2011 International Conference on Information and Network Technology
IPCSIT vol.4 (2011) © (2011) IACSIT Press, Singapore

Enhanced Authentication Protocol for Improving Security in 3GPP LTE Networks

J.Vijay Franklin[1], Dr.K.Paramasivam[2]

[1]Assistant Professor Senior Grade
Department of CSE, Bannari Amman Institute of Technology, Sathyamangalam
[2]Professor, Department of CSE, Bannari Amman Institute of Technology, Sathyamangalam

Abstract. The 3rd Generation Partnership Project(3GPP) standard is developing System Architecture Evolution(SAE)/Long Term Evolution(LTE) architecture for the next generation mobile communication system. To provide secure 3G-WLAN interworking in the SAE/LTE architecture, Extensible Authentication Protocol-Authentication and Key Agreement(EAP-AKA) is used.EAP-AKA protocol has several vulnerabilities such as disclosure of user identity, man-in-the-middle attack, Sequence Number (SQN) synchronization, and additional bandwidth consumption. The analyzes threats and attacks in 3G and proposes a new authentication and key agreement protocol based on EAP-AKA. The proposed protocol combines Elliptic Curve Diffie-Hellman (ECDH) with symmetric key cryptosystem to overcome the vulnerabilities present in the EAP-AKA protocol.

Keywords: EAP-AKA, ECDH, AES

1. Introduction

The next generation mobile communication system is being developed for secure and fast communication. The LTE architecture that is being developed by 3GPP provides more secure communication than Universal Mobile Telecommunication System (UMTS). To provide mutual authentication between User Equipment (UE) and Mobility Management Entity (MME) through E-UTRAN, the SAE/LTE architecture reuses UMTS-AKA. The SAE/LTE architecture reuses EAP-AKA to provide secure 3G-WLAN interworking. When a subscriber attempts to access WLAN, he sends International Mobile Subscriber Identity (IMSI) through Network Access Identifier (NAI) to the Access Point (AP). EAP-AKA is based on UMTS-AKA. For this reason, EAP-AKA can have not only vulnerabilities of UMTS-AKA but also vulnerabilities in 3G mobile communication.

EAP is an authentication framework providing for the transport and usage of keying material and parameters generated by EAP methods. There are many methods defined by RFCs and a number of vendor specific methods and new proposals exist. EAP is not a wire protocol instead it only defines message formats. Each protocol that uses EAP defines a way to encapsulate EAP messages within that protocol's messages.

Extensible Authentication Protocol (EAP) mechanism for Authentication and session key distribution that uses the 3rd Generation Authentication and Key Agreement mechanism, specified for Universal Mobile Telecommunications System (UMTS) in [TS33.102] and for CDMA2000 in [S.S0055-A]. UMTS and CDMA2000 are global third generation mobile network standards that use the same AKA mechanism. The Global System for Mobile communications (GSM) is a second generation mobile network standard, and EAP-SIM specifies an EAP mechanism that is based on the GSM authentication and key agreement primitives. AKA is based on challenge-response mechanisms and symmetric cryptography. AKA typically

runs in a UMTS Subscriber Identity Module (USIM) or a CDMA2000 (Removable) User Identity Module ((R)UIM). In this document, both modules are referred to as identity modules.

Authentication and key agreement is a security protocol used in 3G networks. AKA is also used for one time password generation mechanism for digest access authentication. AKA is a challenge-response based mechanism that uses symmetric cryptography. AKA provides procedures for mutual authentication of the MS and serving system. The successful execution of AKA results in the establishment of a security association (i.e., set of security data) between the MS and serving system that enables a set of security services to be provided.

Major advantages of AKA include larger authentication keys (128-bit), stronger hash function (SHA-1), support for mutual authentication, support for signaling message data integrity, support for signaling information encryption, support for user data encryption, protection from rogue MS. AKA is a mechanism which performs authentication and session key distribution in UMTS networks. AKA is a challenge-response based mechanism that uses symmetric cryptography. AKA is typically run in a UMTS IP Multimedia Services Identity Module (ISIM), which resides on a smart card like device that also provides tamper resistant storage of shared secrets.

Elliptic Curve Diffie-Helman protocol (ECDH) is one of the key exchange protocols used to establishes a shared key between two parties. ECDH protocol is based on the additive elliptic curve group. ECDH begin by selecting the underlying field GF(P) or GF(2k), the curve E with parameters *a*, *b* and the base point *P*. The order of the base point *P* is equal to *n*. The standards often suggest that we select an elliptic curve with prime order and therefore any element of the group would be selected and their order will be the prime number *n*. At the end of the protocol, the communicating parties end up with the same value *K* which is a point on the curve.

Alice	Communication	Bob
Choose a random number a ∈ F_q		Choose a random number b ∈ F_q
Compute aP		Compute bP
Retrieve bP	aP → ← bP	Retrieve aP
Compute abP		Compute abP

Table1.Operations of ECCH

Setup: Alice and Bob agree on a common group G a nd a common group element *g*. Then Alice chooses a secret number *a*, which serves as her secret key and Bob chooses a secret key *b*.

Communication: Alice computes *ga* and sends it to Bob over a public channel.Bob sends *gb* to Alice. Although *ga* and *gb* are closely related to *a* and *b* respectively, the hardness of the DLP ensures that the secret keys cannot be computed from them in a practical situation. Therefore, *ga* and *gb* can serve as public keys, corresponding to the private keys of Alice and Bob respectively.

Final step: Alice takes Bob's public key and computes *(gb)a = gab*, Bob computes *(ga)b = gab*. As we see, Alice and Bob obtain the same result and this result could

not be computed by an adversary who only knows the public keys. Therefore Alice and Bob have agreed on a shared secret key. Note that hardness of the DLP does not guarantee the security of the Diffie-Hellman protocol to computing *gab* from *ga* and *gb* may be easier than computing *a* from *ga* or *b* from *gb*.

2. Design Of Elliptic Curve Deffie Hellman Key Exchange

What one man knows, nobody knows, what two men know, everyone knows. The big problem in cryptography is key distribution. If Bob and Alice have to both know a secret key in order to communicate secretly. So we need a one way operation where you can calculate the encryption key from the decryption key, but you cannot calculate the decryption key from the encryption key.

Then Alice chooses a decryption key at random from a very large set of possible keys (in the case of this particular implementation one of 2^240 possible decryption keys, that is 1 of 1700 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 possible keys) then she calculates the encryption key, and publishes it in plaintext to Bob, and anyone else, and to the whole world. So the decryption key is her secret key, which she alone knows, and which she alone can use to sign messages she sends, and decrypt messages sent to her, and the encryption key is her public key, which everyone knows.have to try 2^120 keys (1 329 000 000 000 000 000 000 000 000 000 000 000 000 keys) in order to obtain enough information to deduce the correct key. (He would also have to have very large storage)

Elliptic curves provide us with a suitable one way operation:

The work factor for this implementation is believed to be the square root of the number of possible keys, not the number of possible keys. An attacker would An elliptic curve is not actually part of an ellipse, rather it is a curve of the form y.y + x·y = x.x.x + a·x + b. For the purposes of elliptic curve cryptography x and y will not be real numbers but will be finite fields, typically integers modulo some quantity. The important and interesting property of an elliptic curve is that given two points on an elliptic curve, there is a function that gives us a third point on the curve. This function is associative and commutative. If P, Q, and S are points on the elliptic curve, then f(P,Q) = f(Q,P) and f(P,f(Q,S)) =f(f(P,Q),S), so it is convenient to represent this operation as addition of points on a curve, so that the preceding relations are represented in as P+Q = Q+P and P + (Q+S) = (Q+P)+S.

3. Methodology

EAP-AKA provides mutual authentication between the UE and the AAA server. That is, EAP-AKA performs a procedure of authentication and key agreement between 3G and Non- 3GPP. The AAA server requests again the user identity because immediate nodes can modify user identity such as IMSI included in EAP Response/Identity message. Therefore, if the UE receives EAP Request /AKA Identiy message, the UE should send EAP Response/AKA Identity message which must contain the same user identity included in EAP Response/Identity message to the AAA server. The AAA server will use user identity received from EAP Response/AKA-Identity message in the rest of the authentication and key agreement procedure

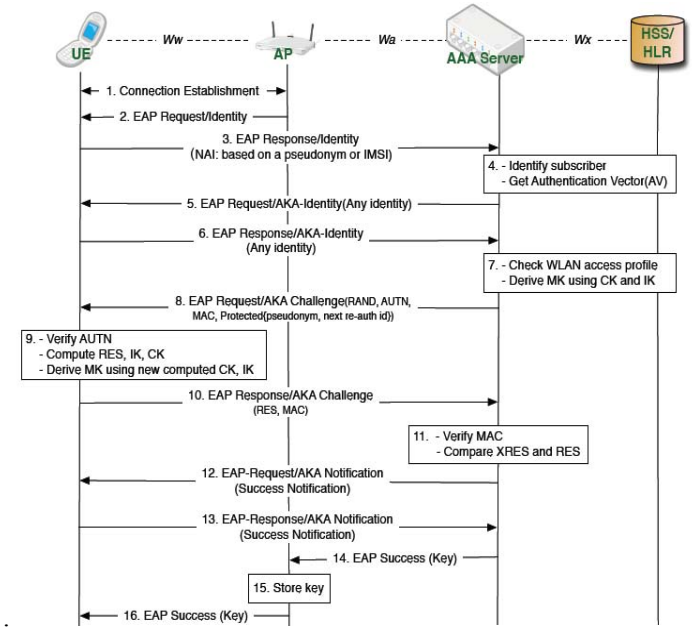

Fig 1. Procedure of EAP-AKA

4. Design of EAP-AKA Protocol

A new authentication and key agreement protocol based on EAP-AKA.

A. Assumption

In the proposed protocol, we assume the following:

- A secure channel is established between the AAA server and the HSS.
- The UE can identify the ID of AAA server and AP in which it is able to access now.

B. The Workflow of the Protocol

Our protocol consists of four procedures which are shown in Fig. 1.

Notation	Description
U, A,H	Denote the UE, the AAA server and the HSS, respectively
cID_{UE}	Current temporary ID of UE
IDx	ID of entity x
Tx	Timestamp generated by entity x
g^i_K	Key generation function using the key K
f^1_K	MAC generation function using the key K
f^2_K	cID_{UE} generation function using the key K
RANDx	Random number by entity x
Kxy	Symmetric keyshared between entity x and y
TK	Temporary Key

Table 2. Parameter used in authentication procedure

1) Initialization:

Step 1. A connection is established between the UE and the AP.

Step 2. To get user identity, the AP sends EAP Request/Identity message to the UE.

2) Registration and Generation of TK:

Step 3. The UE generates T_U and computes $MAC_U=f^1K_{UH}(T_U||ID_{AAA}||ID_{AP})$ using the K_{UH}.
In addition, the UE computes cID_{UE} to prevent the disclosure of IMSI. cID_{UE} can be computed as $f^2K_{UH}(IMSI)$. Therefore, the UE sends cID_{UE}, T_U,MAC_U, and ID_H to the AP. Meanwhile, the UE computes $TK=g^1K_{UH}(T_U)$.

Step 4. The AAA server transmits cID_{UE}, T_U, MAC_U, and ID_{AAA} to the HSS using ID_H received

Step 5. The HSS checks MAC_U. As a result, the UE can verify ID_{AAA} and T_U and authenticate the UE. The procedure of checking MAC_U is as follows:

a) The HSS retrieves ID_{AP} , ID_{AAA}, and T_U from MAC_U.

b) The HSS verifies whether or not ID_{AAA} retrieved from MAC_U equals ID_{AAA} which sent Step 4 message(cID_{UE}, T_U, MAC_U, ID_{AAA}) to the HSS.

c) The HSS verifies whether T_U is in the correct range and then verifies whether T_U retrieved from MAC_U equals received T_U. If the result is correct, the HSS can authenticate the UE and prevent replay attack.

After checking MAC_U, the HSS derives IMSI from cID_{UE} using K_{UH}. The HSS searches the entire DB which stored user identity such as IMSI to identify the requested UE. The HSS computes $TK=g^1K_{UH}(T_U)$ and generate $RAND_H$. Using $RAND_H$ the HSS computes $MAC_H=f1K_{UH}(RAND_H)$.

Step 6. The HSS sends $AUTH_H$, TK, and ID_{AP} to the AAA server. ID_{AP} was obtain from MAC_U. We already assumed that a secure channel was established between the HSS and the AAA server. As a result, TK is secure against attackers although TK is plaintext on the air.

Step 7. The AAA server stores TK, $AUTH_H$, and ID_{AP} .

3) Authentication and Key Agreement:

Step 8. The AAA server generates $RAND_A$ and computes MAC_A. Afterward, the AAA server selects random number *"a"* and computes *a*P on E.Elliptic Curve Diffie-Hellman(ECDH):

User A and B publicly agree on an elliptic curve E over a large finite field F and a point P on that curve. The user A and B each selects random number *"a"* and *"b"*, respectively. Using elliptic curve point-addition, user A and B each publicly compute *"aP"* and "*bP"* on E. Finally, user A and

B each compute "*abP"* using private and public values. As a result, solving ECDH is a computationally difficult problem.

Step 9. The AAA server sends $AUTH_A=(MAC_A||RAND_A||RAND_H)$ and *"aP"* to the UE.

Step 10. The UE verifies MAC_A. The procedure of verifying MAC_A is as follows:

a) The UE computes $MAC`_H=f^1K_{UH}(RNAD_H)$. The $RAND_H$ is derived from $AUTH_A$ in Step9.

b) The UE computesMAC`A=f1TK($MAC`_H||RAND_A||RAND_H$). The $RAND_H$ and $RAND_A$ are derived

c) The UE verifies whether $MAC`_A$ equals $MAC_A$ or not. If $MAC`_A$ is not same MAC_A, the HSS or the AAA server is not valid. Therefore, the UE terminates the procedure.

The UE can authenticate the HSS and the AAA server by verifying MAC_A. As a result, verifying MAC_A prevents replay attack and man-in-the-middle attack. The UE selects random number *"b*" and computes *"bP"* on E.Subsequently, using *"aP*" received from the AAA server in Step 9, the UE can compute symmetric key $K_{UA} = g^2_{TK}(abP)$. Finally, the UE computes $MAC_{UA} = f^1K_{UA}(RAND_A||bP)$ using K_{UA} shared between the UE and the AAA server.

Step 11. The UE transmits *"bP*" and MAC_{UA} to the AAA server and concurrently computes CK and IK.

Afterward, the UE computes MSK using CK and IK as EAP-AKA.

Step 12. Using *"bP*" received from the UE in Step 11, the AAA server can compute K_{UA}. Then the AAA server verifies MAC_{UA}. In other words, the AAA server verifies whether or not $RAND_A$ included in MAC_{UA} equals $RAND_A$ generated from the AAA server in Step 8.

If two values are same, the AAA server can authenticate the UE. The AAA server computes CK and IK. Finally, the UE computes MSK using CK and IK as EAP-AKA.

4) Transmission of MSK:

Step 13. The AAA server sends IDAP||MSK with EAP Success message to the AP. ID_{AP} was received from the HSS in Step 6.

Step 14. The AP verifies whether received ID_{AP} equals AP's own ID or not. If the result is correct, the AP stores MSK. Otherwise the AP does not store MSK and then terminates the execution.

Step 15. The AP sends IDAP||MSK with EAP Success message to the UE.

Step 16. The UE verifies whether or not IDAP received from the AP in Step 15 equals IDAP used in Step 3 to compute MAC_U, and then verifies whether or not MSK received from the AP in Step 15 equals MSK generated in **Step 11**. If the result is correct, the procedure of authentication and key agreement is successful. Consequently, the UE can securely use WLAN service using MSK.

5. Conclusion

The proposed protocol are combined with the AKA with the Extracts of the Elliptic Curve Cryptography (ECC), Diffe-Hellman key exchange to improve the security.The system provides security by modifying the procedure of the authentication process. The authentication header for the transaction is dually verified by the authentication server in order to provide the authorization. In near future the fast re-authentication are to be concentrated and produced the well effective protocol by using the AES and the entire communication can evaluated by the parameters of Secure Socket Layer .

6. References

[1] H. Mun, K. Han and K. Kim, "3G-WLAN Interworking: Security Analysis and New Authentication and Key Agreement based on EAP-AKA", International symposium on Taiwan, April 2009.

[2] Third Generation Partnership Project (3GPP), "Architecture Enhancements for non-3GPP accesses (Release 8)",3GPP TS 33.402 v8.3.0, September 2008.

[3] Third Generation Partnership Project (3GPP), " 3G System Architecture Evolution (SAE): Security architecture (Release 8)" , 3GPP TS 33.401 v8.1.1, October 2008.

[4] Third Generation Partnership Project (3GPP), " Rationale and track of security decisions in Long Term Evolved (LTE) RAN/3GPP System Architecture Evolution (SAE) (Release 8)", 3GPP TS 33.821 v1.0.0, Dcember 2007.

[5] J. Arkko and H. Haverinen, “Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)”, IETF,RFC 4187, January 2006.

[6] L. Han, “A Threat Analysis of the Extensible Authentication Protocol”, IETF, RFC 4286, April 2006.

[7] P. Funk and S.Blake-Wilson, “EAP Tunneled TLS Authentication Protocol, draft-ietf-pppext-eap-ttls-05”, IETF, July 2004.

[8] A. Palekar, D. Simon, S. Josefsson, H. Zhou and G. Zorn, “Protected EAP Protocol (PEAP) Version 2” IETF, October 2004.

[9] H. Haverinen and J.Salowey, “EAP SIM Authentication, draft-arkko-pppexteap- sim-12”, IETE, October 2003.

[10] PlanetMath-Elliptic Curve Diffie-Hellman key exchange, http://planetmath.org/encyclopedia/Diffie HellmanKeyExchange.html

J.Vijay Franklin received BE degree in Bannari Amman Institute of Technology and ME degree in Vishwarya Technology University and Phd degree in Anna University of Technology –Coimbatore. He is currently doing his research work in Networ Security.He is an member of ISTE.

Dr.K.Paramasivam received Phd in the field of VLSI design from Anna University of Technology.He published more than 25 paper on both national and international conferences and journals.He is an member in ISTE.