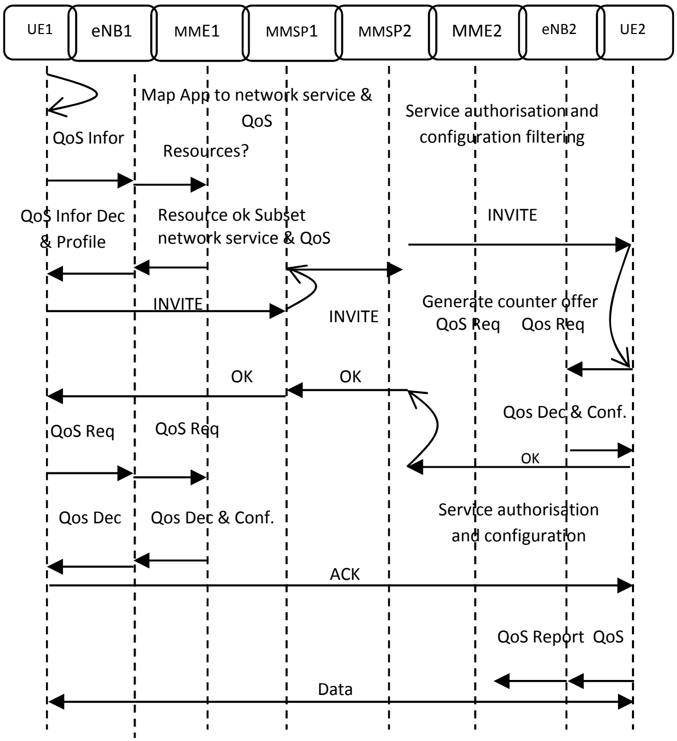

Figure 3: QoS session initiation, UE scenario [2]

The user equipment UE1, with the help of its resident QoS client, maps application needs to the networks and QoS requirements and sends requests to its serving access network QoS broker MME1 via a QoS attendant in the access router eNB1. The QoS signalling between the QoS client and the attendant is implemented as an extension to resource reservation protocol (RSVP). The MME1 respond with information on the available resources according to user profile and network status. If allowed by the MME1, the UE1 sends an INVITE message indicating the initial QoS parameters to UE2. Upon receiving the INVITE message MMSP1 performs service authorisation (with the help of the SEEP aka protocol), filtering out services and applications not allowed in the service level agreement specification outlined for the user UE1. Once authorised, the INVITE is forwarded to UE2. UE2 matches the QoS parameters in the INVITE messages to its own, requests MME2 for available network resources, and generates a counter-offer. Upon receiving this message the MMSP2 filters the services to those authorised. When the response message arrives at UE1, it selects the service to use, informs MME1 to configure the access router accordingly with the required bandwidth and queues available space for the data flows and service classes, and sends the acknowledge message (ACK) containing the final configurations that will be used. This message triggers the sending of QoS reports to MME2 confirming the QoS configuration parameters in the access routers. Applications that make use of out-of-band signalling (signalling that make use of some form of a separate dedicated communication channel) may also be made QoS aware by coding them to invoke this procedure [3].

The second scenario involves the MMSP. The UEs do not perform QoS requests: in this instance they perform some form of SIP (session initiation protocol (SIP) used in signaling communications protocol for controlling multimedia sessions such as voice and video calls over the Internet Protocol networks) signalling through the use of extended proxy servers which are capable of parsing QoS configuration parameters, mapping them to networks resource requirements and contacting the MMEs to perform QoS requests. These proxies also enforce policies set out in the service level agreements configured by the operators as per user service needs and requirements (as reflected by the respective service view of user profile) [2].

4. Security Services Protocol Verification

The following section describes a computational framework for proving the SEEP AKA using a cryptographic verification tool, the CryptoVerif tool. The specification of the CryptoVerif tool is translated into OCaml [5] to produce the implementation of the SEEP AKA protocol. OCaml is a functional language, which also facilitates the compilation because the CryptoVerif specification uses oracles that can be immediately translated into functions [3].

Proving the security protocol alone is not sufficient. The specification of the protocol may be correct, but the implementation can carry some errors as explained in [4]. There are several ways of obtaining a secure implementation of a security protocol, one of which is writing the specification first, proving it correct, and then generating and implementing it. Thus, according to [4] the general belief is to start by designing the protocol, formalise it, prove it secure formally and then finally implementing it. This is the methodology adopted in this research paper.

In order to generate the SEEP AKA implementation a compiler that takes a CryptoVerif specification and returns an implementation in OCaml is pursued [3].

Figure 4 illustrates an overview of the approach used to obtain a proved implementation of a cryptographic protocol. Two distinct steps are observed. First, a written specification of the CryptoVerif protocol is obtained. This specification contains a list of security assumptions on the cryptographic primitives. This specification guarantees the desired security properties, for example, the secrecy, authentication, authorisation, auditing etc., in the computational model by using the CryptoVerif tool [4].

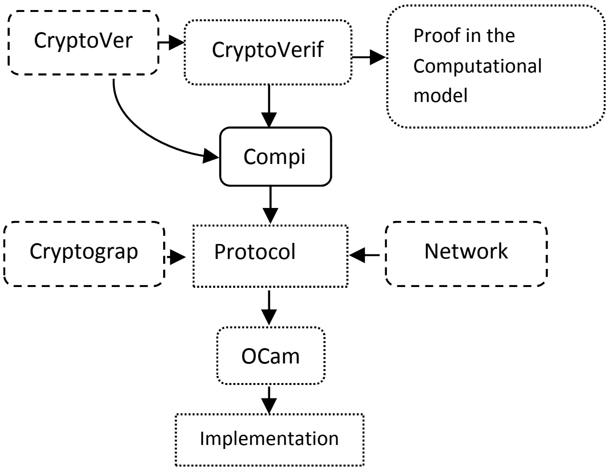

Figure 4: Overview of the approach [5]

Second, the compiler transforms the specification into protocol code. To build the implementation, the following codes are generated:

(i) The code corresponding to the exchange of messages across the network, which uses the results given by the functions in the protocol code. This code can be considered as a part of the adversary, and so it is not required to prove this part of the code.

(ii) The code corresponding to the cryptographic primitives. This is used by the protocol code, and thus must be proved manually that the primitives satisfy the security assumptions made in the specification file [5].

Then, the OCaml compiler is used on these codes to implement the protocol, from which a single protocol specification is obtained as both proof that the protocol is secure in the computational model and in executable implementation of the protocol.

The protocol implementation derived is used to formulate a security framework. The solution so proposed, on the user's side, comprises of the user equipment (UE), whose design is based on some form of trusted mobile platform (TMP) [6], and a biometric reader (BR) as shown in Figure 5. The access network, the home network environment, the service provider and the user equipment manufacturers host some form of public certificate issued by their own trusted authority which in turn should have connectivity to 4G mobile networks to enhance secure flow of information between the user equipment and the various 4G network entities.

Nomenclatures used for the proposed security scheme are donated as follows:

ID_X - X's identity
SK_X - private key
PK_X - public key
$Cert_X$ - digital certificate
Sig_X - digital signature
H(x) - a secure hash function and E(k, x) represents encrypting content x with key k.

The authentication schemes begins by having a user password (PW) and a universal subscriber identification module (USIM) that is capable of checking the integrity and validity of the mobile platform, and also have the capability of storing authentication parameters that includes the user's biometric template (F_U, usually eye iris, facial identity or fingerprints), SK_{User}, $Cert_{User}$, $Cert_{HE}$, x, y and z. The authenticating parameters x, y and z are computed by user's HE as follows before the home environment (HE) issues the USIM card to the user. n is a secure module of RSA signature algorithm given as.

$x=H(F_U||PW)$, $y=x\oplus H(PW)$, $z=S\oplus H(F_U\oplus PW)$, and then $S = H(ID_{user}|| PW || F_U)^{SKHE} \bmod n$.

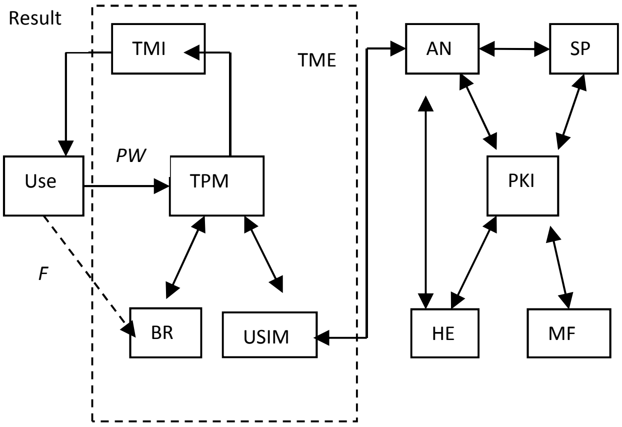

Figure 5: Security framework based on TMP and PKI [6]

The UE stores the symmetric key, SK_{TPM}, the biometric template, K_{Fu}, shared and $Cert_{TPM}$, and as well as integrity metrics of other components in the UE. The HE saves user's security credentials (ID_{User}, S, $Cert_{User}$) securely in its database [6].

Since 4G mobile networks architecture are based on an all-IP network platform authentication is perceived as a service performed at higher protocol layers regardless of the underlying technology. To accomplish mutual authentication between the UE and the 4G mobile network, two phases of authentication are proposed, namely local authentication where the UE checks the integrity and validity of the authentication parameters input by the user (password and biometric information) during initial boot up of the UE, and the remote mutual authentication between the UE, serving network (SN) and the HE when UE initially requests to be attached to the network [6].

5. Local Mutual Authentication

Local authentication procedure can be described in eight steps m_1 to m_8 as outlined as follows.

First, the USIM generates a token r_1 and sends an integrity check request D_1 with (r_1, ID_{USIM}) as m_1 to the TPM.

m_1: USIM →TPM: r1, ID_{USIM}, D_1

On receipt of m_1, the TPM issues a token r_2 and sends an integrity check request D_2 with (r_2, ID_{TPM}) to the BR.

m_2: TPM → BR: r_2, ID_{TPM}, D_2

Upon receiving m_2, BR encrypts its integrity metric D_3 with (r_2, ID_{TPM}) using K_{Fu} and responds with MAC_{BR} to the TPM.

m_3: BR → TPM: MAC_{BR}

$$MAC_{BR} = E(K_{Fu}, r2||ID_{TPM}||D_3) \quad (1)$$

Using the integrity metrics of BR and that of other components pre-stored in its internal database, the TPM checks whether the received MAC_{BR} is valid and also the integrity of other components of the TMP needed to perform the authentication operation are correct. Then the TPM generates a token r_3 and signs its own integrity metric D_4 with (r_1, r_3, ID_{USIM}). The TPM forwards the token r_3, $Cert_{TPM}$ and Sig_{TPM} to the USIM.

m_4: TPM → USIM: r_3, $Cert_{TPM}$, Sig_{TPM}

$$Sig_{TPM} = E[SK_{TPM}, r_1||r_3||ID_{USIM}||D_4] \quad (2)$$

The USIM then issues a token r_4 and calculates (C_1, Sig_{User}) as in equations 3 and 4. The SN's public-key parameters can be gained with the help of PK_{BP}. Then USIM sends them with (r_1, r_3) to SN to verify Sig_{TPM}. Where IDC_{User} is a unique identity of user's certificate and TS is a timestamp [6].

m_5: USIM → SN: r1, r_3, C_1, Sig_{TPM}, Sig_{User}, $Cert_{TPM}$.

$$C_1 = E(PK_{AN}, ID_{User}||IDC_{User}||r_4||TS) \quad (3)$$

$$Sig_{User} = E(SK_{User}, IDC_{User}||ID_{TPM}||r_1||r_3||TS) \quad (4)$$

The serving network SN decrypts C_1, checks TS if it is acceptable and turns to PKI to gain valid $Cert_{User}$ according to (ID_{User}, IDC_{User}). After verifying the validity of ($Cert_{TPM}$, Sig_{TPM}, Sig_{User}), SN pre-authenticates user. Then SN buffers (ID_{User}, $Cert_{User}$, $Cert_{TPM}$, r_4) temporarily and responds to USIM after checking result D_5 on TMP followed by MAC_{AN}.

m_6: SN → USIM: D_5, MAC_{AN} .

$$MAC_{AN} = E(r_4, r3||ID_{User}||ID_{TPM}||D5). \quad (5)$$

If the received MAC_{AN} is correct and (Sig_{TPM}, $Cert_{TPM}$) pair is valid according to D_5, then both TPM and SN are identified by USIM. As shown in Figure 6, USIM generates a token r_5 and sends (C_2, C_3) computed as in equations 6 and 8. The biometric comparison software (CS) is also encrypted in C_3 and sent from USIM to TPM. After USIM shows the current state of platform is trustworthy via TMI, the user is allowed to input his password and the BR to TPM. The captured biometric template (F_u') is encrypted in C_4, and including K_{Fu} which is then sent to the TPM [5].

m_7: USIM → TPM: C2, C3. BR → TPM: C_4.

C_2=E(PK_{TPM}, r_5||y||ID_{USIM}), (6)

K_{ST}=H[(r_5 ⊕r_3)||x||ID_{TPM}], (7)

C_3=E(K_{ST}, r_5||ID_{TPM}||F_U||CS). (8)

C_4=E[K_{BT}, ID_{BR}||ID_{TPM}||r_2||F_U']. (9)

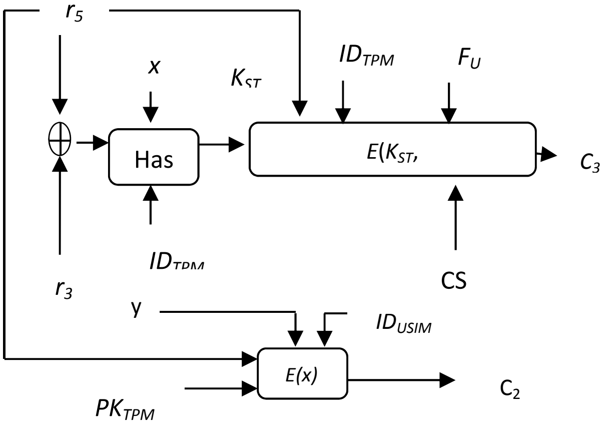

Figure 6: Data encapsulation algorithm in USIM [6]

Once the password PW is input by user and (r_5, y) pair is decrypted from the received C_2, the TMP, first, calculates

x=y ⊕H(PW) (10)

and K_{ST} as in equation 7. Using equation 8 the TPM then decrypts C_3 with K_{ST} and recovers (r_5, ID_{TPM}, BT, CS). If the (r_5, ID_{TPM}) contained in C_3 are both correct, the TPM checks whether equation 11 holds.

H(BT||PW) =ID_x (11)

If equation 11 holds, USIM is identified by TPM. Then TPM decrypts C_4 sent by BR and checks (ID_{BR}, ID_{TPM}, r_2) if they are all correct. TPM makes a comparison between the F_U and F_U' in use in CS to determine to what degree they match. If the match is achieved successfully, the user is authenticated by TPM.

Then, the TPM computes H(F_U ⊕ PW) and transfers C_5 computed as equitation 10 to USIM, where D_6 is the authentication result of the user. If (ID_{USIM}, $_{r5}$) contained in C_5 are both correct and, then the user is valid according to D_6. Both user and TPM are now identified by USIM.

m_8: TPM → USIM: C_5.

C_5=E(K_{ST}, ID_{USIM}||r_5||H(F_U⊕PW)||D_6). (12)

6. Remote Mutual Authentication

The SE-EPS AKA (mutual authentication) follows a seven step process as detailed in Figure 7, employing the encryption keys generated during phase 1 of the mutual authentication process. The UE initiates the network access Attach Request, first by using the HSS public PK_H to encrypt the (international mobile subscriber identity) IMSI and get the A (where A = {IMSI} $_{PKH,}$) and ID_{HSS} pair, which is then subsequently forwarded to the MME during the access request process. Upon receiving the access request of the UE, The MME uses the public PK_H to encrypt its own network identity (SNID), and then derive information B. The encrypted data A and B, regarded as the authentication data request, is sent to the HSS. Upon receiving the authentication data request from the MME, the HSS decrypts A and B to get the IMSI and SNID using its own public SK_H. The IMSI and SNID are validated by comparing them to the information stored in the HSS database. Once the verification process is over the HSS generates the random array RAND(1,...., n) and the authentication vector AV(1,....n).

As outlined in Figure 7, the SE-EPS AKA protocol calculates the following parameters:

$$K_{ASME} = s10_K\,(f3_K\,(\text{RAND}),\ f4_K\,(\text{RAND}), \text{SNID})$$

$$\text{XRES} = \text{RAND} \oplus \text{SNID} \quad \text{and;}$$

$$\text{AV} = \text{RAND} \,||\, \text{SNID} \,||\, K_{ASME} \,||\, \text{XRES}$$

This information is used by the HSS to calculate the encryption data C={AV(1,....,n), IMSI} $_{PK\ H}$ and sends to the MME as the response [5].

The MME then decrypts C to derive the authentication vectors AV(1,...,n) and IMSI. Amongst the authentication vectors AV(1,....,n) the MME chooses only one authentication vector AV(i) which has not been used before and extracts the random number RAND(i) and SNID found in the database. Exclusively the MME allocates the cipher key identifier KSI_{ASME}(i) to K_{ASME}(i) of the authentication vector AV(i) and utilize the IMSI and the algorithm shared by the MME and the UE to create S-TMSI used for access once more. After completing the one-time authentication and cipher key negotiation the UE and the MME both store the corresponding relation between KSI_{ASME}(i) and K_{ASME}(i). If access is required once again the UE and the SN will take into account the K_{ASME}(i) and in so doing confidential communication can be established without initiating the authentication process again. Finally the MME encrypts the RAND(i), S-TMSI (securely generated temporary mobile subscriber identity), K_{ASME}(i) and SNID by public key of the UE to calculate data D, which is then subsequently sent in subscriber authentication request to the UE. Thus eventually the operation MME → UE: D = {RAND(i), SNID, KSI_{ASME}(i), S_TMSI} $_{PKU}$ is completed in the process [6].

Once the UE has received the subscriber authentication request from the MME it decrypts D using the public key SK_U to recover RAND(i), S-TMSI and the SNID. The UE compares S-TMSI derived from the decryption of D to the one it has calculated to realize the authentication to HSS. If there is no consistency, it means the HSS is not valid and the process is terminated. In case of consistency being observed, the UE computes:

RES(i) = RAND(i) ⊕SNID and K_{ASME}(i) = sIO_K($f3_K$(RAND(i)), $f4_k$(RAND(i), SNID) and RES(i)

is considered the response to the subscriber authentication request sent to the MME [7].

The MME compares the RES(i) received to the XRES(i) of the authentication vector AV(1,...,n). If these two agree, the subscriber is valid. For any subsequent local communication the MME and the UE will consider the K_{ASME}(i) as the intermediate cipher key with which to create the encryption cipher key (CK(i)) and integrity cipher key (IK(i)), or else the while process in halted [7].

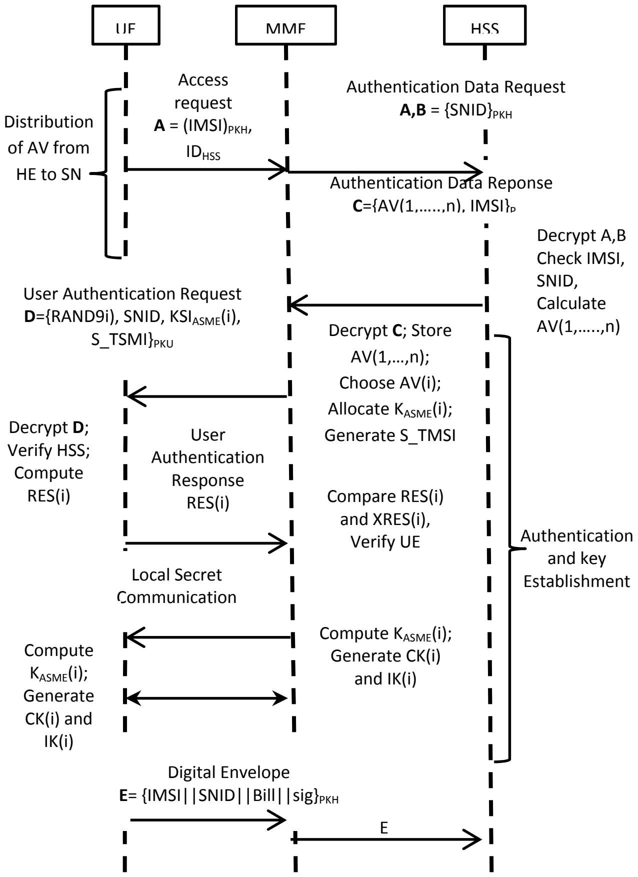
UE
MME
HSS
Access request
A = (IMSI)PKH, IDHSS
Authentication Data Request
A,B = {SNID}PKH
Distribution of AV from HE to SN
Authentication Data Reponse
C={AV(1,......,n), IMSI}P
Decrypt A,B
Check IMSI, SNID,
Calculate AV(1,.....,n)
User Authentication Request
D={RAND9i), SNID, KSIASME(i), S_TSMI}PKU
Decrypt C; Store AV(1,...,n);
Choose AV(i);
Allocate KASME(i);
Generate S_TMSI
Decrypt D;
Verify HSS;
Compute RES(i)
User Authentication Response RES(i)
Compare RES(i) and XRES(i),
Verify UE
Authentication and key Establishment
Local Secret Communication
Compute KASME(i);
Generate CK(i) and IK(i)
Compute KASME(i);
Generate CK(i) and IK(i)
Digital Envelope
E= {IMSI||SNID||Bill||sig}PKH
E

Figure 7: The SE-EPS AKA process [7]

Finally, the MME and the UE store the corresponding relation between the S-TMSI and (IMSI, AV(1,...,n), KSI_{ASME}(i), K_{ASME}(i), CK(i), IK(i)) in their internal databases. After the subscriber and the service network complete the transaction the UE can utilise its own cipher key SK_U to sign the IMSI, SNID and the business information bill for creating charging evidence {IMSI ‖ SNID ‖ bill ‖ sig}. Furthermore, in order to prevent leakage of IMSI and SNID, the public key PK_H is used to create the digital envelope E in which information is transferred to HSS via MME and can be used as evidence for presence and business participation of MME and subscriber as well as creation of related charging relation [6].

7. Implementation and Simulation Results

The simulation model developed is designed to test end-to-end QoS services in mobile wireless networks taking into account the three signaling scenarios outlined earlier. To achieve this session signaling delay and system response to congestion situations due to multiple calls and services requested at the same time are tested as a way of trying to define the QoS parameters of 4G mobile wireless networks. Situations simulated involve (a) UE1, the mobile caller, in the home domain or roaming, (b) UE2, the call recipient, in the home domain or roaming, (c) UE1 and UE2 involved inter- and intra-domain exchange of information and (d) both UE1 and UE2 attached to different wireless access technologies both in inter- and intra-domain scenarios [2].

The 4G network thus developed and designed consists of a pure but basic 4G network architecture interconnected to WiMAX and WiFi networks by a purely IPv6 backbone network, highlighting the presence of inter- and intra-domain interconnections.

The SEEP AKA protocol would be verified using the CryptoVerif tool in conjunction with OCaml.

8. Future Studies and Recommendations

The paper is a reflection of work in progress in which a model for a heterogeneous 4G mobile wireless network is simulated to test and verify the feasibility of integrating QoS and network securing signaling using ns-3 [8], a discrete-event network simulator. Developing a 4G core network with clearly defined network entities to allow an almost real industry-like live network environment that can seamless simulate nearly all network scenarios should be the thrust for future work. Such a network model will make it possible to test new services, especially QoS and security-related issues in order to cope with the ever-changing security threats of the ICT landscape. Current low cost and open-source simulation tools and models should be enhanced and developed.

9. Bibliography

[1] Dahlman, E., Parkvall, S., & Sköld, J. 4G LTE/LTE-Advanced for Mobile Broadband. Academic Press, 2011.

[2] Rui, P. & Sargnto, S. QoS and Session Signaling in a 4Gnetwork. http://www.researchgate.net/publication/4244345_QoS_and_session_signaling_in_a_4G_network

[3] http://caml.inria.fr/

[4] http://www.cryptoverif.ens.fr/

[5] Cadé D. & Blanchet B. From Computationally-proved Protocol Specifications to Implementations, In 7th International Conference on Availability, Reliability and Security (AReS 2012), pages 65-74, Prague, Czech Republic, August 2012. IEEE.

[6] Zheng, Y., He, D., Yu, W. & Tang, X. Trusted Computing-Based Security Architecture For 4G Mobile Networks. Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies. Computer Society, IEEE, 2005.

[7] Xiehua, L. & Yongjun, W.: Security Enhanced Authentication and Key Agreement Protocol in Next Generation Mobile Network, International Journal of Advancements in Computing Technology (IJACT) Vol. 4, No.3, February 2012.

[8] http://www.nsnam.org/z