Authenticated Multiple Key Establishment Protocol for Wireless Sensor Networks

Jayaprakash Kar

Information Security Research Group
Faculty of Computing & Information Technology
Department of Information Systems
King Abdulaziz University, Kingdom of Saudi Arabia
jayaprakashkar@yahoo.com

Abstract. The article proposes a provably secure authenticated multiple key establishment protocol for Wireless Sensor Network. Security of the protocol is based on the computational infeasiblity of solving Elliptic Curve Discrete Logarithm Problem and Computational Diffie-Hellman Problem on Bilinear Pairing. User authentication is a one of the most challenging security requirement in wireless sensor networks (WSN). It is required to establish the correct session key between two adjacent nodes of WSNs to achieve this security goal. Here we prove that, the proposed protocol is secure against the attack on data integrity and known key security attack on session key. It also provides perfect forward secrecy.

Keywords: Bilinear pairing, ECDLP, Key exchange, Session key.

1 Introduction

WSN systems are usually deployed in hostile environments where they encountered a wide variety of malicious attacks. Information that is the cooked data collected within the sensor network is valuable and should be kept confidential. In order to protect this transmitted information or messages between any two adjacent sensor nodes key establishment protocol and a mutual authentication are required for wireless sensor networks. Due to nature restrictions like low power, less storage space, low computation ability and short communication range of sensor nodes, most conventional protocols establish authenticated multiple keys between any two adjacent sensor nodes by adopting a key pre-distribution approach. However, these techniques have vulnerability. With rapid growth of cryptographic techniques, recent results show that Elliptic Curve Cryptography (ECC) is suitable for resource-limited WSNs. Cryptosystems based on Elliptic Curve Cryptography are especially interesting for sensor networks since they are more efficient in resource utilization than any other public key techniques [1] [12]. The computational capability of sensor nodes are limited, so traditional public-key cryptography, in which the computation of modular exponentiation is required, cannot be implemented on WSNs. Fortunately, Elliptic curve cryptosystem (ECC) [6] [2], compared with other public-key cryptography, has significant advantages like smaller key sizes, faster computations. Thus, ECC-based key establishment protocols are more suitable for resource constraints sensor node than other cryptosystem.

2 Preliminaries

Definition 1. Bilinearity *Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of same prime order q. $\mathbb{G}_1$ is an additive group and $\mathbb{G}_2$ is a multiplicative group. Let e be a computable bilinear map $e : \mathbb{G}_1 X \mathbb{G}_1 \rightarrow \mathbb{G}_2$, which satisfies the following properties:*

- **Bilinear***: $e(aP, bQ) = e(P,Q)^{ab}$, where $P,Q \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_q^*$ and for $P,Q,R \in \mathbb{G}_1, e(P+Q,R) = e(P,R)e(Q,R)$.*
- **Non-degenerate***: If P is a generator of $\mathbb{G}_1$, then $e(P,P)$ is generator of $\mathbb{G}_2$. There exists $P,Q \in \mathbb{G}$ such that $e(P,Q) \neq 1_{\mathbb{G}_2}$*
- **Computability***: There exists an efficient algorithm to compute $e(P,Q)$ for all $P,Q \in \mathbb{G}_1$.*

We call such a bilinear map e is an admissible bilinear pairing.

2.1 Mathematical Assumption

Definition 2. Bilinear Parameter Generator *: A bilinear parameter generator $\mathcal{G}$ is a probabilistic polynomial time algorithm that takes a security parameter k as input and outputs a 5-tuple $(q, \mathbb{G}_1, \mathbb{G}_2, e, P)$ as the bilinear parameters, including a prime number q with $|q| = k$, two cyclic groups $\mathbb{G}_1, \mathbb{G}_2$ of the same order q, an admissible bilinear map $e : \mathbb{G}_1 X \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$*

Definition 3. Bilinear Diffie-Hellman Problem*: Let $(q, \mathbb{G}_1, \mathbb{G}_2, e, P)$ be a 5-tuple generated by $\mathcal{G}(k)$, and let $a, b, c \in \mathbb{Z}_q^*$. The BDHP in $\mathbb{G}$ is as follows: Given (P, aP, bP, cP) with $a, b, c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc} \in \mathbb{G}_T$. The (t, ϵ) -BDH assumption holds in $\mathbb{G}$ if there is no algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{BDH}(\mathcal{A}) = Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}] \geq \epsilon$$

where the probability is taken over all possible choices of (a, b, c). Here the probability is measured over random choices of $a, b, c \in \mathbb{Z}_q^*$ and the internal random operation of A. More formally, for any PPT algorithm $\mathcal{A}$ consider the following experiment:
Let $\mathcal{G}$ be an algorithm which on input 1^k outputs a (description of a) group G of prime order q (with $|q| = k$) along with a generator $P \in \mathbb{G}$. The computational Diffie-Hellman (CDH) problem is the following:

$\underline{\mathbf{Exp}_{\mathcal{G}(k)}^{CDH}}$

1. $(\mathbb{G}, q, P) \leftarrow \mathcal{G}(1^k)$
2. $a, b, c \leftarrow \mathbb{Z}_q^*$
3. $U_1 = aP, U_2 = bP, U_3 = cP$
4. if $W = e(P, P)^{abc}$ return 1 else return 0

We assume that BDHP is a hard computational problem: letting q have the magnitude $2k$ where k is a security parameter, there is no polynomial time (in k) algorithm which has a non-negligible advantage (again, in terms of k) in solving the BDHP for all sufficiently large k.

Definition 4. Decisional Diffie-Hellman Problem *: Let $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ be a 5-tuple generated by $\mathcal{G}(k)$, and let $a, b, c, r \in \mathbb{Z}_q^*$. The DBDHP in $\mathbb{G}$ is as follows: Given Given (P, aP, bP, cP, r) with some $a, b, c \in \mathbb{Z}_q^*$, Output is* **yes** *if $r = e(P, P)^{abc}$ and* **no** *otherwise. The (t, ϵ)-HDDH assumption holds in $\mathcal{G}$ if there is no algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{DBDH}(\mathcal{A}) = |Pr[\mathcal{A}(P, aP, bP, cP, e(P, P)^{abc})) = 1] - Pr[\mathcal{A}(P, aP, bP, cP, r) = 1]| \geq \epsilon$$

where the probability is taken over all possible choices of (a, b, c, h).

Definition 5. Hash Decisional Diffie-Hellman Problem *:Let $(q, \mathbb{G}, \mathbb{G}_T, e, g)$ be a 5-tuple generated by $\mathcal{G}(k)$,$\mathcal{H} : \{0, 1\}^* \rightarrow \{0, 1\}^l$ be a secure cryptographic hash function, whether l is a security parameter, and let $x, y \in \mathbb{Z}_q^*, h \in \{0, 1\}^l$, the HDDH problem in $\mathbb{G}$ is as follows: Given (P, aP, bP, cP, h), decide whether it is a hash Diffie-Hellman tuple $((P, aP, bP, cP\mathcal{H}(e(P, P)^{abc}))$. If it is right, outputs 1; and 0 otherwise. The (t, ϵ)-HDDH assumption holds in $\mathcal{G}$ if there is no algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{HDDH}(\mathcal{A}) = |Pr[\mathcal{A}(P, aP, bP, cP\mathcal{H}(e(P, P)^{abc})) = 1] - Pr[\mathcal{A}(P, aP, bP, cP, h) = 1]| \geq \epsilon$$

where the probability is taken over all possible choices of (a, b, h).

3 Notations

The following notations and system parameters are used throughout the article.

- P: a generator of order n on an Elliptic Curve E and satisfies $n \times P = \mathcal{O}$. where q is a large prime number and $\mathcal{O}$ is a point at infinity.
- q : the order of the group.
- $SK_i, 1 \leq i \leq 4$: the established session key between node i and node j.
- Q_i: the public key of sensor node i.
- λ_i : the private key of node i, $\lambda_i \in \mathbb{Z}_q^*$.

4 Security Model

We follow the security model based on et.al. [5] [3]

- **Protocol Participants**: Each participant in authenticated multiple key establishment protocol is a node $i \in \mathcal{I}$.
- **Protocol execution**: The interaction between an adversary A and the protocol participants occurs only via oracle queries, which model the adversary capabilities in a real attack. During the execution, the adversary may create several instances of a participant. While in a concurrent model, several instances may be active at any given time, only one active user instance is allowed for a given intended partner and password in a non-concurrent model. Let U^i denote the instance i of a participant U and let b be a bit chosen uniformly at random. The query types available to the adversary are as follows:
  - $Execute(C^i, S^j)$: This query models passive attacks in which the attacker eavesdrops on honest executions between a client instance C^i and a server instance S^j. The output of this query consists of the messages that were exchanged during the honest execution of the protocol.
  - $Send(U^i, m)$:This query models an active attack, in which the adversary may tamper with the message being sent over the public channel. The output of this query is the message that the participant instance U^i would generate upon receipt of message m.
  - $Reveal(U^i)$: This query models the misuse of session keys by a user. If a session key is not defined for instance U^i or if a $Test$ query was asked to either U^i or to its partner, then return $\perp$. Otherwise, return the session key held by the instance U^i.
  - $Test(U^i)$: This query tries to capture the adversary's ability to tell apart a real session key from a random one. If no session key for instance U^i is defined, then return the undefined symbol $\perp$. Otherwise, return the session key for instance U^i if $b = 1$ or a random key of the same size if $b = 0$.

Notation. An instance U^i is said to be opened if a query $Reveal(U^i)$ has been made by the adversary.We say an instance U^i is unopened if it is not opened. We say an instance U^i has accepted if it goes into an accept mode after receiving the last expected protocol message. Partnering. The definition of partnering uses the notion of session identifications (sid). More specifically, two instances U_1^i and U_2^j are said to be partners if the following conditions are met: (i) Both U_1^i and U_2^j accept. (ii) Both U_1^i and U_2^j share the same session identifications; (iii) The partner identification for U_1^i is U_2^j and vice-versa; and (iv) no instance other than U_1^i and U_2^j accepts with a partner identification equal to U_1^i or U_2^j. In practice, the sid could be taken to be the partial transcript of the conversation between the client and the server instances before the acceptance.

Freshness. The notion of freshness is defined to avoid cases in which adversary can trivially break the security of the scheme. The goal is to only allow the adversary to ask Test queries to fresh oracle instances. More specifically, we say an instance U^i is fresh if it has accepted and if both U^i and its partner are unopened. Semantic security. Consider an execution of the key establishment protocol P by an adversary A, in which the latter is given access to the $Reveal$, $Execute$, $Send$, and $Test$ oracles and asks a single $Test$ query to a $fresh$ instance, and outputs a guess bit $b^{'}$. Such an adversary is said to win the experiment defining the semantic security if $b^{'} = b$, where b is the hidden bit used by the $Test$ oracle. Let $Succ$ denote the event in which the adversary is successful. The advantage of an adversary A in violating the semantic security of the protocol P .

$$\mathbf{Adv}_P^{ake}(A) = 2 \cdot Pr[Succ] - 1 \text{ and } \mathbf{Adv}_P^{ake}(t, R) = max\{\mathbf{Adv}_{ake}P(A)\}$$

Where maximum is considered over all A with most t time complexity using resources at most R $i.e$ the number of queries to its oracles. The definition of time-complexity that we use henceforth is the usual one, which includes the maximum of all execution times in the experiments defining the security plus the code size.

5 Proposed Protocol based on ECC

Consider two arbitrary nodes i and j would like to share session keys to establish secure communication. Node i has computed long-term private and public key as $Q_i = \lambda_i \cdot P$. Similarly node j has computed his long-term private and public key as $Q_j = \lambda_j \cdot P$.

The set of session key are computed by node i and j as follows

- **Step-I** Node i chooses two r_{i1} and r_{i2} randomly and computes $V_{i1} = r_{i1} \cdot P$, $V_{i2} = r_{i2} \cdot P$. Where $r_{i1}, r_{i2} \in \mathbb{Z}_q^*$. Let X_{i1} and X_{i2} be x-coordinate of the points V_{i1} and V_{i2} respectively. Then node i computes S_i by the following equation:

$$S_i = \lambda_i - r_{i1}X_{i1} - r_{i2}X_{i2} \bmod n \tag{1}$$

  Node i sends the tuples $\{V_{i1}, V_{i2}, S_i, Cert(Q_i)$ to node j

- **Step-II** Similarly node j chooses r_{j1} and r_{j2} randomly and computes $V_{j1} = r_{j1} \cdot P$, $V_{j2} = r_{j2} \cdot P$. Where $r_{j1}, r_{j2} \in \mathbb{Z}_q^*$. let X_{j1} and X_{j2} be x-coordinate of the points V_{j1} and V_{j2} respectively. Node j computes S_j be the following equation:

$$S_j = \lambda_j - r_{j1}X_{j1} - r_{j2}X_{j2} \bmod n \tag{2}$$

  Node j sends the tuples $\{V_{j1}, V_{j2}, S_j, Cert(Q_j)$ to node i

- **Step-III** Node i prove the validation of message by the following equation taking x-coordinate X_j and X_{j2} from V_{j1} and V_{j2}.

$$Q_j = S_j \cdot P + X_{j1} \cdot V_{j1} + X_{j2} \cdot V_{j2} \tag{3}$$

  If it hold, node i compute the following set of session keys are

$$\begin{aligned} SK_1 &= r_{i1} \cdot V_{j1} \\ SK_2 &= r_{i1} \cdot V_{j2} \\ SK_3 &= r_{i2} \cdot V_{j1} \\ SK_4 &= r_{i2} \cdot V_{j2} \end{aligned}$$

- **Step-IV** Similarly node j prove the validation of message by the following equation taking x- coordinate X_i and X_{i2} from V_{i1} and V_{i2}.

$$Q_i = S_i \cdot P + X_{i1} \cdot V_{i1} + X_{i2} \cdot V_{i2} \tag{4}$$

  If it hold, node i compute the following set of session keys as

$$\begin{aligned} SK_1 &= r_{j1} \cdot V_{i1} \\ SK_2 &= r_{j1} \cdot V_{i2} \\ SK_3 &= r_{j2} \cdot V_{i1} \\ SK_4 &= r_{j2} \cdot V_{i2} \end{aligned}$$

The protocol is presented as

Node i		Node j
Select r_{i1} and r_{i2} randomly Where $r_{i1}, r_{i2} \in \mathbb{Z}_q^*$ Computes $V_{i1} = r_{i1} \cdot P, V_{i2} = r_{i2} \cdot P$ Computes $S_i = \lambda_i - r_{i1}X_{i1} - r_{i2}X_{i2} \bmod n$		
	$\{V_{i1}, V_{i2}, S_i, Cert(Q_i)\}$ $\longrightarrow$	
		Take the X_{i1} and X_{i2} from V_{i1} and V_{i2} Verify $Q_i \overset{?}{=} S_iP + X_{i1}V_{i1} + X_{i2}V_{i2}$ If holds computes the Session key $SK_1 = r_{j1} \cdot V_{i1}$ $SK_2 = r_{j2} \cdot V_{i1}$ $SK_3 = r_{j1} \cdot V_{i2}$ $SK_4 = r_{j1} \cdot V_{i2}$ Select random numbers r_{j1} and r_{j2} Where $r_{j1}, r_{j2} \in \mathbb{Z}_q^*$ Computes $V_{j1} = r_{j1} \cdot P, V_{j2} = r_{j2} \cdot P$ Computes $S_j = \lambda_j - r_{j1}X_{j1} - r_{j2}X_{j2} \bmod n$
	$\{V_{j1}, V_{j2}, S_j, Cert(Q_j)\}$ $\longleftarrow$	
Take the X_{j1} and X_{j2} from V_{j1} and V_{j2} Verify $Q_j \overset{?}{=} S_jP + X_{j1}V_{j1} + X_{j2}V_{j2}$ If holds computes the Session key $SK_1 = r_{i1} \cdot V_{j1}$ $SK_2 = r_{i1} \cdot V_{j1}$ $SK_3 = r_{i2} \cdot V_{j2}$ $SK_4 = r_{i2} \cdot V_{j2}$		

6 Security Analysis

In this section, we analyze the security of our proposed protocol . The security of the protocol is based on the difficulty of breaking of Elliptic Curve Discrete Logarithms. We claim that the proposed protocol is resistant against attack on data integrity of sensor node. Also this protect the known session keys, if the adversary can able to compute the previous session keys. Subsequently we prove that the protocol achieves the most important security requirement implicit key authentication and full forward secrecy.

Definition 6. *Authentication multiple key establishment protocol is said to achieve the property of data integrity, if there is no polynomial time algorithm that can alter or manipulate the transmitted messages.*

Theorem 1 *The proposed Protocol is resistant against the attack on data integrity if and only if Elliptic Curve Discrete Logarithm Problem is hard to solve(ECDLP).*

Proof : While the node i sends the sensitive data to another node j by the communication channel, the adversary alter or manipulate the data and cheat the honest nodes by relying the wrong session keys. Assume that, adversary would like to compute S_i to validate the verification equation-4 for cheating node j. He can select randomly two points V_{i1} and V_{i2} and extract the x-coordinate X_{i1} and X_{i2} respectively. After that, he has to find μ in the elliptic curve that satisfies the Equation $\mu \cdot P = Q_i - X_{i1}V_{i1} - X_{i2}V_{i2}$. But to compute μ in the elliptic curve, it is required for the adversary to solve Elliptic Curve Discrete Logarithm Problem. Therefore it is computationally infeasible to forge a valid message to cheat node j by relying invalid common session keys. $\square$

Definition 7. *A protocol can protect the subsequent session keys from disclosing even if the previous session keys are revealed by the intendant user is called Known key security.*

Theorem 2 *It is computationally infeasible for an adversary to generate the correct session keys even if the previous keys are disclosed.*

Proof : Node i and j select fresh random number in each round of the protocol and compute S_i and S_j by equation-1 and 2. This implies that the four generated session keys are distinct and does not depend in each round in the execution of the protocol. It is computationally infeasible for the adversary to derive the random numbers in each round of the protocol that need to compute the session key. Hence the proposed protocol is resistant against known key attack. □

Definition 8. *An authenticated multiple Key establishment protocol provides perfect forward secrecy if the compromise of both the node's secret keys cannot results in the compromise of previously established session keys [10] [11].*

Theorem 3 *The proposed protocol provides perfect forward secrecy if and only if Elliptic Curve Discrete Logarithm Problem is hard to solve(ECDLP).*

Proof : From the above equation, Session keys are established by established by two random numbers and the generator of group of points on Elliptic curve. Therefore if is infeasible for the adversary to derive previous session keys by the long-term secret keys directly. The adversary would like to take publicly known information and use the following equation to derive possible session keys.
$SK_{ij} = \lambda_i \lambda_j \cdot P$
$\lambda_i = S_i + r_{i1}X_{i1} + r_{i2}$ and $\lambda_j = S_j + r_{j1}X_{j1} + r_{j2}$
$\lambda_i\lambda_j = (S_i + r_{i1}X_{i1} + r_{i2})(S_j + r_{j1}X_{j1} + r_{j2})$
$= S_iS_j + S_ir_{j1}X_{j1} + S_ir_{j2} + r_{i1}X_{i1}S_j + r_{i1}r_{j1}X_{i1}X_{j1} + r_{i1}r_{j1}X_{i1} + r_{i2}r_{j1}X_{j1} + r_{i2}r_{j2}$
So $\lambda_i\lambda_j \cdot P = (S_iS_j + S_ir_{j1}X_{j1} + S_ir_{j2} + r_{i1}X_{i1}S_j + r_{i1}r_{j1}X_{i1}X_{j1} + r_{i1}r_{j1}X_{i1} + r_{i2}r_{j1}X_{j1} + r_{i2}r_{j2}) \cdot P = S_iS_jP + S_ir_{j1}X_{j1}P + S_ir_{j2}P + r_{i1}X_{i1}S_jP + r_{i1}r_{j1}X_{i1}X_{j1}P + r_{i1}r_{j1}X_{i1}P + r_{i2}r_{j1}X_{j1}P + r_{i2}r_{j2}P$
$= S_iS_jP + S_iX_{j1}V_{j1} + S_iX_{j2}V_{j2} + S_jX_{i1}V_{i1} + S_jX_{i2}V_{i2} + X_{i1}X_{j1}SK_1 + X_{i1}X_{j2}SK_2 + X_{i2}X_{j1}SK_3 + X_{i2}X_{j2}SK_4$

We can note that the above equation consists of four unknown variables. Therefore it is not possible for the adversary to solve the equation to compute the correct session keys. On the other hand, the adversary may try to compute random numbers r_{i1}, r_{i2}, r_{j1} and r_{j2} from the publicly known parameters V_{i1}, V_{i2}, V_{j1} and V_{j2}. For that it need to solve ECDLP problem. Hence the proposed protocol provides perfect forward secrecy.□

7 Proposed Protocol on Bilinear Pairings

Consider two arbitrary nodes i and j would like to share session keys to establish secure communication. Node i has computed long-term private and public key as $Q_i = \lambda_i \cdot P$. Similarly node j has computed his long-term private and public key as $Q_j = \lambda_j \cdot P$.

The set of session key are computed by node i and j as follows

- **Step-I** Node i chooses two r_{i1} and r_{i2} randomly and computes $V_{i1} = r_{i1} \cdot P$, $V_{i2} = r_{i2} \cdot P$. Where $r_{i1}, r_{i2} \in \mathbb{Z}_q^*$. Let X_{i1} and X_{i2} be x-coordinate of the points V_{i1} and V_{i2} respectively. Then node i computes S_i by the following equation:
$$S_i = (r_{i1}X_{i1} + r_{i2}X_{i2}) \cdot V_{i1} + \lambda_i \cdot V_{i2} \bmod n \quad (5)$$
Node i sends the tuples $\{V_{i1}, V_{i2}, S_i, Cert(Q_i)$ to node j
- **Step-II** Similarly node j chooses r_{j1} and r_{j2} randomly and computes $V_{j1} = r_{j1} \cdot P$, $V_{j2} = r_{j2} \cdot P$. Where $r_{j1}, r_{j2} \in \mathbb{Z}_q^*$. let X_{j1} and X_{j2} be x-coordinate of the points V_{j1} and V_{j2} respectively. Node j computes S_j be the following equation:
$$S_j = (r_{j1}X_{j1} + r_{j2}X_{j2}) \cdot V_{j1} + \lambda_j \cdot V_{j2} \bmod n \quad (6)$$
Node j sends the tuples $\{V_{j1}, V_{j2}, S_j, Cert(Q_j)$ to node i

– **Step-III** Node i prove the validation of message by the following equation taking x-coordinate X_j and X_{j2} from V_{j1} and V_{j2}.

$$e(S_j, P) = e(X_{j1}V_{j1} + X_{j2}V_{j2}, V_{j2}) \cdot e(V_{j2}, Q_j) \tag{7}$$

If it hold, node i compute the following set of session keys are

$$\begin{aligned} SK_1 &= e(r_{i1}V_{j1}, Q_i + Q_j) \\ SK_2 &= e(r_{i1}V_{j2}, Q_i + Q_j) \\ SK_3 &= e(r_{i2}V_{j1}, Q_i + Q_j) \\ SK_4 &= e(r_{i2}V_{j2}, Q_i + Q_j) \end{aligned}$$

– **Step-IV** Similarly node j prove the validation of message by the following equation taking x- coordinate X_i and X_{i2} from V_{i1} and V_{i2}.

$$e(S_i, P) = e(X_{i1}V_{i1} + X_{i2}V_{i2}, V_{i1}) \cdot e(V_{i2}, Q_i) \tag{8}$$

If it hold, node i compute the following set of session keys as

$$\begin{aligned} SK_1 &= e(r_{j1}V_{i1}, Q_i + Q_j) \\ SK_2 &= e(r_{j1}V_{i2}, Q_i + Q_j) \\ SK_3 &= e(r_{j2}V_{i1}, Q_i + Q_j) \\ SK_4 &= e(r_{i2}V_{i2}, Q_i + Q_j) \end{aligned}$$

The protocol is presented as

Node i		Node j
Select r_{i1} and r_{i2} randomly Where $r_{i1}, r_{i2} \in \mathbb{Z}_q^*$ Computes $V_{i1} = r_{i1} \cdot P, V_{i2} = r_{i2} \cdot P$ Computes $S_i = (r_{i1}X_{i1} + r_{i2}X_{i2}) \cdot V_{i1} + \lambda_i \cdot V_{i2} \bmod n$		
	$\{V_{i1}, V_{i2}, S_i, Cert(Q_i)\}$ $\longrightarrow$	
		Take the X_{i1} and X_{i2} from V_{i1} and V_{i2} Verify $e(S_i, P) \overset{?}{=} e(X_{i1}V_{i1} + X_{i2}V_{i2}, V_{i1}) \cdot e(V_{i2}, Q_i)$ If holds computes the Session key $SK_1 = e(r_{j1}V_{i1}, Q_i + Q_j)$ $SK_2 = e(r_{j1}V_{i2}, Q_i + Q_j)$ $SK_3 = e(r_{j2}V_{i1}, Q_i + Q_j)$ $SK_4 = e(r_{i2}V_{i2}, Q_i + Q_j)$ Select random numbers r_{j1} and r_{j2} Where $r_{j1}, r_{j2} \in \mathbb{Z}_q^*$ Computes $V_{j1} = r_{j1} \cdot P, V_{j2} = r_{j2} \cdot P$ and $S_j = (r_{j1}X_{j1} + r_{j2}X_{j2}) \cdot V_{j1} + \lambda_j \cdot V_{j2} \bmod n$
	$\{V_{j1}, V_{j2}, S_j, Cert(Q_j)\}$ $\longleftarrow$	
Take the X_{j1} and X_{j2} from V_{j1} and V_{j2} Verify $e(S_j, P) \overset{?}{=} e(X_{j1}V_{j1} + X_{j2}V_{j2}, V_{j2}) \cdot e(V_{j2}, Q_j)$ If holds computes the Session key $SK_1 = e(r_{i1}V_{j1}, Q_i + Q_j)$ $SK_2 = e(r_{i1}V_{j2}, Q_i + Q_j)$ $SK_3 = e(r_{i2}V_{j1}, Q_i + Q_j)$ $SK_4 = e(r_{i2}V_{j2}, Q_i + Q_j)$		

8 Security Analysis

In this section, we analyze the security of our proposed protocol . The security of the protocol is based on the difficulty of breaking of Elliptic Curve Discrete Logarithms. We claim that the proposed protocol is resistant against attack on data integrity of sensor node. Also this protect the known session keys, if the adversary can able to compute the previous session keys. Subsequently we prove that the protocol achieves the most important security requirement implicit key authentication and full forward secrecy.

Definition 9. *Authentication multiple key establishment protocol is said to achieve the property of data integrity, if there is no polynomial time algorithm that can alter or manipulate the transmitted messages.*

Theorem 4 *The proposed Protocol is resistant against the attack on data integrity if and only if Elliptic Curve Discrete Logarithm Problem is hard to solve(ECDLP).*

Proof : While the node i sends the sensitive data to another node j by the communication channel, the adversary tries to alter or manipulate the data and cheat the honest nodes by relying the wrong session keys. Assume that, adversary would like to compute S_i to validate the verification equation-4 for cheating node j. He can select randomly two points V_{i1} and V_{i2} and extract the x-coordinate X_{i1} and X_{i2} respectively. After that, he has to find a S_i that satisfies the Equation $e(S_iP) = e(X_{i1}V_{i1}+X_{i2}V_{i2}, V_{i1})\cdot e(V_{i2}, Q_i)$. But it is computationally infeasible for the adversary to compute S_i without the knowledge of λ_i by equation-1. To compute λ_i from the known Q_i for the adversary, it is required to solve ECDLP. Therefore it is not possible to forge a valid message to cheat node j by relying invalid common session keys. $\square$

Definition 10. *A protocol can protect the subsequent session keys from disclosing even if the previous session keys are revealed by the intendant user is called Known key security.*

Theorem 5 *It is computationally infeasible for an adversary to generate the correct session keys even if the previous keys are disclosed if and only if Elliptic Curve Discrite Logarithm and Computational Diffie-Hellman Problem are hard.*

Proof : Node i and j select fresh random number in each round of the protocol and compute S_i and S_j by using equation-1 and 2. This implies that the four generated session keys are distinct and does not depend in each round in the execution of the protocol. Even the session keys are reveled, it is not possible for the adversary to find the random numbers. Since to compute the random numbers r_{i1}, r_{12}, r_{j1} and r_{j2} from the four session keys SK_1, SK_2, SK_3 and SK_4, it is required to solve ECDLP problem. So it is computationally infeasible for an adversary to compute the long-term secret key by using equation-1 and 2 without the knowledge of these random numbers. Hence the adversary does not collect the related information to compute the later session keys. Further, all the four session keys are generated in the execution of the protocol. Let us assume that the adversary can able to collect all the four session keys SK_1, SK_2, SK_3 and SK_4 and try to derive the long-term session key SK_{ij}. The adversary may try to compute $SK_{ij} = e(\lambda_i\lambda_jP, Q_i + Q_j)$ from the respective public keys $Q_i = \lambda_iP$ and $Q_j = \lambda_jP$ of nodes i and j. In order to compute $\lambda_i\lambda_jP$, the adversary has to solve Computational Diffie-Hellman Problem. Further, the adversary tries to find the random numbers r_{i1}, r_{i2}, r_{j1} and r_{j2} using SK_1, SK_2, SK_3 and SK_4. Hence we conclude that, it is computationally infeasible to solve like this. Again λ_i and λ_j are unknown, the adversary will have no enough information to derive long-term shared session key SK_{ij} from the above equations.

$$\begin{aligned} SK_{ij} &= e(\lambda_i\lambda_jP, Q_i + Q_j) \\ &= e(\lambda_i\lambda_jP, \lambda_iP + \lambda_jP) \\ &= e(P, \lambda_iP + \lambda_jP)^{\lambda_i\lambda_j} \\ &= e(P, P)^{\lambda_i\lambda_j(\lambda_i+\lambda_j)} \end{aligned}$$

Hence all the four session keys exist in the protocol and is resistant against known key attack. $\square$

Definition 11. *An authenticated multiple Key establishment protocol provides perfect forward secrecy if the compromise of both the node's secret keys cannot results in the compromise of previously established session keys [10] [11].*

Theorem 6 *The proposed protocol provides perfect forward secrecy if and only if Elliptic Curve Discrete Logarithm Problem is hard to solve(ECDLP).*

Proof : From the above equation, Session keys are established by two random numbers and the generator of group of points on Elliptic curve. The four short-term session keys are computed as

$$\begin{aligned} SK_1 = &\ e(r_{i1}V_{j1}, Q_i + Q_j) = \ e(P,P)^{r_{i1}r_{j1}(\lambda_i+\lambda_j)} \\ SK_2 = &\ e(r_{i1}V_{j2}, Q_i + Q_j) = \ e(P,P)^{r_{i1}r_{j2}(\lambda_i+\lambda_j)} \\ SK_3 = &\ e(r_{i2}V_{j1}, Q_i + Q_j) = \ e(P,P)^{r_{j2}r_{j1}(\lambda_i+\lambda_j)} \\ SK_4 = &\ e(r_{i2}V_{j2}, Q_i + Q_j) = \ e(P,P)^{r_{i2}r_{j2}(\lambda_i+\lambda_j)} \end{aligned}$$

Long-term shared session key SK_{ij} is

$$SK_{ij} = \ e(\lambda_i\lambda_j P, Q_i + Q_j) = \ e(P,P)^{\lambda_i\lambda_j(\lambda_i+\lambda_j)}$$

Hence the proposed protocol provides perfect forward secrecy.□

9 Conclusion

In this article we have proposed a novel construction of Multiple key establishment protocols for WSNs which have the memory space required for each node is fixed. Also here the sensor node can establish secure communications with other adjacent nodes by protecting the subsequent session keys even if the previous session keys are revealed by the intendant user. The protocol is secure against perfect forward key secrecy and modification attack. The proposed protocol provides two significant advantages as (1) the memory space required for each node is fixed. So it is compatible for implementation in WSN, (2) the sensor node can establish secure communications with other adjacent nodes by protecting the subsequent session keys.

References

1. D. Hankerson, A. Menezes, AND S. Vanstone Guide to Elliptic Curve Cryptography, Springer, 2004.
2. V.S. Miller Use of elliptic curves in cryptography, in: Proceedings of the Advances in Cryptology - Crypto'85, New York, USA, 1985, pp. 417-426.
3. M. Bellare and P. Rogaway Entity Authentication and Key Distribution. In proceedings of Crypto 1993, LNCS 773, pp. 231-249, Springer-Verlag, 1994.
4. M. Bellare and P. Rogaway Provably Secure Session Key Distribution: The Three-party Case. In proceedings of STOC 1995, pp. 57-66, ACM Press, 1995.
5. M. Bellare, D. Pointcheval, and P. Rogaway Authenticated Key Exchange Secure Against Dictionary Attacks. In proceedings of Eurocrypt 2000, LNCS 1807, pp. 139-155, Springer-Verlag, 2000
6. N. Koblitz Elliptic curve cryptosystem, Mathematics of Computation 48 (1987), 203-209.
7. N. Koblitz. *A course in Number Theory and Cryptography ,2nd edition* Springer-Verlag-1994
8. K. H Rosen *"Elementary Number Theory in Science and Communication"*, 2nd ed., Springer-Verlag, Berlin, 1986.
9. A. Menezes, P. C Van Oorschot and S. A Vanstone *Handbook of applied cryptography. CRC Press, 1997.*
10. J.Kar and B.Majhi An Efficient Password Security of Three Party Key Exchange Protocol based on ECDLP" 12th International Conference on Information Technology 2009 (ICIT 2009), Bhubaneswar, India, Tata McGrow Hill Education Private Limited, pp75-78, 2009.
11. J.Kar and B.Majhi An Efficient Password Security of Multiparty Key Exchange Protocol based on ECDLP" International Journal of Computer Science and Security (IJCSS) , Malyasia, Vol.3 (5), pp 405-413, Nov 2009.
12. J.Kar and B. Majhi A Secure Two-Party Identity Based Key Exchange Protocol based on Elliptic Curve Discrete Logarithm Problem", Journal of Information Assurance and Security, USA Vol-5(1), pp 473-482, 2009.
13. Aumann,Y.and Rabin M. Authentication, enhanced security and error correcting codes,Advances in Cryptology - Crypto'98, LNCS, 1462, 299-303.