Subgroup Operations in Identity Based Encryption Using Weil Pairing for Decentralized Networks

N. Chaitanya Kumar, Abdul Basit, Priyadarshi Singh, and V. Ch. Venkaiah
(Corresponding author: Abdul Basit)

School of Computer and Information Sciences, University of Hyderabad
Hyderabad-500046, India
(Email: abdulmcajh@gmail.com)
(Received June 10, 2018; Revised and Accepted Aug. 18, 2018; First Online Jan. 14, 2019)

Abstract

One of the drawbacks of the conventional public key systems is that the sender must know the public key of the recipient in advance for the key setup and retrieval. This problem can be solved in Identity Based Encryption (IBE) by taking some identifier string (*e.g.* an e-mail or phone number, *etc.*) as the public key. When a user wants to send a message then he only has to know this identifier string. The receiver requests the private key from a Trusted Third Party called PKG (Private Key Generator) to decrypt the message. The job of the PKG can be decentralized using the Shamir secret sharing scheme. The Weil Pairing on the elliptic curve is suitable to implement IBE, as it is based on bilinear maps between groups. In this paper, we propose a scheme that allows threshold decryption involving a subgroup of participants of the network.

Keywords: Identity Based Encryption; Subgroup Operations; Weil Pairing

1 Introduction

Identity Based Encryption (IBE) will allow the sender to use the receiver's identity in order to encrypt the message instead of using his public key. The usage of identity instead of public key has wide range of applications. The identity based encryption system uses an arbitrary string as an identity. The identity based encryption system is first developed by Shamir in 1984 [20] to simplify the management of certificates in an e-mail system. For example, when A wants to send a mail to B at B123@company.com, A encrypts the message simply by using B123@company.com. With this there is no need for A to obtain public key certificate of B. When B receives the mail then B contacts Private Key Generator (PKG) a third party organization and obtains the private key by authenticating himself. Finally, B can read the mail which was sent by A. Weil pairing is a mapping of two computational Diffie-Hellman groups where one group being hard. Initially Weil pairing was used to attack elliptic curve systems [17]. Later, Joux [11] designed a protocol using one round diffie-hellman key exchange among three parties and proved that weil pairing can also be used for good. Sakai *et al.* [19] also used weil pairing for the exchange of keys. Operations performed among the sub group of users belonging to a network and how they deal when a new user wants to be part of the network is known as Subgroup operations. Our proposed scheme demonstrates a protocol for subgroup operations and also decentralizes the job of PKG. The advantage of PKG being decentralized is that the communication becomes secure, more reliable when compared to existing systems. It also allows the new users to have the same abilities as that of the initial users and each user has their share for the remaining life of the network.

2 Preliminaries

2.1 Shamir Secret Sharing

The secret sharing mechanism shares the secret s among a group of participants $P = \{p_1, p_2, \cdots, p_n\}$ of n parties by using a special figure called dealer. The dealer sends privately the share of a secret to each party. Reconstruction process is adopted by the authorized subsets to extract the secret s from the given shares. The group of such authorized subsets are called as access structure. Shamir secret sharing scheme [21] uses the Lagrange's interpolation polynomial to implement (t,n) access structure where t is the threshold value and n is the no.of participants. For example let us consider n participants, s is the secret, t is the threshold and the finite field is denoted by F_p. Shamir secret sharing scheme has two phases namely: Distribution and reconstruction [2]. In the construction phases shares are distributed to the users and in the reconstruction phase the users compute the secret from their shares.

2.2 Elliptic Curve Cryptography

In cryptography, elliptic curve is defined over a finite field that contains all the points satisfying equation $y^2 = x^3+ax+b$ where $4a^3+27b^2 \neq 0$ along with a distinguished point at infinity denoted by O. The ECC security depends on the difficulty of elliptic curve discrete logarithm problem [10]. The Elliptic Curve Cryptosystems are hard under the discrete logarithmic problem which play a vital role in its security.

2.3 Weil Pairing

The Weil pairing is used to construct admissible pairings that can be used as the basis for cryptographic systems. Let us consider p as a prime number and is given by $p = 12q - 1$ for some random prime q. Let $y^2 = x^3 + 1$ be a super singular elliptic curve(E) over a finite field F_p. A cyclic group having order as $p + 1$ is formed by a group of rational points given by: E(F_p)=$\{(x, y) \in F_pXF_p : (x, y) \in E\}$. Now, as p+1=12q. There is a cyclic subgroup G_1 of order q. Let us consider G as a generator for G_1 and G_2 be the subgroup containing the elements having order q.

2.4 Identity Based Encryption

The identity based encryption schemes were first proposed by Shamir in [22] which is not practical in its approach. Later, Boneh and Franklin [5] proposed a scheme on identity based encryption which was secure and practical. Their scheme efficiently used the concept of bilinear mapping among groups which plays a vital role in our work. The identity based encryption scheme consists of four algorithms. They are: Setup, Extract, Encrypt and Decrypt.

1) Setup: In this algorithm, the system parameters are made public where as the master-key is known to Private Key Generator (PKG). This phase initially takes a security parameter as an input and returns the system parameters and master-key.

2) Extract: This algorithm obtains private key from the given public key.This algorithm takes the input parameters, arbitrary ID $\in \{0, 1\}^*$ and master key as input and return d as output. Here ID is a random string that is used as a public key and d is the private key which will be used later for decryption.

3) Encrypt: This algorithm takes the input parameters, message, ID as input and returns the ciphertext.

4) Decrypt: This algorithm takes the input parameters, cipher-text and d (private key) as input and returns the correspond message.

One of the main concern of IBE is to distribute the role of an authority or a trusted third party among the users. As a result there were many schemes proposed which adopted the secret sharing techniques. Zhou and Haas [26] were the first to propose such a scheme using the concept of threshold cryptography which is not that practical in its approach. Later Kong *et al.* [16] proposed another scheme but it was insecure. Other works [12, 18] distribute only a part of master key in identity-based environments. All of the above works use shamir secret sharing scheme and whenever a new user wants to be part of the network it imposes certain limitations like having a lot of interaction with existing users or not having the same ability as compared to other users. Blundo *et al.* [4] proposed a scheme in which new users can join the network dynamically without the need of any authority by using bivariate polynomials. Some other works Anzai *et al.* [1] and Daza *et al.* [6] used bivariate polynomials to decentralize the role of trusted authority.

2.5 Decentralization

In identity based encryption the master key is stored at the PKG and should be protected. To achieve this we will be distributing the master key among several users by using the concept of threshold cryptography. The users exchange the bivariate polynomial to decentralize the work of PKG. When working in subgroups it is suggested to work in small subgroup of a curve in order to increase the performance of an IBE system. Here we use Weil pairing to decentralize the PKG. In this system, public key of each user is transformed to a point on the group by hashing the ID to a point which is on the curve and later the point is multiplied by a constant.

3 Proposed System

In our system the role of PKG is fully decentralized as discussed in Section 3.2. After the initial exchange of polynomials each user has a share of a secret. He can communicate with other users or can perform subgroup operations using the given protocol.

3.1 Setup

Let L denote the initial set of N users in the network. This initial N users are known as founding users of the network. All those users will run the protocol designed in the initialization phase (specified in Subsection 4.2). The main goal is to decentralize the role of the PKG by using Shamir's secret sharing scheme, weil pairing and identity based encryption. Groups G_1, G_2 are taken for pairing each of them having a hash function. Threshold values t and t^1 are used for performing subgroup operations.

3.2 Initialization

Our scheme will have the following parameters which are made public. A group G which is additive of a prime order q and produced by a random point P under the assumption that the discrete logarithm problem is hard. In addition to the above, a bilinear pairing and two hash

functions are made public, bilinear pairing $e : G \times G \rightarrow G_T$, hash function $h : \{0,1\}^* \rightarrow Z_q$, hash function $H : \{0,1\}^* \rightarrow G$). Two threshold values t, t^1 are chosen, where the threshold value t will be used in significance to test the security of the designed network *i.e.* it will test that maximum $t-1$ nodes are deceptive. Another threshold value t^1 is used for looking after the security of the threshold operations computed in the users subgroup.The required condition for security is $t^1 \leq t \leq L$.
The bilinear pairing e and hash function H are needed to generate the individual keys based on identity or when we want to compute the threshold operations on subgroup of users. Initialization phase of our designed algorithm is described below:

1) Each user in L choses a random bivariate polynomial $F_i(x,z) \in Z_q[x,z]$ with degree utmost $t-1$ in the variable x and z. Here, L denote the initial set of N users in the network. Each polynomial $F(x,z) = \sum_{L_i \in L} F_i(x,z)$ (Here, L_i is the i^{th} user in given initial set of N users) share the same properties. The constant term of the given polynomial is $f_{i,0} = F_i(0,0)$.

2) Each user $L_i \in L$ secretly sends the bi-variate polynomial to the other users $L_j \in L$(founding users) in the form of $F_{ij}(x) = F_i(x, h(L_j))$. Later, user L_i computes $Y_i = f_{i,0}P$ and uses this value in every message.

3) After each user in L performs the above step, each user L_j will compute their final secret value and is given by:

$$\begin{aligned} S_j(x) &= \sum_{L_i \in L} F_j(x) \\ &= \sum_{L_i \in L} F_i(x, h(L_j)) \\ &= F(x,z). \end{aligned}$$

Each user computes their public key and make it public based on the information received from the other users $L_j \in L$. The public key(PK) will be as follows:

$$\begin{aligned} PK &= sP \\ &= \sum_{L_i \in L} f_{i,0}P \\ &= \sum_{L_i \in L} Y_i. \end{aligned}$$

Note: Implicitly secret key(s) is F(0,0). A share $[s_j] = s_j(0) = F(0,0) = F(0, h(L_j))$ of the secret key can be computed by each user in L_j from its partial information $S_j(x)$.This set up runs securely only when $t \leq L$.

3.3 Network Management

After the initialization phase is completed. If a new user N_k desires to be part of the network then he should run the below steps:

1) The new user N_k will select a group L_m which consists minimum of t users in the network and request them to include him in their Network.

2) If any of the user in L_m (suppose N_j) agrees to include this new user(N_k) in their network then he sends the following value:

$$\begin{aligned} S_j(h(N_k)) &= F(h(N_K), h(N_j)) \\ &= F(h(N_j), h(N_k)) \\ &= S_k(h(N_j)). \end{aligned}$$

3) When the new user N_k gets this information from t users then he uses Lagrange interpolation to extract secret polynomial as follows:

$$\begin{aligned} & \sum_{N_k \in L_m} \prod_{N_i \in L_m, i \neq j} \frac{x - h(N_i)}{h(N_j) - h(N_i)} S_j(h(N_k)) \\ = & \sum_{N_k \in L_m} \prod_{N_i \in L_m, i \neq j} \frac{x - h(N_i)}{h(N_j) - h(N_i)} F(h(N_j), h(N_k)) \\ = & F(x, h(N_k)) \\ = & S_k(x). \end{aligned}$$

4) Finally the share $[s_k] = S_k(0)$ is computed by N_k.

3.4 Secure Communication Using IBE

In IBE, the public key is derived directly from the identity of nodes in L_m *i.e.* $pk_m = H(L_m) \in G$ where $H : \{0,1\}^* \rightarrow G$ which is chosen as hash function during initialization phase. Since it is a decentralized network, the user N_k needs to contact other users to compute the secret key sk_m=sH(L_m) where the master secret key is s. The designed protocol is as follows:

1) The user N_k approaches a group of users (L_m) having minimum of t users to request for their share.

2) If any of the user (N_j) in the group of users (L_m) accepts the identification of the user N_k then he sends the following value: $\sigma_j m = S_j(0)H(N_k) = F(0, h(N_j))H(N_k) \in G$.

3) The user N_k should receive t such values to compute the secret key sk_m where

$$sk_m = F(0,0)H(N_k) = sH(N_k) \in G.$$

Then the Encryption and Decryption is done as discussed in [7].

3.5 Subgroup Operations

As mentioned in the initialization phase, each user adopts Shamir secret sharing scheme and holds the shares of secret key of the entire system corresponding to the threshold t. These shares can be used by the users in order to perform certain operations with minimum of t nodes being involved in the network. In our system, the nodes

encrypt the messages among the subgroup(sub) of users by using the Subgroup key. The decryption is possible only when t^1 users in the subgroup cooperate. Now, if a member of the subgroup wants to decrypt the message then the following steps are to be followed in order to get the share of its secret key:

1) The user N_k approaches a group of nodes (L_m) having minimum of t' users.

2) Any user (N_j) in L_m accepting the identity of the new user N_k need to send the following value to N_k:

$$\begin{aligned}\tau_k &= S_j(h(N_k))H(ID_{sub}) \\ &= F(h(N_j), h(N_k))H(ID_{sub}) \in G.\end{aligned}$$

3) The share of the user N_k is computed by using lagrange's interpolation after the user N_k has received t^1 such distinct values (as in above step).The share of the user is given by:

$$[SK_{sub}]_k = F(0, h(L_m))H(ID_{sub}) \in G.$$

3.6 Example

Setup.

- Let the initial set of users $N = \{N_1, N_2, N_3, N_4\}$ No. of users L = 4.
- Public Parmeters: An additive group G of prime order q=4019.
  - The curve used is $E(F_{4019}) : y^2 = x^3 + 1$ k1=67(field of polynomials).
  - The Generator is $P = E(3198, 578)$, Let $th = 3$ and $th^1 = 2$.
- A collision resistant explicit hash function - HTR.
- A collision resistant explicit hash function - HTP.
- Each user chooses a random bivariate polynomial in GF(67):

$$\begin{aligned}N1 &= 3x^2z + 3z^2x + 8xz + 5z + 5x + 2 \\ N2 &= 5x^2z + 5xz^2 + 3xz + 8z + 8x + 5 \\ N3 &= 8x^2z + 8xz^2 + 5xz + 3x + 3z + 3 \\ N4 &= 2x^2z + 2xz^2 + 4xz + 8z + 8x + 4.\end{aligned}$$

- The implicit polynomial defined by all the users is

$$\begin{aligned}F(x, z) &= N_1 + N_2 + N_3 + N_4 \\ &= 18x^2z + 18xz^2 + 20xz + 24x + 24z + 14.\end{aligned}$$

The secret s of the NETWORK is $F(0, 0) = 14$.

- Each user secretly sends to each of other founding users the univariate polynomial $F_{ij} = F_i(x, h(N_j))$.
- The hash values of the users computed using standard hash function are

$$\begin{aligned}h_{n1} &= HTR('user1', k1) = 37 \\ h_{n2} &= HTR('user2', k1) = 54 \\ h_{n3} &= HTR('user3', k1) = 25 \\ h_{n4} &= HTR('user4', k1) = 17\end{aligned}$$

Share Distribution.

- Each user sends the following values to other users:
- N1 also includes $Y_1 = 2Q = (167, 1358)$, $N_{11} = 44x^2 + 53x + 53$, $N_{12} = 28x^2 + 6x + 4$, $N_{13} = 8x^2 + 3x + 60$, $N_{14} = 51x^2 + 3x + 20$. Similarly N2, N3 and N4 also send data to other users.
- Then all the users calculate their secret univariate polynomial from the received values.

$$\begin{aligned}S_1(x) &= 63x^2 + 13x + 31 \\ S_2(x) &= 34x^2 + 59x + 37 \\ S_3(x) &= 48x^2 + 49x + 11 \\ S_4(x) &= 38x^2 + 5x + 20.\end{aligned}$$

- The public key, $PK = sQ = 14E(3198, 578) = E(100, 1874)$.
- PK should also be equal to $Y_1 + Y_2 + Y_3 + Y_4 = E(167, 1358) + E(152, 1437) + E(1356, 3203) + E(3863, 2497) = E(100, 1874)$.

Network Communication Example.

- If user N_5 wants to join the network, It should identify it self to 3 other users and request for acceptance: $\{N_1, N_2, N_3\}$,

$$h_{n5} = HTR('user5', k1) = 27.$$

- N_5 receives the following values

$$N_{15} = 12, N_{25} = 18, N_{35} = 12.$$

- N_5 computes its secret univariate polynomial by using Lagrange interpolation:

$$S_5(x) = 17 * x^2 + 18 * x + 59.$$

Obtention of Individual Keys by Indentity Based Encryption (IBE) Scenario Example.

- Take a public parameter $P_{Pub} = msk \times P$.
- The hash to the point HTP method signature is $QHTP(E, p, q, id, hashfcn)$.
- The key generation is: This method takes public params and secret and generate the key to respective ID.
  $defKeyGen(E, p, q, hashfcn, msk, id)$:

$$\begin{aligned}Q_{id} &= HTP(E, p, q, id, hashfcn) \\ sk_{id} &= msk \times Q_{id} \\ sec &= (Q_{id}, sk_{id})\end{aligned}$$

Return sec.

- Encrypt,This method take public params ,id,message and return cipher text

$$Encrypt(p, q, P, id, m, Q_{id}).$$

- Decrypt,This method takes public params, secret key and cipher text and decrypt the message.

$$Decrypt(p, q, P, C, S_{id}).$$

- now user n2 want to send the message $m = 1712$ to user n1. n2 calls encrypt method.

$$C = Encrypt(p, q, P,' Node1', m, Q_{id1});$$

Cipher text is (1807, 1481) 1718, after receiving encrypted message user n1 calls decrypt method.

$$msg = Decrypt(p, q, P, C, Sk_{id1}).$$

After decrypting message is m= 1712.

Threshold Decryption on Sugroup Example.

- Take the shares of users L={N1,N2,N3,N4} as a subgroup. Each user is having its own secret polynomial.

$$\begin{aligned} S_1(x) &= 63x^2 + 13x + 31; \\ S_2(x) &= 34x^2 + 59x + 37 \\ S_3(x) &= 48x^2 + 49x + 11 \\ S_4(x) &= 38x^2 + 5x + 20. \end{aligned}$$

- create an id for the sub group. $hsg1234 = hfun('SG1234', q)$.
- To find share of jth node remaining users contribute their shares and lagranges interpolation is applied.
  Share of user1 (3125, 1868), user2 (2292, 3913), user3 (2350, 780) and user4 (163, 2657).
  To verify the shares of users caluclate the hash of users they are $rd1, rd2, rd3, rd4$.

$$\begin{aligned} sg1 &= int(rd1)HTP(E, p, q,' SG1234', hashfcn) \\ sg2 &= int(rd2)HTP(E, p, q,' SG1234', hashfcn) \\ sg3 &= int(rd3)HTP(E, p, q,' SG1234', hashfcn) \\ sg4 &= int(rd4)HTP(E, p, q,' SG1234', hashfcn). \end{aligned}$$

  The shares of the users must be (Calculated from F). Share of user1 (3125, 1868), user2 (2292, 3913), user3 (2350, 780) and user4 (163, 2657). Secret of Subgroup is $14HTP(E, p, q,' SG1234', hashfcn)$; Secret of Subgroup is (3857,1351).
- Secret of a subgroup is Lagrange interpolation is applied on users then we get $k1, k2, k3$ from nodes $n1, n2, n3$.

$$\begin{aligned} a1 &= int(k1)sg1; \\ a2 &= int(k2)sg2; \\ a3 &= int(k3)sg3. \end{aligned}$$

  Secret of subgroup $a1 + a2 + a3 = (3857, 1351)$.
- Here ENCRYPTION and DECRYPTION methods are same but in decryption method we have the one more parameter *i.e.* k11 for user 1 it is formed from Lagrange interpolation with t^1 users.
- Any user want to send a message. let $m = 50$, encrypt the message.

$$C = Encrypt(p, q, P,' SG123', m1, Q_{id}).$$

  Threshold is 2 so any two users $n1$ and $n2$ compute l1 = Decrypt(p,q,P,C,sg1,k11); l2 = Decrypt(p,q,P,C,sg2,k22); r1 = l1 l2; and r = HTR(r1,q).
  Now the encrypted message with r output is message = 50.

4 Security Analysis

For a public key encryption scheme the acceptable notion for security is cipher-text security. But the definition concerning the chosen cipher-text should be strengthened. This is because if an adversary outbreaks the public ID of an identity based system then the adversary might posses the private keys of the users. Thus the designed system should withstand such an attack and should be secure. We assume that the identity based encryption system is secure against chosen cipher-text attack.

Note: The adversary A should not have any advantage against the challenger.

Setup: The initialization phase is run by the challenger by taking the security parameter k as input. The system parameters are obtained by the adversary but the master key is kept with it.

Phase 1: The adversary issues either the extraction query or the decryption query.

- Extraction Query: The extract algorithm (defined in 3.1) is run by the challenger. As a result of this, the private key is generated corresponding to particular public key. This private key is sent to the adversary.
- Decryption Query: The extract algorithm (defined in Section 3.1) is run by the challenger. As a result of this, the private key is generated corresponding to particular public key. The decrypt algorithm is run by it using the private key to decrypt the cipher-text. The resulting plain text is sent to the adversary.

Challenge: Two equal length plain texts and ID are generated by adversary after Phase 1 is over. ID is the parameter on which the adversary desired to be challenged.ID did not appear anywhere in the extraction of query in phase 1. A random bit b $\in$ {0,1} is picked by challenger and sets c=Encrypt(parameters,ID,M_b). Challenger sends C to the adversary as a challenge.

Phase 2: In this phase more and more queries are posed by the adversary and it can be either of the following:

- Extraction Query: Here $ID_i \neq$ ID.Then the challenger replies as in phase1.
- Decryption Query: Challenger replies as in phase1 if $(ID_i, C_i) \neq (ID, C)$. Here C is the cipher-text notation.

Guess: $b^1 \in \{0, 1\}$ is displayed by the adversary and the game is won by adversary if $b^1 = b$.

Adversary A has the advantage of attacking the identity based scheme with the help of following function: The function takes the security parameter k as input.Adv[k]=$|Pr[b^1 = b] - \frac{1}{2}|$. This is done by the random bits chosen by the adversary and challenger. The security of the chosen cipher text is demonstrated with the help of this game for Identity based encryption schemes.

Attack: Let the number of players trying to recover the secret S_i be less than or equal to $t_i - 1$. Here t is the threshold value.

Analysis: The recovery of the secret in the proposed scheme completely revolves around the concept of Lagrange's Interpolation polynomial. In order to solve t_i in the process of getting to know the unknown symbol, we are definitely going to need t_i number of equations. Therefore, it is only t_i or more players who can have a complete knowledge of the secret. There is no chance for t_i or lesser players to crack the secret.

5 Conclusion

Now a days many applications demands the network without the presence of trusted third party (TTP). This can be achieved by distributing the role of TTP among the network users using secret sharing concept. In our paper we proposed an efficient way to decentralize the network and to establish a secure communication among the users of the network using Identity based encryption. We also discussed the suitable protocol to perform sub group operations among the sub set of users of a network. Our scheme is useful for the applications where secure communication is required without the presence of trusted third party.

References

[1] J. Anzai, N. Matsuzaki and T. Matsumoto, "A quick group key distribution scheme with entity revocation," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 333-347, 1999.

[2] A. Basit, N. C. Kumar, V. C. Venkaiah, S. A. Moiz, A. N. Tentu and W. Naik, "Multi-stage Multi-secret sharing scheme for hierarchical access structure," in *International Conference on Computing, Communication and Automation (ICCCA'17)*, pp. 557-563, 2017.

[3] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of the National Computer Conference*, vol. 48, pp. 313-317, 1979.

[4] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Annual International Cryptology Conference*, pp. 471-486, 1992.

[5] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Annual International Cryptology Conference*, pp. 213-229, 2001.

[6] V. Daza, J. Herranz and G. Sez, "Constructing general dynamic group key distribution schemes with decentralized user join," in *Australasian Conference on Information Security and Privacy*, pp. 464-475, 2003.

[7] V. Daza, J. Herranz, P. Morillo and C. Rafols, "Cryptographic techniques for mobile ad-hoc networks," *Computer Networks 51*, no. 18, 2007.

[8] G. Frey, M. Muller and H. G. Ruck. "The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems," *IEEE Transactions on Information Theory 45*, no. 5, pp. 1717-1719, 1999.

[9] O. Goldreich, R. Ostrovsky, E. Petrank, "Computational complexity and knowledge complexity," vol. 27, no. 4, pp. 1116-1141, 2001.

[10] M. S. Hwang, C. C. Lee, J. Z. Lee, and C. C. Yang, "A secure protocol for bluetooth piconets using elliptic curve cryptography", *Telecommunication Systems*, vol. 29, no. 3, pp. 165-180, 2005.

[11] A. Joux, "A one round protocol for tripartite Diffie-Hellman," in *International Algorithmic Number Theory Symposium*, pp. 385-393, 2000.

[12] A. Khalili, J. Katz and W. A. Arbaugh, "Toward secure key distribution in truly ad-hoc networks," in *Symposium on Proceedings of Applications and the Internet Workshops*, pp. 342-346, 2003.

[13] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation 48*, no. 177, pp. 203-209, 1987.

[14] N. C. Kumar, A. Basit, P. Singh, V. C. Venkaiah and Y. V. Rao, "Node authentication using BLS signature in distributed PKI based MANETS," *Cryptography and Security*, vol.9, no. 4, 2017.

[15] N. C. Kumar, A. Basit, P. Singh and V. C. Venkaiah "Proactive secret sharing for long lived MANETs using elliptic curve cryptography," in *IEEE International Conference on Inventive Computing and Informatics (ICICI'17)*, pp. 312-316, 2017.

[16] H. Luo, J. Kong, P. Zerfos, S. Lu and L. Zhang, "URSA: Ubiquitous and robust access control for mobile ad hoc networks," *IEEE/ACM Transactions on Networking (ToN'04) 12*, no. 6, pp. 1049-1063, 2004.

[17] A. J. Menezes, T. Okamoto and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *IEEE Transactions on Information Theory 39*, no. 5, pp. 1639-1646, 1993.
[18] J. Pan, L. Cai, X. S. Shen and J. W. Mark, "Identity-based secure collaboration in wireless ad hoc networks," *Computer Networks 51*, no. 3, pp. 853-865, 2007.
[19] R. Sakai, K. Ohgishi and M. Kasahara, *Cryptosystems Based on Pairing*, SCIS 2000-C20, Jan. 2000. (`https://www.researchgate.net/publication/243538884_Cryptosystem_based_on_Pairings`)
[20] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 47-53, 1984.
[21] A. Shamir, "How to share a secret," *Communications of the ACM 22*, no. 11, pp. 612-613, 1979.
[22] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 47-53, 1984.
[23] N. Singh, A. N. Tentu, A. Basit and V. C. Venkaiah, "Sequential secret sharing scheme based on Chinese remainder theorem," in *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC'16)*, pp. 1-6, 2016.
[24] A. N. Tentu, A. Basit, K. Bhavani and V. C. Venkaiah, "Multi-secret sharing scheme for level-ordered access structures," in *International Conference on Number-Theoretic Methods in Cryptology*, pp. 267-278, 2017.
[25] X. Yi, "An identity-based signature scheme from the Weil pairing," *IEEE Communications Letters 7*, no. 2, pp. 76-78, 2003.
[26] L. Zhou and J. H. Zygmunt, "Securing ad hoc networks," *IEEE Network 13*, no. 6, pp. 24-30, 1999.

Biography

N Chaitanya Kumar received M.Tech from JNTU Hyderabad and he did Bachelor degree in computer science. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Information security, Cryptography in MANET.

Abdul Basit received Master of computer application from Jamia Hamdard University New Delhi. He did Bachelor of Science in Information technology from SMU Gangtok. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Information security, Cryptography and Cyber security.

Priyadarshi Singh received M.Tech from IIT(ISM) Dhanbad. He did Bachelor degree in Information Technology. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Cryptography, Public key infrastructure.

V. Ch. Venkaiah obtained his PhD in 1988 from the Indian Institute of Science (IISc), Bangalore in the area of scientific computing. He worked for several organisations including the Central Research Laboratory of Bharat Electronics, Tata Elxsi India Pvt. Ltd., Motorola India Electronics Limited, all in Bangalore. He then moved onto academics and served in IIT Delhi, IIIT Hyderabad and C R Rao Advanced Institute of Mathematics, Statistics, and Computer Science. He is currently serving in School of Computer and Information Sciences, University of Hyderabad. He is a vivid researcher. He designed algorithms for linear programming, subspace rotation and direction of arrival estimation, graph coloring, matrix symmetriser, integer factorisation, cryptography, knapsack problem, *etc.*