Design and Implementation to Authentication over a GSM System Using Certificate-Less Public Key Cryptography (CL-PKC)

Imran Memon · Mohammed Ramadan Mohammed · Rizwan Akhtar · Hina Memon · Muhammad Hammad Memon · Riaz Ahmed Shaikh

Abstract Recent years, the mobile technology has experienced a great increment in the number of its users. The GSM's architecture provides different security features like authen-tication, data/signaling confidentiality and secrecy of user yet the channel is susceptible to replay and interleaved. It always remains relevant as it is important in all types of application. Global system for mobile (GSM) communications has become the most popular standard for digital cellular communication. The GSM security system depends on encryption, authenti-cation algorithms and information from SIM card. In this research paper, we proposed the design and implementation of a new authentication scheme by using certificate-less public key cryptography (CL-PKC) over the GSM system was attempted to miss some system detail. This research paper, we also proposed the GSM system and its security and public key cryp-tography with a focus in the CL-PKC; the CL-PKC is a simple, useful and robust security scheme designed and implemented over GSM. Our approach is more efficient than other competing topologies. We solved the GSM problem in A3 algorithm such as eavesdropping and this problem solved by CL-PKC because of its robustness against this type of attack by providing mutual authentication make the system more secure.

Keywords GSM security · GSM authentication · Certificate less public key cryptography · Public key cryptography

I. Memon (✉)
College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, Zhejiang, China
e-mail: Imranmemon52@zju.edu.cn; imranmemon52@cs.zju.edu.cn

M. R. Mohammed · M. H. Memon · R. A. Shaikh
School of Computer Science and Engineering, University of Electronic Science and Technology, Chengdu 611731, Sichuan, China

R. Akhtar
School of Communication and Information Engineering, University of Electronic Science and Technology, Chengdu 610054, Sichuan, China

H. Memon
Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Pakistan

1 Introduction

These mobile phones are used on a daily basis by hundreds of millions of users over radio links due to the fact that unlike a fixed phone which offers some level of physical security (i.e. Physical access is needed to the phone line for listening in) with a radio link anyone with a receiver is able to passively monitor the airwaves. Therefore, it is highly important that reasonable technological security measures are taken to ensure the privacy of user's phone calls and text messages (data), as well to prevent unauthorized use of the service. In general the GSM system [1] is less secure than a wired communication as it opens the door to eavesdroppers with appropriate.

Authentication and encryption in GSM both rely on a secret key Ki that is unique to the subscriber. Copies of Ki are held on the SIM and in the Authentication Center (AuC), and Ki is never transmitted across the Air interface. The detect security vulnerabilities in algorithm (A3) which is responsible for the authentication process as explained in this research. An important and well known shortcoming of GSM security is that it does not provide a means for subscribers to authenticate the network. This oversight allows for false base station attacks. Man-in-the-middle is the capability whereby the intruder puts itself in between the target user and a genuine network and has the ability to eavesdrop, modify, delete, re-order, replay, and spoof signaling and user data messages exchanged between the two parties. The required equipment is modified BTS in conjunction with a modified MS. The above reasons make the GSM system needs a hard work to calibrate the communication parameters (Bandwidth, Time delay) with a high security scheme.

The main aim of this research is that attempt to implement a new approach of authentication scheme based on certificate less public key cryptography (CL-PKC) which is perfect for mobile communication because CL-PKC no need for a certificate for the public key so this feature provides low memory, low bandwidth, and low cost properties which are useful for mobile communication application. The name GSM first comes from a group called Group Special Mobile (GSM) [2] which was formed in 1982 by the European Conference of Post and Telecommunications Administrations (CEPT) to develop a pan-European cellular system that would replace the many existing incompatible cellular systems already in place in Europe.

Our contribution is quantities and qualitative method to design and implement the new approach for authentication Over GSM.

2 Related Work

Security has become an essential topic in current mobile and wireless networks. The security tools and techniques used to attack such networks also increases. To protect the entity from any third party attacks, such as revealing a particular identity, data modification or data hijacking, eavesdropping, impersonating an identity, Protection mechanisms are used. Devoted technologies for securing data and communication are mandatory in wireless networks which vary according to the category of wireless technology deployed. Security in mobile networks handles a diversity of issues from authenticating a user accessing a network to data integrity and data encryption.The digital communication system, security management is very easy to be realized for GSM. In the GSM system, security management consists of four parts, authentication and encryption, TMSI reallocation and equipment identification, but there is some possible vulnerability which is concerned among many researches. Most them are the weakness in the algorithm used in both authentications to mobile user (COMP128) and encryption to communication data (A5/1, A5/2). In past years, these algorithms are considered to be secure, but many attacks become possible nowadays such as brute force attacked also can be done within a considerable amount of time. Recently, countermeasure against these vulnerabilities has been considered and during implementation. A5/3 and MILENAGE algorithm are expected to be used for the new security system. They are also open for the cryptographic community to help examine the algorithm which is differed from the past strategy and trying to keep the algorithm in secrecy. The growth of technology and discovering new methodology, it still is possible that potential attack can be found and may cause this new algorithm proposed for GSM security system become vulnerable again. Security algorithm has not broken yet, GSM architecture would still be vulnerable to attacks targeting the operator's backbone network or HLR and to various social engineering scenarios in which the attacker bribes an employee of the operator [3]. The security methods standardized for the GSM System made it the basic standard for the recent generations e.g. 3G and LTE. Although the confidentiality of a call and the anonymity of the GSM subscriber 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 realized using digital systems and signaling. Particularly compared to the previous analog systems. Part of the enhanced security of GSM is due to the fact that it is a digital system utilizing a speech coding algorithm, Gaussian Minimum Shift Keying (GMSK) digital modulation, slow frequency hopping, and Time Division Multiple Access (TDMA) time slot architecture. To intercept and reconstruct this signal would require more highly specialized and expensive equipment to perform the reception, synchronization, and decoding of the signal. In addition, the authentication and encryption schemes, but in the last years there are many attacks on the GSM system have been done. The Authentication process lies before the ciphering process, and it is a most important process in the GSM security, because it is ensured that the ciphering scheme achieved for the right party. There are many works took GSM security, some of them are only analytic study, and some of these studies are improvements of GSM security, and in the following section the most important related works are posted [4] and described the attacks over the IP networks and their suggested solutions to counter measures for the attacks in 2.5G and 3G Cellular IP Networks. Forsgren et al. [5] Proposed Security and Trust of Public Key Cryptography Options for Host Identity Protocol to give verified identities to host using public key certificates and certificate-less public key cryptography (CL-PKC). Various public key approaches for Host Identity Protocol peer authentication have been discovered. The Identity based approaches such as CL-PKC allow only PKG's to be validated, while digital signature algorithms deliver built-in key validation. Public key certificates are used, revocation lists should be checked at least every time these lists are updated. CL-PKC certificates are used to attach trust to PKG parameters, not to public user keys. In this case, revocation lists are much smaller and it is necessary to verify certificates only in Base Exchange. Seo et al. [6] solve the key escrow and key management problem, Proposed Certificate Less Hybrid Sign-Cryption without pairing operation and implemented hybrid sign-cryption scheme. Islam et al. [7] Proposed secure and efficient certificate-less strong designated verifier multisignature scheme using elliptic curve cryptography (ECC) and bilinear pairings. This scheme allows to number of signatures generate common signature design verification and verified multisignature and this scheme is useful one document required to be authenticated by a number of persons, applicable in several applications like workflow, decision making, processes, etc. Way et al. [8] certificate-less proxy re-encryption (CL-PRE) scheme for data sharing to the cloud. In CL-PRE, a data owner and identified security flaws with several certification and key establishment protocols for mobile communications, they establish that the protocols do not provide authentication as intended. Liu et al. [9] presented the most important security flaws of the Self-Generated-Certificate Public Key Cryptography to captures denial of decryption attack and also provide self-generated certificate public key encryption scheme. Further, they implemented and signature and certificate scheme. Cho et al. [10] presented composite trust-based public key management (CTPKM) with no centralized trust entity with the goal of maximizing performance and fully distributed trust-based public key management approach for MANETs using a soft security mechanism based on the concept of trust, using hard security approaches, as in traditional security techniques, to eliminate security vulnerabilities. Meyer et al. [11], reduce the signaling overhead and add some other security features, they proposed a new generalized approach in their paper based on asymmetric cryptography for user/network authentication and communication encryption in GSM/GPRS and UMTS with reduced signaling overhead.

3 Methods and It's Techniques

Public key cryptography (PKC) permits users to create secure transmission channels without the need for any previous exchange of secret keys. Each user generates a pair of keys called public and private key. The former is used for encryption and the latter for decryption. This ground breaking idea solves the issue of key distribution and reduces the number of required crypto-keys. In fact, it shifts the problem of key distribution to the problem of binding a user with his key pair. This binding is really at the core of PKC security. Fortunately, in practice, it is much easier to certify particular binding than to deliver the keys themselves.

There are several methods, of which public-key infrastructure is the best known. Public key infrastructure (PKI) proves authenticity of users' keys by means of certificates. The organizational level sets of authorities which handle digital certificates. PKI has been often the choice, it has significant shortcomings. First the infrastructure is heavy-weight and rather expensive. Moreover, certificates must be verified by users (whether they match the correct identity or not), but non-technical users usually have problems with that. This is usually the case in big deployments of PKI. Certificate-less cryptography (CL-PKC) [12] is an interesting alternative to traditional PKI. It makes use of identities, which are users' public keys formed of arbitrary strings, in place of certificates. Its infrastructure is lightweight and can be deployed at much lower cost. Moreover, it offers transparent encryption, so that non-technical users could easily secure their data.

3.1 Public Key Infrastructure (PKI)

Public key infrastructure is the most popular solution for proving authenticity of public keys. Similar to the web of trust, it applies certificates to confirm the relation between a user and his public key. The PKI's model is centralized and hierarchical; it's composed of special nodes called Registration Authority (RA) and Certificate Authority (CA), which make up the infrastructure. The authorities are trusted third parties, which are not run by ordinary users. The role of the RA and CA is as follows: Registration Authority collects requests from users to issue digital certificates for their public keys. Before CA can sign the key, Registration Authority must verify the credentials of the client. Upon successful verification, Certificate Authority generates a certificate, which contains user's key, identity and CA's signature. The fundamental question is: who granted CA the right to authorize users' keys? Usually it is another Certificate Authority, who is even more trusted. PKI [13] form hierarchical structure in which CA's keys are further signed by other CA's. The roots

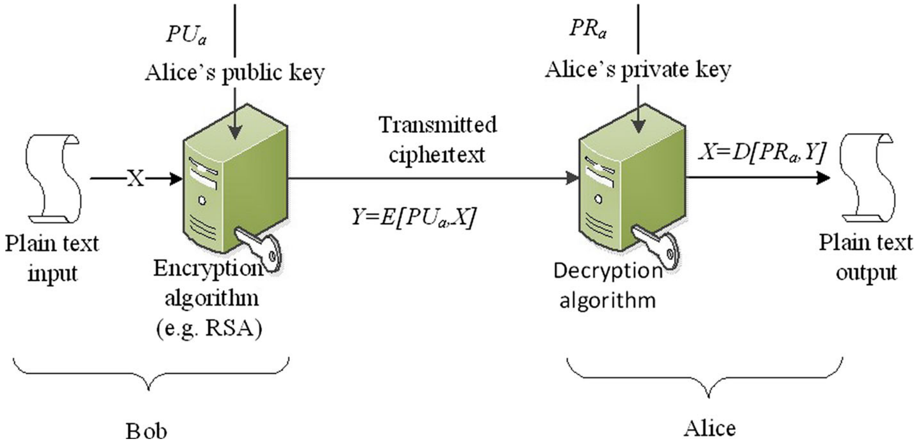


Fig. 1 Public key cryptography scheme

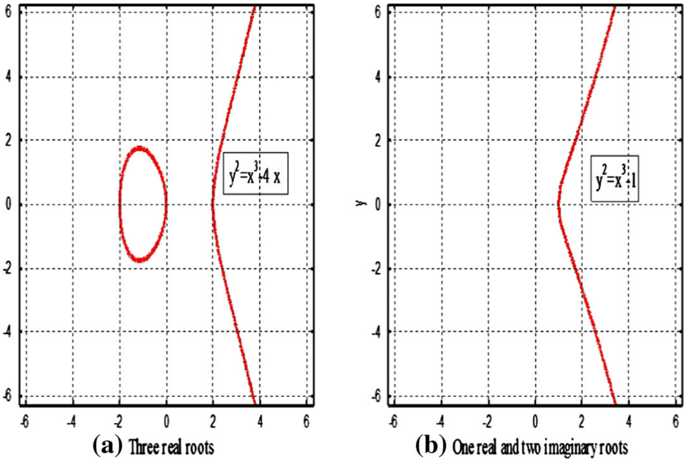
Certificate Authorities sign their keys themselves and for this reason those certificates are usually deployed by software. The development of public key cryptography is the greatest and perhaps the only true revolution in the entire history of cryptography. It is asymmetric, involving the use of two separate keys, in contrast to symmetric encryption, that uses only one key. Anyone knowing the public key can encrypt messages or verify signatures, but cannot decrypt messages or create signatures, counterintuitive though, this may seem. The use of two keys has profound consequences in the areas of confidentiality, key distribution, and authentication. It works by the clever use of number theory problems that are easy one way but hard the other. Note that public key schemes are neither more nor less secure than a private key (security depends on the key size for both), nor do they replace private key schemes (they are too slow to do so), rather they complement them. Both also have issues with key distribution, requiring the use of any suitable protocol in Fig. 1.

3.2 Elliptic Curve Cryptography (ECC)

Elliptic curve cryptography is a public key cryptosystem [14] based on the algebraic structure of elliptic curves over finite a field that relies on the believed difficulty of the elliptic curve discrete logarithm for its security. It accepted as an alternative to cryptosystems such as RSA and ElGamal over finite fields. Elliptic curve cryptography makes use of elliptic curves in which the variables and coefficients are all restricted to elements of a finite field. Two families of elliptic curves are used in cryptographic applications: prime curves over Z_p (best for software use), and binary curves over GF (2^m) (best for hardware use).There is no obvious geometric interpretation of elliptic curve arithmetic over finite fields. The algebraic interpretation used for elliptic curve arithmetic overdoes readily carry over.

ECC properties

* Addition is analogous to modulo multiplication.
* Repeated addition is analogous to modulo exponentiation.
* The inverse of a point (x, y) is $(x, -y)$, where $-y$ is the additive inverse of y. For example, if we have a curve $y^2 = x^3 - 4x$, and p = 13, the inverse of (4, 2)is (4, 11). Because 2+11 mod 13 = 0
* ECDLP: is a "hard" problem, equivalent to solving the discrete logarithm:



(a) Three real roots

(b) One real and two imaginary roots

Fig. 2 Elliptic curves

- $Q = kP$, where Q, P belong to a prime curve.
- is "easy" to compute Q given k, P
- but "hard" to find k given Q, P

This property, known as the elliptic curve logarithm problem (ECLP)

The general equation of an elliptic curve is:

$$y^2 + b_1xy + b_2y = x^3 + a_1x^2 + a_2x + a_3$$

The straightforward way of computing a point multiplication is through repeated addition, with the exception of the first addition since adding a point to itself is usually undefined since the slope of the line through the point is (0). However, this is a fully exponential approach to computing the multiplication.

Elliptic curves over real numbers use a special class of elliptic curves of the form:

$$y^2 = x^3 + ax + b$$
$$\text{Where } 4a^3 + 27b^2! = 0$$

The left-hand side has a degree of 2 while the right-hand side has a degree of 3. This means that a horizontal line can intersect the curve in three points if all roots are real. However, a vertical line can intersect the curve at most in two points.

The following figure shows two elliptic curves with equations:

$$y^2 = x^3 - 4x \quad \text{and} \quad y^2 = x^3 - 1$$

However, the first has three real roots ($x = -2, x = 0$, and $x = 2$), but the second has only one real root ($x = 1$) and two imaginary ones. The Fig. 2 shown an example for elliptic curves which used in cryptographic systems:

Also we can make all points on an elliptic curve are belong to Abelian Group so called (EC over Real Numbers), e.g. A tuple P(x1, y1) represents a point on the curve if x1 and y1 are coordinates of a point on the curve that satisfy the equation of the curve, i.e. the points P(2, 0), Q(0, 0), R(−2, 0), S(10, 30.98) are all points on the curve (y2 = x3 − 4x), so each point is represented by two real number.

3.3 Pairings

Pairing based cryptography [15] is a field in which cryptosystems are constructed upon pairings. Most of identity-based schemes, among which are IBE, CBE and CL-PKE, belong to this area. To use pairing between elements two cryptographic groups to a third group to construct cryptographic systems. If the same group used for the first two groups, the pairing is called symmetric and is a mapping from two elements of one group to an element from a second group. This way pairings can be used to reduce a hard problem in one group to a different, usually easier problem in another group. Simply the group supplied with a bilinear mapping such as the Weil pairing. In practice, it allows to solve certain problem in one group, even if the problem is said to be hard in another group. Although pairing is a more general concept, its definition within cryptography is as follows.

Let G_1 and G_2 be Abelian groups, written additively.
Let n be a prime number such that $[n]\ P$ for all P in G_1 and G_2.
Let G_3 be a cyclic group of order n, written multiplicatively.
Then a pairing is a map:
$G_1 \times G_2 \rightarrow G_3$
Admissible pairing if satisfies the following properties:

1. Bilinear:

$$e\left(P+P', Q\right) = E\left(P, Q\right) e\left(P', Q\right)\ for\,all\ P, P' \in G1, Q \epsilon G2$$
$$e\left(P, Q+Q'\right) = e\left(P, Q\right) e\left(P, Q'\right)\ for\,all\ P \in G1, Q, Q' \epsilon G2$$

2. Non-degenerate:

$$\text{For all non-zero P} \epsilon\ G_1,\ \text{there is a } Q \epsilon G_2 \text{ such that e } (P, Q) \neq 1$$
$$\text{For all non-zero } Q \epsilon G_2,\ \text{there is a } P \epsilon G_1 \text{ such that e } (P, Q) \neq 1$$

There exists P and Q in G1 such that $\hat{e}(P, Q) \neq 1$

3. Computable:
There is an efficient algorithm to compute $e(P, Q)$ for any $P \epsilon G_1$, $Q \epsilon G_2$

The above definition is sometimes called the asymmetric pairing, when groups G_1 and G_2 are the same group, then we can say that pairing is symmetric. Moreover, it may be hard to find discrete logarithms in the three groups, but the Decision Diffie-Hellman problem (DDH) might be easy in G_1 and G_2. The DDH problem is important from a theoretical point of view when it comes to security of pairing-based cryptosystems. The concrete implementations of pairings usually involve modifing Weil or Tate pairing.

Weil Pairing

The Weil pairing calculation as following:

Given two points P, Q ϵ E[n] we show how to compute e(P,Q) ϵ F_p^* using O(log p) arithmetic operations in F_p.

We assume $P \neq Q$. We proceed as follows:

* Pick two random points R_1, $R_2 \epsilon E[n]$.
* Consider the divisors $A_P = (P + R_1) - (R1)$ and $A_Q = (Q + R_2) - (R_2)$.

These divisors are equivalent to $(P) - (O)$ and $(Q) - (O)$ respectively. Hence, we can use A_P and A_Q to compute the Weil pairing as:

$$e(P, Q) = f_P(A_Q)/f_Q(A_P) = f_P(Q + R_2) f_Q(R_1)/f_P(R_2) f_Q(P + R_1)$$

This expression is well defined with very high probability over the choice of R_1, R_2 (the probability of failure is at most O (log (p / p)). In the rare event that a division by zero occurs during the computation of e (P, Q) we simply pick new random points R_1, R_2 and repeat the process.

3.4 Identity-Based Cryptography (IBC)

Identity Based Cryptography (IBC) Identity-based cryptography [16] is a type of public-key cryptography in which a publicly known string representing an individual or organization is used as a public key. The public string could include an email address, domain name, or a physical IP address. IBC allows implementing transparent data encryption in various communication systems, such as email or cellular telephone. The transparent encryption is not important from a theoretical point of view; it's a significant factor in real world implementations. Usually non-technical users have no knowledge on computer security and for this reason misuse security related software. Phishing, perhaps, is the best example of how to deceive an average user and make him reveal sensitive data even if the connection seems protected properly. Shamir similarly proposed identity based encryption, which appeared particularly attractive since there was no need to acquire an identity's public key prior to encryption. However, he was unable to come up with a concrete solution and identity based encryption remained an open problem for many years. Aims of public key infrastructure and identity-based cryptography (IBC) are quite similar but the way they approach certain problems are slightly different. First of all, users of IBC can set the public key to be an arbitrary. The key can be something easily memorable like an email address or a phone number. Secondly, there are no certificates binding a user with his public key; Bob's unique ID string guarantees that no user besides he should be able to decrypt the content. IBE relies on a trusted third party, often called Private Key Generator (PKG), which is responsible for generating secret keys corresponding to users' public keys. PKG has its own key pair called master public key and master private key (aka master key). The latter is involved in the process of creating user's private key from the given identity. Security of the scheme is based on elliptic curve analogue of the computational Diffie-Hellman assumption, so the underlying mathematical problem is a hardness of finding discrete logarithms in finite cyclic groups [17]. The Fig. 3 illustrates the basic Identity-Based Cryptography scheme:

3.5 Certificate Less-Public Key Cryptography (CL-PKC)

Certificate less cryptography is a variant of ID-based cryptography intended to prevent the key escrow problem. Ordinarily, keys are generated by a certificate authority or a key generation center (KGC) who is gives complete power and is implicitly trusted. To prevent a complete breakdown of the system in the case of a compromised KGC, the key generation process is split between the KGC and the user. The KGC first generates a key pair where the private key is now the partial private key of the system. The remainder of the key is a random value generated by the user and never revealed to anyone. All cryptographic operations by the user are performed by using a complete private key which involves both the KGC's partial key, and the user's random secret value. One disadvantage of this is that the identity information no longer forms the entire public key. To encrypt a message to another user, three pieces of information are needed (the other user's public key, identity, and the third party's public information), to decryption process a user just needs to use their private key. For tight security, a certificate less system has to prove its security against two types of adversaries. Type 1 Adversary- Refers to any third party who can fake the user's public keys, corresponding to the

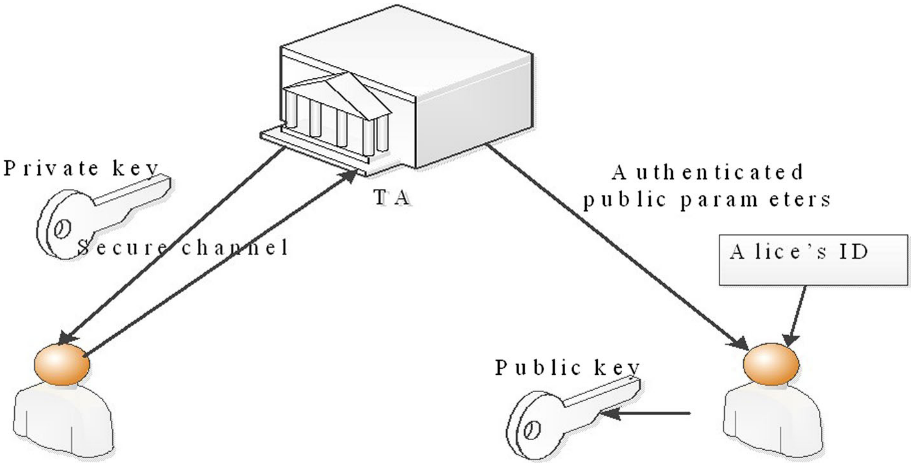


Fig. 3 Identity-base cryptography scheme

user's random secret value. Type 2 Adversary- Refers to a compromised or malicious KGC, who has access to the partial public and private keys of all users. The infrastructure needed to support CL-PKC is lightweight when compared to a traditional PKI. This is because, just as with ID-PKC, the need to manage certificates is completely eliminated. This immediately makes CL-PKC attractive for GSM, where the need to transmit and check certificates has been identified as a significant limitation. However, it should be pointed out that recently introduced signatures schemes enjoying very short signatures could be used to significantly decrease the size of certificates and create a lightweight PKI; in addition, CL-PKC signature scheme can also support true non-repudiation, because private keys remain in the sole possession of their legitimate owners. Certificate less cryptography (CL-PKC) was firstly presented by Al-Riyami and Paterson in their paper [18]. The work was highly influenced by the BF scheme [19] and as a result is an extension of the original IBE. CL-PKC eliminates the key escrow feature found in the Private Key Generator. Instead, the creation of the private key is split between a user and trusted third party called Key Generation Center (KGC). Consequently, user's public key is a pair composed of identity ID and public key PA. The key is no longer easily memorable as in original IBE but the trust level placed on third party is much lower. It looks like t functionality of CL-PKC is somewhere between traditional certified PKI and identity based cryptography. Flexibility is one of the most significant attributes of certificate less cryptography; in fact, it can be transformed into traditional PKI or IBE. Similarly to IBE, mathematical foundations of CL-PKE come from elliptic curves and hardness of finding discrete logarithms in finite groups.

Main features of CL-PKC includes the lack of key escrow property, no certificates to guarantee authenticity of public keys, the use of identities and existence of trusted third party which participates in key generation. To encrypt a message one needs public parameters, recipient's identity and public key. The use of identity in encryption prevents any other party from decrypting the content, even if one tries to forge the second part of the public key. Furthermore, the second part of public key (i.e. a point of elliptic curves) prevents KGC from deciphering the message.

In contrast to PKI, certificate less scheme does not require expensive infrastructure composed of different kind of authorities. Similarly to IBE, only Key Generation Center and

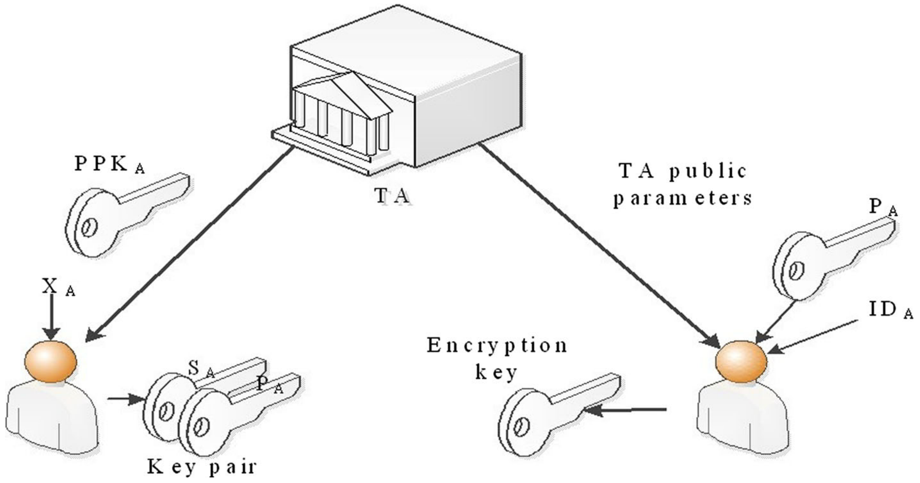


Fig. 4 Certificate less-public key cryptography scheme

Public Parameters Server are needed. The optimal choice is to place them per namespace, such as DNS zone. These two servers shall cope with all the traffic.

The Fig. 4 illustrates Certificate less Public Cryptography scheme:

Certificate Less Public Key Encryption (CL-PKE)

CL-PKE consists of seven algorithms as described in [20], and this scheme shown as following:

Setup.

First, let k ≡ a security parameter given to the Setup algorithm.

IG ≡ a BDH parameter generator with input k.

This algorithm runs as follows:

1. Run IG on input k to generate output $\{G_1, G_2$, and e$\}$ where G_1 and G_2 are groups of some prime order q and e: $G_1 \times G_1 \rightarrow G_2$ is a pairing.
2. Choose an arbitrary generator $P \epsilon G_1$.
3. Select a master-key s uniformly at random from Z_q^* and set $P_0 = sP$
4. Choose cryptographic hash functions $H_1 : \{0, 1\}^* \rightarrow G_1^*$ and $H_2 : G_2 \rightarrow \{0, 1\}^m$ m ≡ the bit-length of plaintexts.

The system parameters are params = $\{G_1, G_2$, e, m, P, P_0, H_1, and $H_2\}$.

The master key is $S \epsilon Z_q^*$

The message space is M= $\{0, 1\}^m$

The cipher text space is $C = G_1 \times \{0, 1\}^m$

Partial-Private-Key-Extract.

This algorithm takes as input an identifier $ID_A \epsilon \{0, 1\}^*$, and carries out the following steps to construct the partial private key for entity A with identifier ID_A:

1. Compute $Q_A = H_1(ID_A) \epsilon G_1^*$
2. Output the partial private key $D_A = s Q_A \epsilon G_1^*$

N.B. correctness of the Partial-Private-Key-Extract algorithm output by checking

$$e(D_A, P) = e(Q_A, P_0).$$

Set-Secret-Value.

This algorithm takes as inputs params and an entity A's identifier ID_A as inputs.

It selects $x_A \epsilon Z_q^*$ at random and outputs x_A as A's secret value.

Set-Private-Key.

This algorithm takes as inputs params, an entity A's partial private key D_A and A's secret value $x_A \epsilon Z_q^*$. It transforms partial private key D_A to private key S_A by computing

$$S_A = x_A D_A = x_A s Q_A \epsilon G_1^*$$

Set-Public-Key.

This algorithm takes params and entity A's secret value $x_A \epsilon Z_q^*$ as inputs and constructs A's public key as

$$P_A =< X_A, Y_A > \quad \text{where,} \quad X_A = x_A P \text{ and } Y_A = x_A P_0 = x_A s P$$

Encrypt .

To encrypt M for entity A with identifier $ID_A \epsilon \{0, 1\}^*$ and public key $P_A = (X_A, Y_A)$, perform the following steps:

1. Check that $(X_A, Y_A) \epsilon G_1$ and that the equality $e(X_A, P_0) = e(Y_A, P)$ holds. If not, output $\perp$ and abort encryption.
2. Compute $Q_A = H_1(ID_A) \epsilon G_1^*$
3. Choose a random value $r \epsilon Z_q^*$
4. Compute and output the cipher text: $C = < r P, M \oplus H_2(e(Q_A, Y_A)^r) >$

Decrypt .

Suppose C = {U, V}. To decrypt this cipher text using the private key S_A, compute and output:

$$M = V \oplus H_2(e(S_A, U))$$

Notice that if $\{U = r P, V\}$ is the encryption of M for entity A with public key $P_A = (X_A, Y_A)$, then we have:

$$\begin{aligned} V \oplus H_2(e(S_A, U)) &= V \oplus H_2(e(x_A s Q_A, rP)) \\ &= V \oplus H_2(e(Q_A, x_A s)^r) = V \oplus H_2(e(Q_A, Y_A)^r) = M \end{aligned}$$

Certificate Less-Public Key Signature (CL-PKS)

CL-PKS [19] consists of seven algorithms, the first five algorithms are same as in CL-PKE so we need here to discuss the two algorithms (sign and verify) as following:

Sign

This algorithm takes as inputs params, a message M to be signed and a private key S_A. It outputs a signature (σ), and it performs the following steps:

1. Choose random $a \epsilon Z_q^*$
2. Compute $r = e(aP, P) \epsilon G_2$
3. Set $v = H(M, r) \epsilon Z_q^*$
4. Compute $U = v S_A + a P \epsilon G_1$
5. Output

Verify

This algorithm takes as inputs parameter, a message M, identifier ID_A, public key P_A of an entity A, and σ as the signature to be verified, it outputs valid or invalid, and this algorithm performs the following steps:

1. Check that the equality $e(X_A, P0) = e(Y_A, P)$ holds. If not, output $\perp$ and abort verification.
2. Compute $r = e(U, P) . e(Q_A, -Y_A)^v$
3. Check if $v = H(M, r)$ holds. If it does, output valid, otherwise output invalid.

4 System Designing

The infrastructure needed to support CL-PKC is lightweight unlike a traditional PKI because it does not need certificate management. This feature makes CL-PKC more attractive for mobile communication applications because of the following features: It provides low bandwidth, Low power situations, Low handshaking negotiations between subscribers and network operator, Low computational cost.

In Fig. 5 proposed model makes assumes two users that wish to authenticate one another third party that is present to provide the authentication parameters. The two users being considered are the Mobile Station MS,(HLR\AuC), third party being the Authentication Center (AuC) in the core and the AuC as equal and as being the same component. The AuC is specific database for security purpose where the need to transmit and check certificates has been identified as a significant limitation and CL-PKC introduced signatures schemes enjoying very short signatures could be used to significantly decrease the size of certificates and create a lightweight PKI. They provided true non-repudiation because private keys remain in the possession of subscribers, i.e. SIM card and AuC, furthermore the revocation of keys in CL-PKC systems can be handled. This research took a particular part of CL-PKS; only the authentication schemes in the basic CL-PKC, with the possibility of achieving the encryption scheme CL-PKE. A little change was necessary by making the MSC\VLR the component which is responsible for the Encryption\Decryption scheme and let AuC be set as a third party. The proposed model needs some modifications for suitability in the basic GSM system and in the basic CL-PKC to be ready to be implemented in the new scheme of CL-PKC. The most important thing that the key escrow problem is returned to CL-PKC, because a third

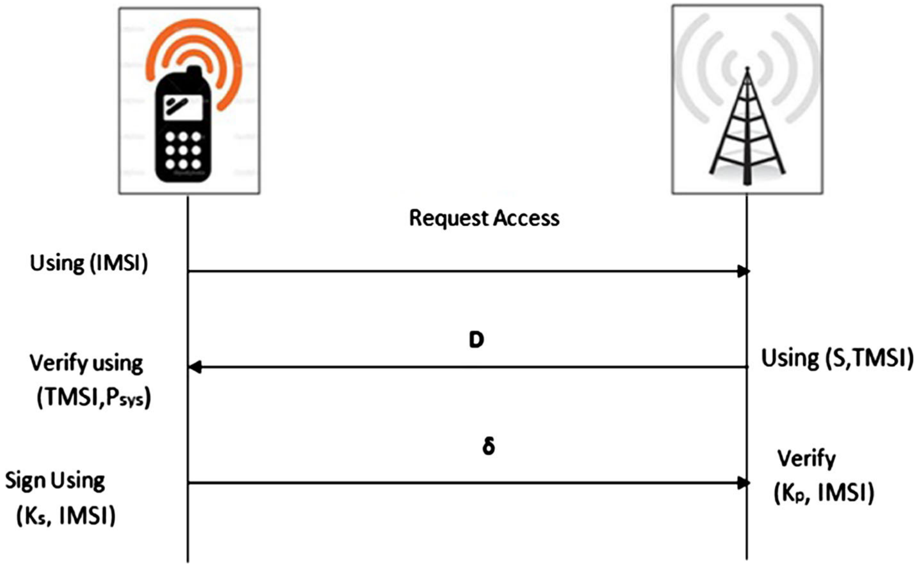


Fig. 5 The main mutual authentication process

party (AuC) is trusted, no matter what the AuC possess the private keys for all subscribers, we need the partial private key to provide mutual authentication and we cannot send the private key via unsecure air-interface. The following ten assumptions and requirements will make CL-PKC compatible with standard GSM as much as possible:

1. Assume that the weaknesses in the GSM infrastructure is in the Air-interface (from MS to BTS) and the other interfaces (from BSC to MSC, from MSC to HLR) are almost secure.
2. We need to make the Authentication Center (AuC) and the Mobile Station (MS) more flexible store to achieve CL-PKS parameter algorithms, (Software, Memory, Processors, etc.)
3. Using a TMSI number as a unique identifier for each subscriber, which is randomly assigned by the (VLR) to every mobile in the area when it is switched on. The number is local to a specific area and it has to be updated each time the mobile moves to a new geographical area that it is made useful for the current design.
4. Using the IMSI number as a message (M) which stored on both SIM card and AuC. We need it in the signature and verification algorithms, because it is a permanent number.
5. Assume that all subscribers have a unique key (K_0), and that it illustrates the secret value in the basic CL-PKC, also that the AuC has copies for all subscriber keys stored together with their identifiers IMSI, and it is only stored in the SIM card and at the (AuC), and should never be transmitted across the network on any link.
6. Suppose that the KGC in CL-PKC illustrated by AuC in the GSM architecture to compute the partial private key for each subscriber and generate the system parameters,
7. Assume that the AuC is a trusted party for all subscribers, because here the AuC can compute all the subscriber's private keys by using (K_0) which is stored.
8. Suppose that the (SIM card) includes all the necessary information (IMSI, system parameters $P_{sys},\ K_0$).
9. Suppose that the (AuC) includes all the necessary information (IMSI, system parameters $P_{sys},\ K_0$, and master key S).
10. The (MSC/VLR) is responsible for the assignment of TMSI for all subscribers in each locality, and checks IMSI from HLR/AuC, then sends Q=H(TMSI) to the AuC on a secured link.

These assumptions will ensure that the authentication process is computable with the GSM system it makes the system more secure and more flexible with the proposed design as shown in the following.

Using TMSI as an identifier for the subscribers in our proposed model makes the private key more secure because it makes the private key changeable when the subscriber moves from one location area to another. This assumption gives the scheme good flexibility by using IMSI as a message which needs to be signed also provides another level of security to the system by checking IMSI in the call setup process by default, and again in the verification algorithm. The IMSI number stored in both MS and AuC, and the MSC knows the TMSI number from the VLR and sends it to the AuC as a hash value to compute the partial private key.

4.1 Design of CL-PKS over GSM System

The main procedures for making a call over a mobile communications in general are:

1. Call Setup
2. Data Transfer
3. Call Clearing

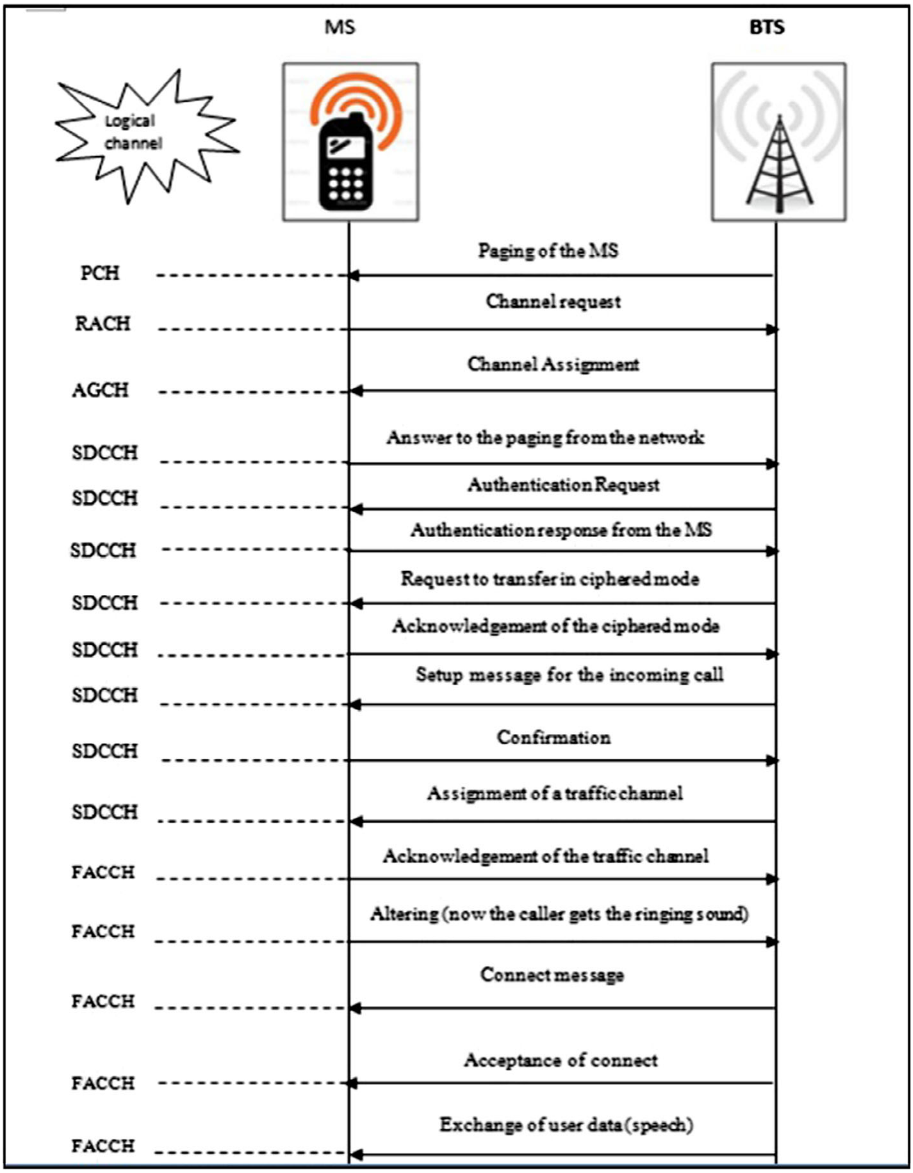

MS
BTS
Logical channel
PCH
Paging of the MS
RACH
Channel request
AGCH
Channel Assignment
SDCCH
Answer to the paging from the network
SDCCH
Authentication Request
SDCCH
Authentication response from the MS
SDCCH
Request to transfer in ciphered mode
SDCCH
Acknowledgement of the ciphered mode
SDCCH
Setup message for the incoming call
SDCCH
Confirmation
SDCCH
Assignment of a traffic channel
FACCH
Acknowledgement of the traffic channel
FACCH
Altering (now the caller gets the ringing sound)
FACCH
Connect message
FACCH
Acceptance of connect
FACCH
Exchange of user data (speech)

Fig. 6 Call setup model

A GSM system implements two processes before the call setup is synchronized and location updated.

The design proposed for CL-PKS related to the first procedure (Call Setup), has two handshaking processes to authenticate the subscriber by the network operator. The proposed scheme also requires two handshaking processes to implement CL-PKS; this proposed design provides a mutual authentication by using (TMSI and K_0) for each subscriber to ensure that this is its own network operator, and by the operator to ensure that this subscriber belongs to the network. In Fig. 6 the main description of this idea is to let the MSC assign the subscriber's

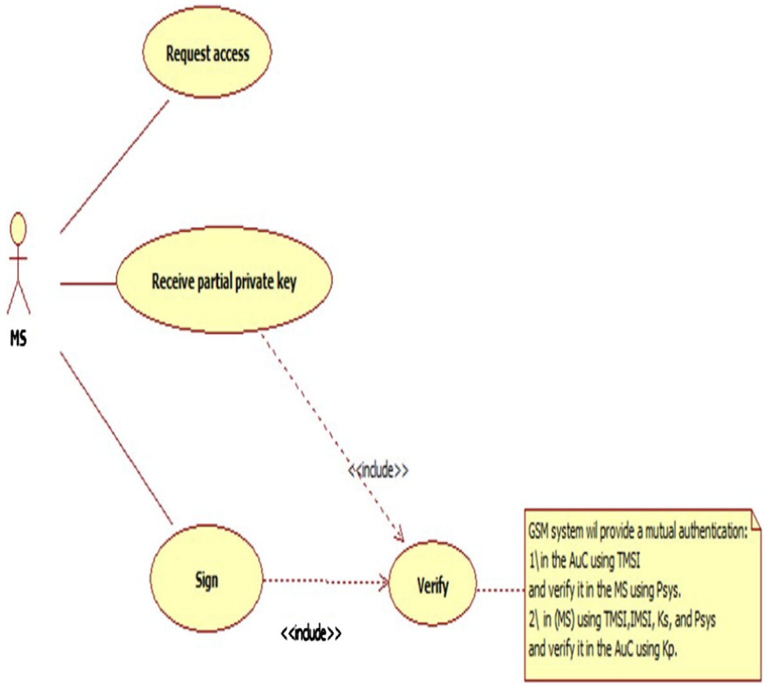

Request access
Receive partial private key
MS
<<include>>
Sign
Verify
<<include>>
GSM system wil provide a mutual authentication:
1\in the AuC using TMSI
and verify it in the MS using Psys.
2\ in (MS) using TMSI,IMSI, Ks, and Psys
and verify it in the AuC using Kp.

Fig. 7 Main mutual authentication model

TMSI and then hash it and send the hashed assignment to the AuC, given that in the basic GSM system the TMSI never transmits to the AuC for security purposes. When this *TMSI* is allocated in advance by MSC\VLR and sends it to a specific MS in the call setup process or in the paging process in case the MS is called from the network. Note that there is no secret value here as in the basic CL-PKC, and it is replaced *by K0,* which is stored as a secure key in both the MS and AuC, there are four algorithms achieved by the HLR\AuC framework, Setup (P_{sys}), Partial Private Key Extract (D), Set Public key (K_{p}), and Verify (σ).

On the mobile station side there are three algorithms achieved by the MS\SIM, Verify (D), Set Private Key (K_{s}), and Sign IMSI (σ).

The main procedure for call setup in the GSM system is

The system was designed using the UML notations by: use case, sequence and activity which are offered by UML and the Rational Rose Tool. Use cases can be described by text descriptions which explain how the use case, describes the inter-action between actors to achieve a goal of observable value. The following use case description in Fig. 7 shows the analysis of the main mutual authentication process system in use cases, which is the interaction between the Mobile station (MS) and Network operators (Table 1).

In our proposed design we suppose the KGC is a trusted party and represented by the AuC in HLR, firstly when a MS requests access to the network, the MSC/VLR will normally require the MS to authenticate. The MSC will forward the IMSI to the HLR and request authentication parameters, and then when the HLR receives the IMSI and the authentication request, it first checks its database to make sure the IMSI is valid and belongs to the network. Once it has accomplished this, it will forward the IMSI and authentication request to the Authentication Center (AuC). See the Fig. 8.

Table 1 Use case description for the main mutual authentication

System analysis use case (main mutual authentication)		
Use case name:	Main mutual authentication	
Use case ID:	UC001	
Lse case type:	System analysis	
Version number:	1.0	
Primary actor(s):	MS	
Participating actor(s):	None	
Description:	Request access from network operator	
Precondition:	Each MS, has unique permanent $IMSI$, and $TMSI$	
Trigger:	This use cass initiated when the MS send the service request	
Main scenario:	Actor action	System response
	Step 1: MS send request access to network operators include *IMSI*	Step 2: System will generate partial private key D using AuC's master key, then send it to MS
	Step 3: MS will make verify using ($TMSI$, P_{sys})	
	Step 4: MS sign ($IMSI$)	Step 5: System will mike verify using ($\mathrm{k_p}$)
Alternate scenarios:		Step 3: system will check *IMSI* is valid and belongs to the network. Otherwise system will deny the request
Exceptions:	During the authentication process if there is connection loss, system will not make authentication and MS will be repeat the request	
Conclusion:	MS and N.W. operators are authenticated	
Open issues:	None	

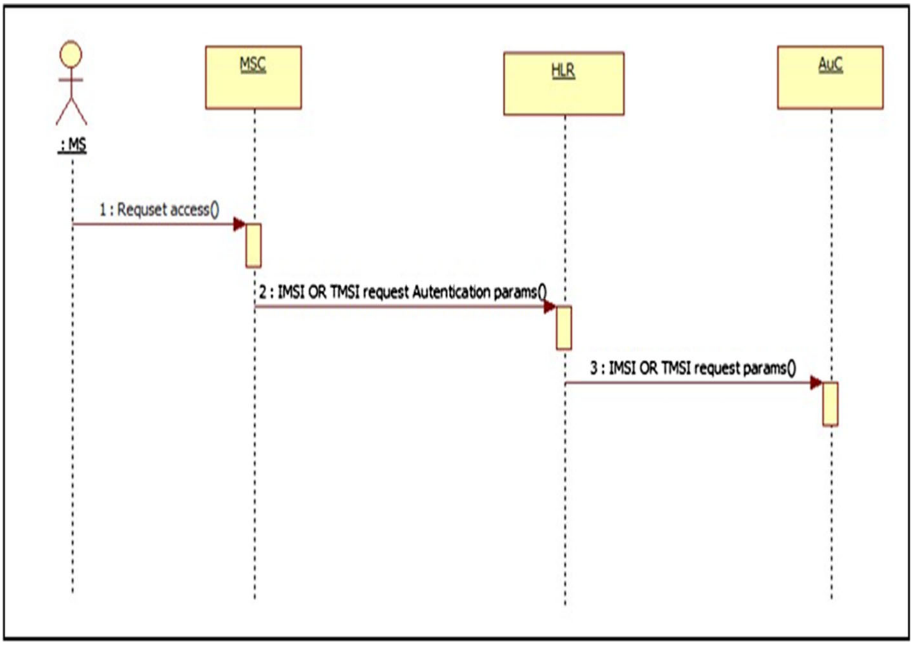

: MS
MSC
HLR
AuC
1 : Requset access()
2 : IMSI OR TMSI request Autentication params()
3 : IMSI OR TMSI request params()

Fig. 8 Access request and IMSI check model

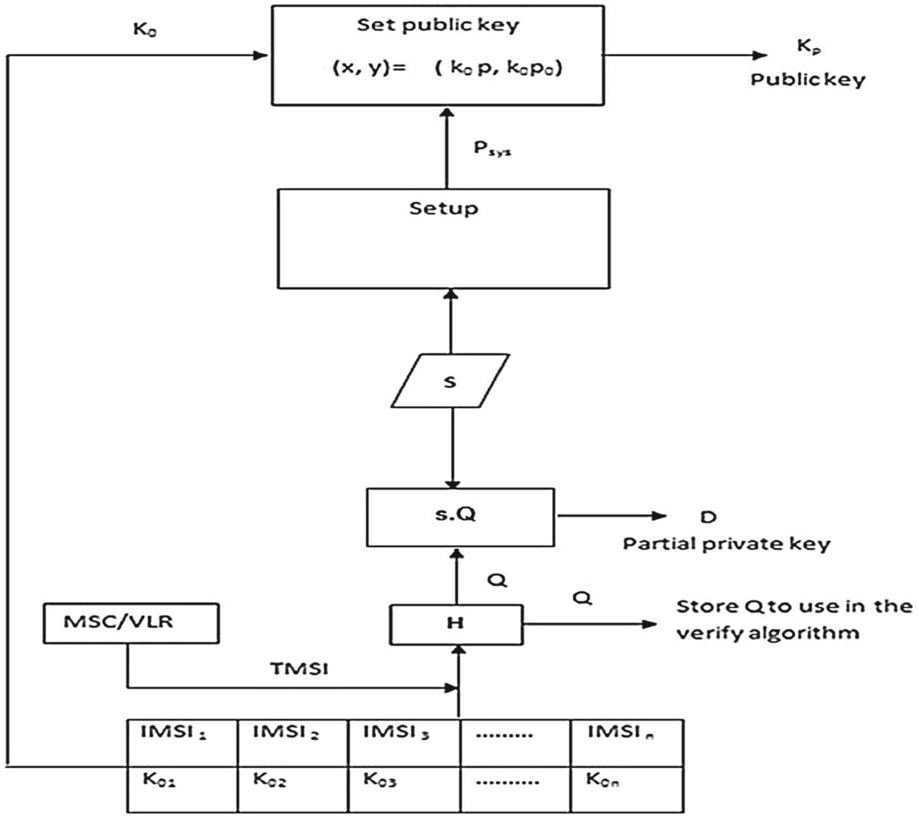


Fig. 9 Public and partial private keys rithms AuC

The AuC can make the main processes in CL-PKC (Setup, Partial Private Key extract, and set public key) generates (P_{sys}, D, and K_p), these processes depends on the Authentication Center master key (S), and the Secret key (K_0).The output System parameters (P_{sys}) with the secret key K_0 are used to generate the public key K_p, and are then used to verify the algorithm. The partial private key (D) sends to MS via MSC when authentication is requested, The AuC runs the next algorithm (Set public key) immediately and stores it for a specific time period to verify the signature (σ), when it is received from the Mobile station (MS), these processes are described in Fig. 9.

The proposed model provides a mutual authentication, when the mobile station receives a partial private key (D), it will start to authenticate the network operator (see Fig. 9) by using the following equation:

$e\ (D,P) = e(Q,P_0)$
Where, $P_0 = SP$
$D = SQ$
So we get, $e\ (D, P) = e\ (Q, SP)$
By using bilinear property for pairing:
We get, $e\ (D, P) = e(SQ, P) = e(D, P)$

The master key (S) is not contained in the SIM card (Subscriber side) but the algorithm takes the system parameters (P, P_0) and its own TMSI which is assigned in the paging

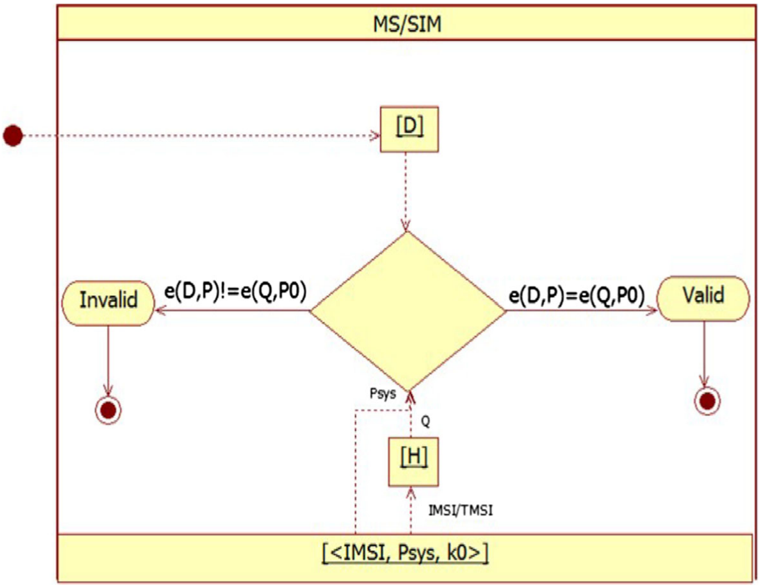


Fig. 10 Authentication process in the MS

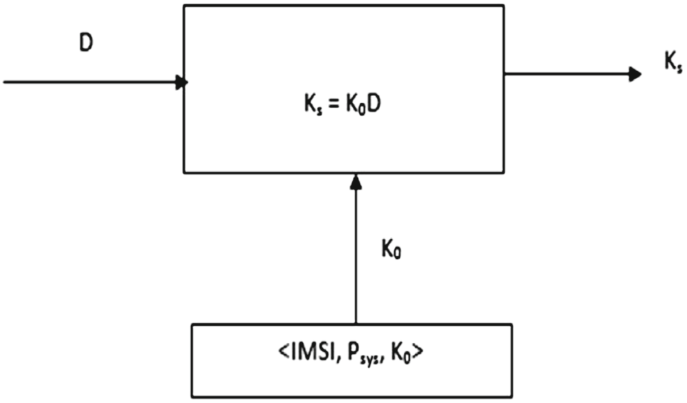


Fig. 11 Set private key (Ks)

process (call setup). It inputs the TMSI in the hash function to get (Q), and compares the above equation, so that the partial private key can be confirmed to be from the AuC; then the AuC is authenticated, this process is described in Fig. 10.

After the subscriber authenticates the network and outputs (Valid), the algorithm runs the next step (set private key-Ks) by using the K_0 which is stored in the SIM card as following Fig. 11:

Sign.

After MS generates the temporary private key (Ks), it depends on the current TMSI; the next algorithm is executed to sign a message, this algorithm runs in the MS. It takes as inputs (P_{sys}), a message (IMSI) to be signed and a private key (Ks) which was generated in the

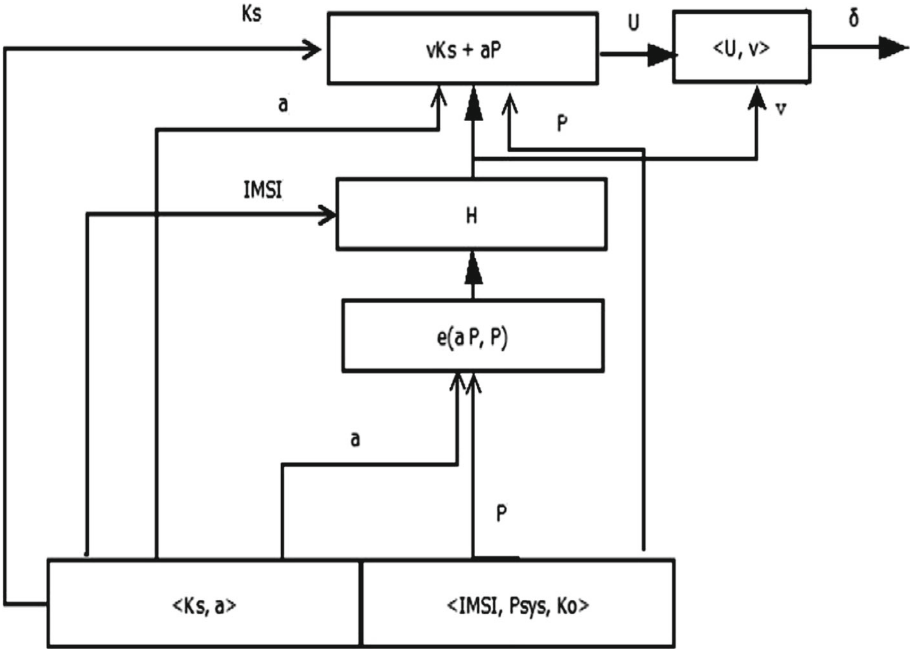


Fig. 12 Signature algorithms in the MS

previous algorithm, and it outputs (σ). These are then sent it to the AuC as a signature; the Sign algorithm performs the following steps (Fig. 12):

1. Choose random a ϵ Z_q^*
2. Compute r = e (aP,P) ϵ G_2
3. Set v = H (IMSI, r) ϵ Z*q
4. Compute U $=$ v $K_s +$ a P ϵ G_1
5. Output as the signature σ $=<$ U, v $>$

Verify.

This algorithm runs at the AuC, and it takes as inputs (P_{sys}), a message (*IMSI*), the identifier (*TMSI*) and public key (K_p) of an specific subscriber, and (σ) as the signature to be verified, it outputs (Valid) or (Invalid), the Verify algorithm performs the Fig. 13.

1. Compute $r = e(U, P).e\,(Q, -Y)^v$
2. Check if $v = H(IMSI, r)$ holds.

If it does, output (Valid), otherwise output (Invalid).

If one of the authentication parameters entered wrong, then this algorithm outputs Invalid, and these parameters are:

1. System parameters P_{sys}
2. Private Key K_s by change TMSI or K_0
3. Public key K_p by change P_{sys} or K_0
4. IMSI

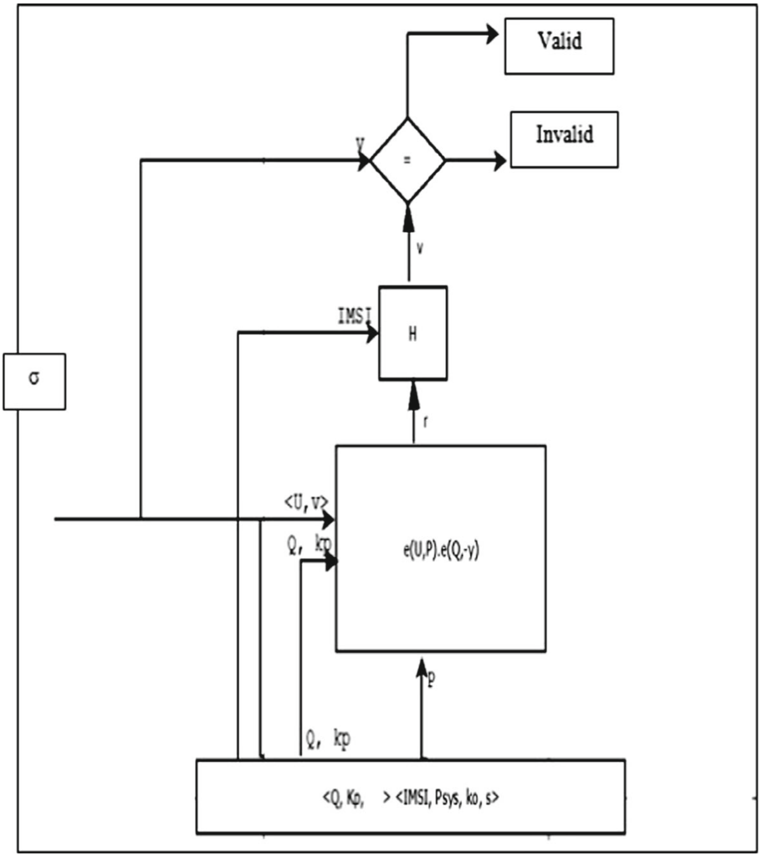


Fig. 13 Verify algorithms AuC

5 Experiment and Evaluation

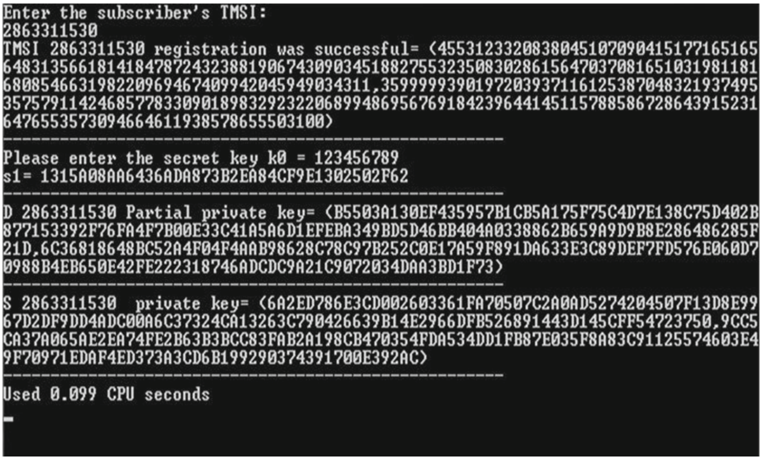
In this section we showed the implementation of the CL-PKS method with a few modifications, generally this implementation consists of four programs that implement each of the steps in our scheme. The main aim of this implementation is to create applications according to the available requirements; it focuses at this stage, on specific tools such as libraries in the internet, which allow quick production of software of high quality. Certificate less cryptography is implemented in conformity with the object-oriented paradigm, by using C++ language; most benefits are derived from the MIRACLE library on the internet. The implementation took less than 1 sec for the setup, partial private key, full private key, and public key algorithms, when it achieved on PC with specification (2CPUs: 2.4 GHz, RAM = 4 GB), and the time consumption is good compared with the call setup process in the GSM system, and this short period because this approach used a low computation cost by using the basic CL-PKC.

This implementation divided into four parts as follows:

5.1 Setup

This program generates the system parameters (P_{sys}), and stores them in the MS and the AuC, it also generates the public key (K_p) using the AuC Master key and the secret key

K0 that is shared in advance, this public key is stored at the Auc for later use in the verify algorithm.

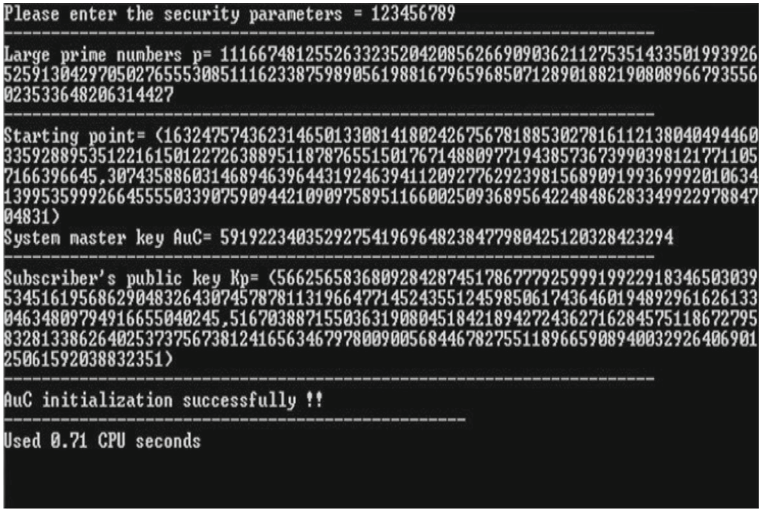


```
Enter the subscriber's TMSI:
2863311530
TMSI 2863311530 registration was successful= (455312332083804510709041517716516
6483135661814184787243238819067430903451882755323508302861564703708165103198118
1680854663198220969467409942045949034311,35999993901972039371161253870483219374
953575791142468577833090189832923220689948695676918423964414511578858672864391523
16476553573094664611938578655503100)
------------------------------------------------------
Please enter the secret key k0 = 123456789
s1= 1315A08AA6436ADA873B2EA84CF9E1302502F62
------------------------------------------------------
D 2863311530 Partial private key= (B5503A130EF435957B1CB5A175F75C4D7E138C75D402B
877153392F76FA4F7B00E33C41A5A6D1EFEBA349BD5D46BB404A0338862B659A9D9B8E286486285F
21D,6C36818648BC52A4F04F4AAB98628C78C97B252C0E17A59F891DA633E3C89DEF7FD576E060D7
0988B4EB650E42FE222318746ADCDC9A21C9072034DAA3BD1F73)
------------------------------------------------------
S 2863311530  private key= (6A2ED786E3CD002603361FA70507C2A0AD5274204507F13D8E99
67D2DF9DD4ADC00A6C37324CA13263C790426639B14E2966DFB526891443D145CFF54723750,9CC5
CA37A065AE2EA74FE2B63B3BCC83FAB2A198CB470354FDA534DD1FB87E035F8A83C91125574603E4
9F70971EDAF4ED373A3CD6B1992903743917 00E392AC)
------------------------------------------------------
Used 0.099 CPU seconds
_
```

The security parameters are entered into the software and the system parameters calculated; the public key (Kp) is needed it in the verify algorithm.

5.2 Partial Private Key and Full Private Key Extraction

This program extracts a partial private key from the proffered identity string (TMSI), and stores it in file D, it then calculates the private key simply, by multiplying the partial private key by the secret key which stored in advance. This class takes the subscriber's TMSI and the secret key K0 as input and outputs the partial private key and the private key which stored for the signature algorithm in the mobile station. We note that both the partial private key and the private key directly depend on the TMSI number. The TMSI number it depends on the VLR, which is assigned to each subscriber, when he\she moves from one coverage area to another; so the strength of the keys here are related to the TMSI changing.

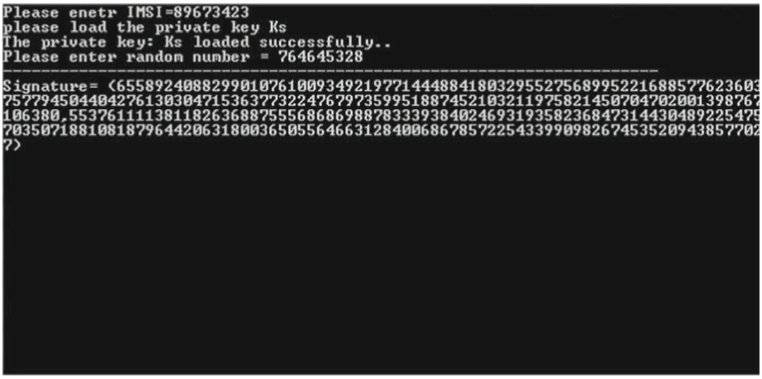


```
Please enter the security parameters = 123456789
------------------------------------------------------------------
Large prime numbers p= 1116674812552633235204208562669090362112753514335019939265
25913042970502765553085111623387598905619881679659685071289018821908089667935560
23533648206314427
------------------------------------------------------------------
Starting point= (16324757436231465013308141802426756781885302781611213804049446
0335928895351221615012272638895118787655150176714880977194385736739903981217711
057166396645,30743588603146894639644319246394112092776292398156890919936999201063
41399535999266455550339075909442109097589511660025093689564224848628334992297884
704831)
System master key AuC= 5919223403529275419696482384779804251203284 23294
------------------------------------------------------------------
Subscriber's public key Kp= (566256583680928428745178677792599919922918346503039
5345161956862904832643074578781131966477145243551245985061743646019489296162613
30463480979491665504024 5,516703887155036319080451842189427243627162845751186727
95832813386264025373756738124165634679780090056844678275511896659089400329264069
0125061592038832351)
------------------------------------------------------------------
AuC initialization successfully !!
------------------------------------------------
Used 0.71 CPU seconds
```

5.3 Signature Algorithms in the MS

This program accepts the subscriber's TMSI (which is his private key), and signs a message which is his IMSI num-ber.

After the MS generates the temporary private key (Ks), which is dependent on the current TMSI, then the next step is the signature algorithm. This algorithm runs in the MS, and it takes as inputs (Psys), a message (IMSI) to be signed and a private key (Ks) which is generated in the previous algorithm, and it outputs (σ), it then sends this output to the AuC as a signature; the Sign algorithm results are shown in the following screenshot:



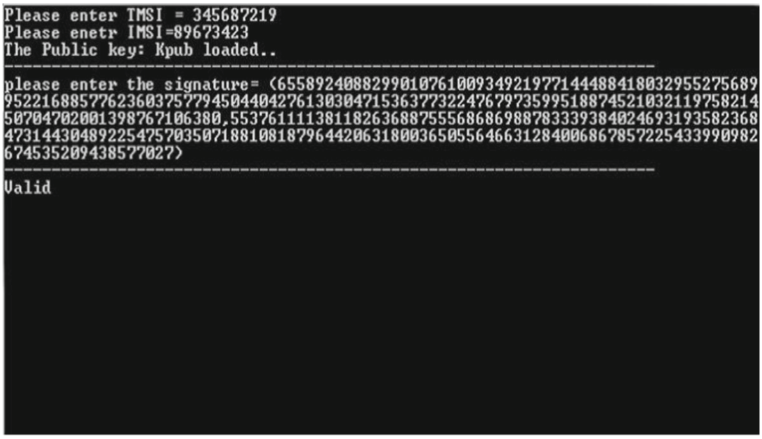
```
Please enetr IMSI=89673423
please load the private key Ks
The private key: Ks loaded successfully..
Please enter random number = 764645328
------------------------------------------------------------------------
Signature= <65589240882990107610093492197714448841803295527568995221688577623603
75779450440427613030471536377322476797359951887452103211975821450704702001398767
106380,55376111138118263688755568686988783339384024693193582368473144304892254757
70350718810818796442063180036505564663128400686785722543399098267453520943857702
7>
```

So the trick of this algorithm exists in using the (r) parameter by picking a random number (a), and then generates the first part of the signature (v), hence it is used to compute the other part (U), as described in Sect. 5.2. After all this the MS sends the signature to the AuC to be verified, and then the authentication process is complete, and is transferred to ciphering mode after the verification process.

5.4 Verify Algorithm at the AuC

The verify algorithm inputs the system parameters (Psys), a message (IMSI), the identifier (TMSI) and public key (Kp) of a specific subscriber, and (U,v) being the signature to be verified, it outputs (Valid) or (Invalid), the result of the verify algorithm as shown in the following screenshot:

The basic idea trying to be conveyed is how to calculate the parameter (r), without knowing the random number which is chosen by the sender MS.It must be noted that the verification process depends on the subscriber's public key parameter (r), and the message IMSI, so the AuC can get the specific IMSI from the HLR and hash it with r; which is calculated in advance. (v) is then computed, and compared with the other (v) that is sent from the mobile station MS.



```
Please enter TMSI = 345687219
Please enetr IMSI=89673423
The Public key: Kpub loaded..
----------------------------------------------------------
please enter the signature= (655892408829901076100934921977144488418032955275689
95221688577623603757794504404276130304715363773224767973599518874521032119758214
50704702001398767106380,55376111138118263688755568686988783339384024693193582368
47314430489225475703507188108187964420631800365055646631284006867857225433990982
674535209438577027)
----------------------------------------------------------
Valid
```

6 Conclusion

GSM comes close to fulfilling the requirements for a personal communication system, close enough that it is being used as a basis for the next generation of mobile communication technology in the world such as UMTS, CDMA and LTE. In wireless services, secure and secret communication is desirable; this is in the interest of both, the subscribers and the service providers. These parties would never want their resources and services to be used by unauthorized users. In this research a technique to provide a mutual authentication for the GSM system, with a little handshaking procedure was demonstrated similar to the A3 algorithm. Certificate-less cryptography can supply one of the most flexible infrastructures for public key cryptography. It combines the best aspects of both traditional public-key infrastructure and identity-based encryption. The IBE, certificate-less crypto-graph can be used as the underlying mechanism for transparent email/sms (short message service) encryption, in this research this system is made as suitable as possible for the GSM system. The design and implementation of CL-PKC to provide both of the encryption and mutual authentication, there are many new schemes for the CL-PKC to give a high security but hard to implement them over GSM because they needs a lot of security requirements. In the future, ensure that this approach is integrated with the GSM system and compatible it will be easy to check the system performance. We can make the private key changing by let the identity fixed and change the master key, i.e. let the identity IMSI number and the master key be the TMSI number. This research provided for me a rare opportunity of combining the most interesting fields of computer science; information security, and communication engineering, mobile communications, which altogether create a bridge between theory and practice. Nevertheless, the research is not at all closed and it will be extended further various other security schemes.

References

1. Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing* (pp. 227–236). Berlin: Springer.
2. Tarjan, R. E., & Yannakakis, M. (1984). Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM Journal on Computing*, *13*(3), 566–579.

3. Goswami, S., Laha, S., Chakraborty, S., & Dhar, A. (2012). Enhancement of GSM Security using elliptic curve cryptography algorithm. In *Intelligent systems, modelling and simulation (ISMS), 2012 third international conference on* (pp. 639–644). IEEE.
4. Driessen, B., Hund, R., Willems, C., Paar, C., & Holz, T. (2013). An experimental security analysis of two satphone standards. *ACM Transactions on Information and System Security (TISSEC)*, *16*(3), 10.
5. Forsgren, H., Grahn, K., Karvi, T., & Pulkkis, G. (2010). Security and trust of public key cryptography options for HIP. In *Computer and information technology (CIT), 2010 IEEE 10th international conference on* (pp. 1079–1084). IEEE.
6. Seo, S. H., Won, J., & Bertino, E. (2014). POSTER: A pairing-free certificateless hybrid sign-cryption scheme for advanced metering infrastructures. In *Proceedings of the 4th ACM conference on data and application security and privacy* (pp. 143–146). ACM.
7. Islam, S. H., & Biswas, G. (2012). Certificateless strong designated verifier multisignature scheme using bilinear pairings. In *Proceedings of the international conference on advances in computing, communications and informatics* (pp. 540–546). ACM.
8. Yin, X., & Han, J. (2003). CPAR: Classification based on predictive association rules, *Proceedings 2003 SIAM International Conference on Data Mining (SDM'03)*, San Francisco CA, pp. 331–335.
9. Liu, J. K., Au, M. H., & Susilo, W. (2007). Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model. In *Proceedings of the 2nd ACM symposium on information, computer and communications security*, (pp. 273–283). ACM.
10. Cho, J.-H., Chan, K. S., & Chen, I.-R. (2013). Composite trust-based public key management in mobile ad hoc networks. In *Proceedings of the 28th annual ACM symposium on applied computing*, (pp. 1949–1956). ACM.
11. Meyer, U., & Wetzel, S. (2004). A man-in-the-middle attack on UMTS. In *Proceedings of the 3rd ACM workshop on wireless security* (pp. 90–97). ACM.
12. Teng, J., & Wu, C. (2012). A provable authenticated certificateless group key agreement with constant rounds. *Journal of Communications and Networks*, *14*(1), 104–110.
13. Hableel, E., Byon, Y.-J., & Beak, J. (2013). Public key infrastructure for UAE: A case study. In *Proceedings of the 6th international conference on security of information and networks* (pp. 336–340). ACM.
14. Ray, S., & Biswas, G. (2012). An ECC based public key infrastructure usable for mobile applications. In *Proceedings of the second international conference on computational science, engineering and information technology* (pp. 562–568). ACM.
15. Praveen, I., & Sethumadhavan, M. (2012). A more efficient and faster pairing computation with cryptographic security. In *Proceedings of the first international conference on security of internet of things* (pp. 145–149). ACM.
16. Nicanfar, H., TalebiFard, P., Hosseininezhad, S., Leung, V., & Damm, M. (2013). Security and privacy of electric vehicles in the smart grid context: problem and solution. In *Proceedings of the third ACM international symposium on design and analysis of intelligent vehicular networks and applications* (pp. 45–54). ACM.
17. Otto, M. (2012). Highly acyclic groups, hypergraph covers, and the guarded fragment. *Journal of the ACM (JACM)*, *59*(1), 5.
18. Al-Riyami, S., & Paterson, K. (2003) Certificateless public key cryptography. In *Advances in Cryptology-Asiacrypt'2003*, Lecture Notes in Computer Science (vol. 2894, pp. 452–473,Springer).
19. Hanaoka, G. (2013). On the properties of public key encryption from group signatures. In *Proceedings of the first ACM workshop on Asia public-key cryptography* (pp. 1–2). ACM.
20. Li, J., Huang, X., Mu, Y., Susilo, W., & Wu, Q. (2007). Certificate-based signature: Security model and efficient construction. In *Public key infrastructure* (pp. 110–125). Springer.

Imran Memon B.S. Electronics 2008 from IICT University of Sindh Jamshoro Sindh Pakistan. M.E. Computer Engineering from University of Electronic Science and Technology Chengdu Sichuan China. I am doing Ph.D. from college of computer science and technology Zhejiang university. Nowadays, I got Academic Achievement Award 2011–2012 from UESTC China and also got Excellent Performance Award 2011–2012 from UESTC China, published more than 20 international conference papers and 10 journal papers and reviewer 4 science citation index journals, 2 EI index and journals many international conferences. Current research interests; Artificial intelligence system, Network security, Embedded system, Information security, Peer to Peer networks.

Mohammed Ramadan Mohammed Bachelor's degree, Electrical, Electronics and Communications Engineering from Sudan 2007., MS (Information Security) from University of Electronic Science and Technology of China 2013. Holder of three Professional Certifications: CCNA, CCNP and CISA. Currently at University of Electronic Science and Technology of China-UESTC pursuing a Ph.D degree in Computer Science (Information Security). Published more than 3 papers in both International Journals and Conferences, indexed by EI. Current Research interest: Network Security and Privacy, Location Based Services Privacy, Cloud Based Systems, Reputation Systems, Security Protocols.

Rizwan Akhtar received his B.E.Computer Engineering Degree from the Department of Electrical Engineering at the Comsats University of Information Technology, Islamabad, Pakistan in 2006 and his M.E. Telecom Engineering degree from the Department of Electronics Engineering at University of Engineering Technology, Peshawar, Pakistan in 2010. Currently he is a Ph.D major in Communication engineering student in school of Electronic Engineering at University of Electronic Science and Technology of China, Chengdu City, China. His research interests are mainly in the wireless and mobile communication, Mobile Cloud Computing, Mobile Social Networks and Security in Wireless Ad hoc Networks. He is the reviewer of wireless personal communication journal.

Hina Memon BS Computer Science from Institute of Mathematics and Computer Science at University of Sindh Jamshoro Pakistan. She published more than 4 papers in both International Journals and Conferences, indexed by EI. Current Research interest: information Security and Privacy, Cloud Based Systems, Data mining and Computer vision.

Muhammad Hammad Memon He was born in Hyderabad Sindh, Pakistan, in 1989. He has done bachelor degree in commerce from University of Sindh Jamshoro, in 2009, and he has MS Computer Engineering in the field of Cloud Computing at University of Electronic Science and Technology of China (UESTC), in 2014. Current research interests; Artificial intelligence system, Network security, Embedded system, Information security ,Peer to Peer networks & Cloud Computing.

Riaz Ahmed Shaikh I have done M.Sc. Computer science from Shah Abdul Latif University, Khairpur, Sindh, Pakistan. I am working as Lecturer since August 2008 in the department of Computer Science, Shah Abdul Latif University, Khairpur, Sindh, Pakistan. Now, I am doing PhD in Computer Science and Engineering, University of Electronics Science & Technology, Chengdu, Sichuan, China. Current research interests; Artificial intelligence system, Network security, Embedded system, Information security ,Image segmentation, Image processing and computer vision.