A NOVEL SECURITY PROTOCOL FOR WIRELESS SENSOR NETWORKS BASED ON ELLIPTIC CURVE SIGNCRYPTION

Anuj Kumar Singh[1] and B.D.K.Patro[2]

[1]Dr. A.P.J. Abdul Kalam Technical University, Lucknow, (U.P), India
[2]Rajkiya Engineering College, Kannauj, (U.P.), India

ABSTRACT

With the growing usage of wireless sensors in a variety of applications including Internet of Things, the security aspects of wireless sensor networks have been on priority for the researchers. Due to the constraints of resources in wireless sensor networks, it has been always a challenge to design efficient security protocols for wireless sensor networks. An novel elliptic curve signcryption based security protocol for wireless sensor networks has been presented in this paper, which provides anonymity, confidentiality, mutual authentication, forward security, secure key establishment, and key privacy at the same time providing resistance from replay attack, impersonation attack, insider attack, offline dictionary attack, and stolen-verifier attack. Results have revealed that the proposed elliptic curve signcryption based protocol consumes the least time in comparison to other protocols while providing the highest level of security.

KEYWORDS

Wireless Sensor Network, Security,Protocol, Signcryption, Elliptic Curve

1. INTRODUCTION

To monitor the harsh, hostile, or unattended environments, there is a need forhaving dedicated infrastructure which is capable of collecting the required data when needed. The Wireless Sensor Network (WSN) composed of tiny sensors distributed spatially, is such an infrastructure which is used to monitor and gather data about the physical situations of an environment or location. WSN collects the data using wireless sensors also called as nodes. Generally, the sensor node comprises of a microcontroller, analog-to-digital converter (ADC), transceiver,powersource, and sensors.The schematic diagram of a wireless sensor node architecture has been depicted in Figure 1 (a).The role of the microcontroller is to processes the collected data and to regulate the functions of the other elements of the sensor node. The transceiver is equipped with an antenna and performs the functions of both the transmitter and the receiver. Two categories of memory are used in a sensor node, the user memory which is used to store user data, and the program memory which is used to program the device. Sensor node operates on power and thus a power source, commonly a battery is deployed to supply power to the sensor node. Sensor nodes are also equipped with sensors, which are hardware devices capable of measuring the change in the physical conditions of surroundings like temperature, pressure, etc. ADC is deployed to convert analog values to the digital signals.

DOI: 10.5121/ijcnc.2019.11506

The architecture of a WSN typically consists of three components - a gateway, sensor nodes, and the user [1]. The sensor nodes and gateway are connected through wireless links, and the data among them is passed using radio signals. Gateway also known as a sink, gathers all the data and transmits this data to the user through the Internet or a network. The basic architecture of a WSN has been demonstrated in Figure 1(b). Except for the gateway and the sensor node, the user is another party involved in the communication. The communication between the gateway and the sensor node is highly insecure because of the usage of wireless links. Due to the capability of monitoring, sensing, and controlling, WSNs are being applied in the areas including environmental monitoring, medical, military, healthcare, industry, robotics and many more. Furthermore, with the evolution of the Internet of Things (IoT), application of wireless sensors have grown to a large scale, since wireless sensors are an important component of IoT

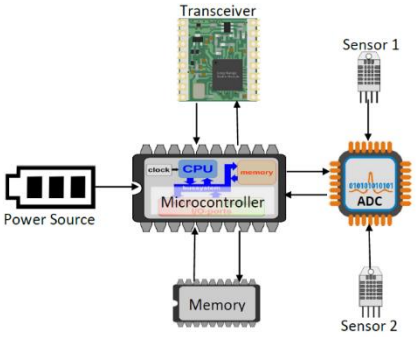

Figure 1(a). Sensor Node Architecture

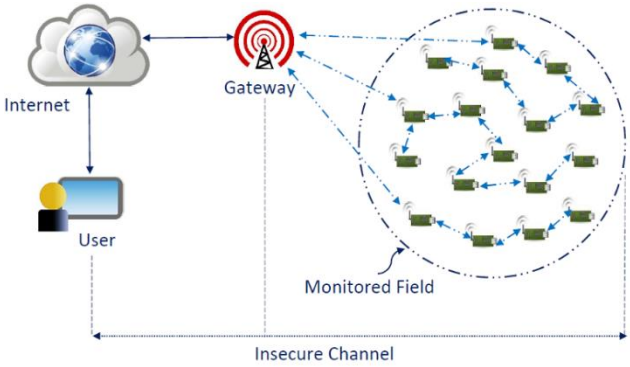

Figure 1(b). Architecture of a WSN

1.1. Security Requirements of WSN

Besides confidentiality, non-repudiation, authentication, and integrity, which are the major security features for any system, WSNs require the implementation of some more security attributes, since they function in the wireless medium. It has been pointed out by Lopez et al. [2] that for WSNs authorization, availability, data freshness, forward security, and self-organization must be efficiently implemented in addition to confidentiality, authentication, non-repudiation, and integrity. The security features that must be satisfied by a WSN are listed below.

Confidentiality: The data gathered from sensor nodes must be sent securely to the gateway and the user.

Integrity: It is the assurance that the data collected by sensor nodes has not been altered in transit.
Mutual Authentication: User, gateway, and the sensor nodes must authenticate each other before transmitting any data.

Session Key Establishment: Upon successful mutual authentication by all the parties, the session key must be secretly established between the communicating parties.

Non-Repudiation: It is the assurance that any party in communication cannot deny after sending or receiving the data.

Availability: Each wireless sensor node must be able to send the data all the time. Therefore, all the sensor nodes must be kept secure from heavy computations and denial of service attacks.
Authorization: A sensor node must be permitted to perform the computations assigned to it in the network only if, it is authorized to do so.

Data Freshness: Every node must collect data without delay and the data must not be forged.
Self-Organization: The sensor nodes must be independently able to organize and heal themselves in abnormal or problematic conditions.

Forward Security: When a new node enters the WSN as a fresh node or in the position of the old node, it cannot obtain the previous messages. Similarly, when a node exits the network it is infeasible for it to get the future messages.

1.2 WSN Security Challenges

Designing efficient security protocols for WSNs have been a continuous challenge due to the following technical limitations.

- Less Computational Capacity - Wireless sensor nodes typically possess a processing capacity of few MIPS, RAM of few 100s KB and flash memory of less than 1MB. Due to the less computing capability of wireless sensors, designing and implementing security schemes which satisfy all the required security functionalities is very challenging.
- Limited Power Supply – Since sensor nodes operate on limited battery power, the security mechanisms should be selected and implemented such that they avoid heavy computations.
- Unreliable Communication – The data is sent by the sensor nodes through wireless channels which are unreliable medium and are vulnerable to many threats and attacks. This requires the implementation of strong security schemes which thwart the attacks on WSN.

These limitations enforce the two major challenges in securing WSNs – threats and the attacks on WSNs, and difficulties in implementing efficient security measures to counter these threats and attacks. Dhakne and Chatur [3] have presented an exhaustive survey over attacks made on WSNs and divided them into five categories – attacks on authentication, attacks on privacy, attacks based on perspectives, attacks on layers, and other attacks. The detailed classification of attacks on WSNs has been publicized in Figure 2.

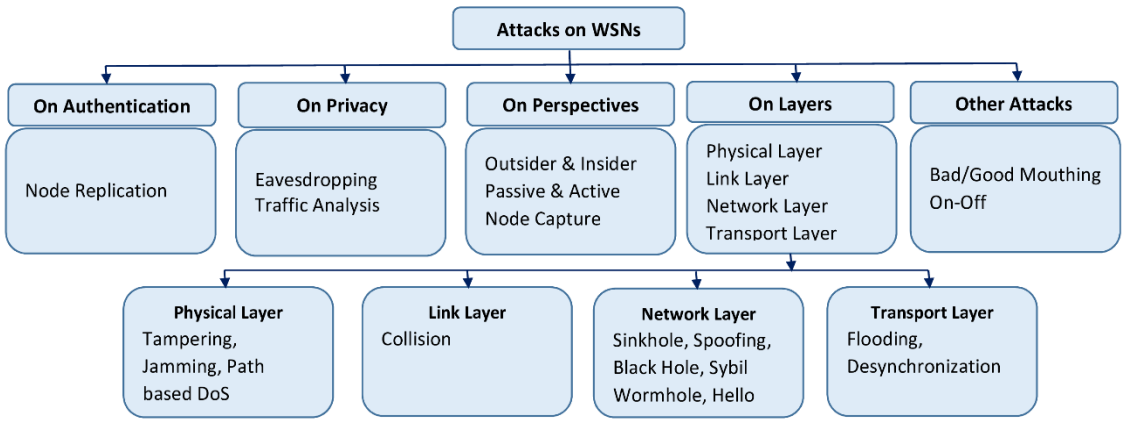

Figure 2. Taxonomy of attacks on WSN

Since wireless sensors are resource-constrained devices, it hasalways challenging to design and implement efficient security schemes for WSNs satisfying all the security requirements while simultaneously providing resistance to all the threats and attacks.

2. RELATED WORK

Various security protocols for WSNs based on different cryptographic systems with different level of security have been proposed by different authors. But the recent focus of the researchers has been on designing Elliptic Curve Cryptography (ECC) based security mechanisms for WSNs, since ECC based solutions are suitable for applications involving low computing power devices like wireless sensors [4]. Therefore in this section, the security protocols for WSNs based on elliptic curves proposed by different authors have been highlighted.

Choi et al. [5] presented an ECC based authentication mechanism for WSNs which addressed the security flaws of session key attacksensor energy exhausting attack, and stolen smart card attack, in the protocol given by Shi and Gong [6].

Wu et al. [7] designed a mutual authentication scheme for the mobile network, which provides forward security and resistance against insider attack, de-synchronization attack, forgery attack, replay attack, and known-key attack.

Amin et al. [8] suggested a 3-factor key agreement and authentication scheme which was an improvement over the protocol developed by Farash et al. [9]. They stated that their protocol provides additional security features of identity change and smartcard revocation phases, at the same time protecting from stolen smart-card attack, user impersonation attack, session-specific attack, and password guessing attack.

Y.H. Park and Y. Park [10] suggested a 3-factor ECC based key-agreement and biometric authentication scheme which provides user anonymity, forward security, intraceability, mutual authentication, secure password update and can resist from stolen smart card attack, user impersonation attack, replay attack, man-in-the-middle attack, and off-line password guessing attack.

Later, Jiang et al. [11] proved that scheme of Amin et al. [8] is prone to lost smart card attack, KSSTI (known-session specific temporary information) attack, and tracking attack. They also designed a Rabin Cryptosystem based 3-factor authentication and key establishment protocol which overcome all the weaknesses of the protocol given by Amin et al.

Jung et al. [12] exposed that the protocol given by Chang et al. [13] cannot protect against password guessing, session key compromise, and user impersonation. Furthermore, Jung et al. pointed out that Chang's protocol puts a high computational load on the gateway. They also designed an anonymous key establishment and authentication scheme for WSNs overcoming security flaws of Chang et al. scheme while consuming less computational cost.

Wang et al. [14] proved that Jung's [12] protocol is exposed to impersonation attack and offline dictionary attack. They also revealed that Park & Park's [10] scheme was unable to satisfy user anonymity and was also weak against an offline dictionary attack. Then they proposed a 3-factor user authentication scheme for WSNs which overcame the weaknesses of the schemes given by Jung et al. and Park et al.

Li et al. [15] revealed that Jiang's [11] protocol lacks user-friendliness, is inapplicable to IoT environment, and is vulnerable to KSSTI attack. They designed an anonymous 3-factor authentication scheme for WSNs which can be used for the IoT environment. Moreover, they claimed that this scheme provides all the necessary security functionalities and is computationally efficient.

Recently, Zhang et al. [16] suggested an elliptic curve-based key exchange and authentication mechanism for WSNs which provides mutual authentication, key establishment, key privacy, user anonymity and resistance from off-line dictionary attack, replay attack, insider attack, impersonation attack, stolen verifier attack, and compromised sensor node attack. But, this scheme consumes more total computational time and puts the high computational load on the gateway, in comparison to the other similar protocols.

3. A BRIEF REVIEW OF ZHANG'S PROTOCOL

In this section, a brief review of Zhang's protocol has been presented. As mentioned in the related work discussed in section 2, the protocol of Zhang et al. [16] enforces heavy computations on the gateway and consumes more total computational time. The three parties involved in the protocol are the user$U$, gatewayGWN, and the sensor nodeS_i. The protocol has been divided into three phases – first is the setup phase, second is the registration phase, and last is the authenticated key exchange phase. In the setup phase, global parameters for the protocols are selected. If a user U wants to collect the data from the sensor node S_i then it has to register with the gateway nodeGWN. Moreover,each sensornode$S_i$also registers with the gatewayGWN. User registration and sensor node registration is done in the registration phase using a secure channel. Here, only the computations done by the gateway node have been analyzed. The detailed protocol can be referred from [16]. The steps carried out by the gateway in Zhang's protocol are given below. The symbols used in these steps are:

$label$ – session label; X, T, c_1, s_m, s_a- values computed by the user; S_{GN} – secret key of the gateway node; P – the base point of elliptic curve;H_1, H_3, H_4 - hash computations; GN – gateway identity; S_i – sensor node identity; $Y, Auth_{S_i}$- values computed by the sensor node; $T_{GN}, {T_{S_i}}^*, T_{S_i}$ – timestamps; ΔT – expected transmission delay; σ_{GN} - signature of r_{GN} signed by the gateway.

1. Upon receiving the message $\{label, X, T, c_1, s_m, s_a\}$ from the user, the gateway node computes the following:

(i) $V = S_{GN}T$

(ii) ${R_3}^* = s_aP - c_1V - s_mT$

(iii) ${c_1}^* = H_1(P, T, {R_3}^*, X, label)$

(iv) Checks if $c_1 = {c_1}^*$

(v) $K_{(GN,S_i)} = H_3(GN, S_i, S_{GN})$

(vi) $Auth_{GN} = H_4(K_{(GN,S_i)}, X, lable, T_{GN})$

2. Upon receiving the message $\{S_i, Y, T_{S_i}, Auth_{S_i}\}$ form the sensor nodeS_i, the gateway performs the following computations:

(i) Checks if ${T_{S_i}}^* - T_{S_i} \leq \Delta T$

(ii) Computes $K_{(GN,S_i)} = H_3(GN, S_i, S_{GN})$ and verify the validity of$Auth_{S_i}$.

(iii) Computes $r_{GN} = H_1(label, X, T, c_1, s_m, s_a, Y)$

(iv) Creates the signature $\sigma_{GN} = sign_{S_{GN}}(r_{GN})$

The most time-consuming operation in elliptic curve based security schemes is the elliptic curve point multiplication (ECPM) operation. Moreover, the time consumed by all the operations is very small as compared to the ECPM operation. Therefore, the count of ECPM operations can be used for the analysis of computational time. In Zhang's protocol, the gateway node GWN is required to execute four ECPM operations out of which one ECPM operation is executed in the step (i) of point no. 1 and three ECPM operations are executed in step (ii) of point no. 1. No ECPM operation is executed in the computations mentioned in point no.2. A total of ten ECPM operations are executed by Zhang's protocol. This means that the gateway node GWN bears the 40 % computation overhead of the whole protocol, which is the major drawback of Zhang's protocol. The computational overhead on the gateway node GWN as well as the total computational time of the protocol can be reduced by using elliptic curve based signcryption which has been discussed in the next section.

4. PRELIMINARIES

This section provides an introduction to the basic concepts which have been applied in designing the proposed protocol.

4.1. Mathematics of Elliptic Curve

For cryptographic applications, the elliptic curves defined by Weierstrass Equation $y^2 = x^3 + Ax + B$ over finite field F_q are used, where $A, B \in F_q$ are constants such that$4A^3 + 27B^2 \neq 0$. The main reason for using the Weierstrass Equation for defining elliptic curve is that, frameworks for implementation are available in many programming languages including java and python. An elliptic curve symbolized by E over F_q is the set of all the points (x, y) along with a distinct point Oknown as the point on infinity. These points are represented as:

$$E(F_q) = \{(x, y) \in F_q \times F_q : y^2 = x^3 + Ax + B\} \cup \{O\}$$

The operation and rules for elliptic curve $E(F_q)$ are given below.

- Identity Element – For each point $R \in E(F_q)$, there subsists an identity element O such that
$$O + R = R + O = R$$
- Point Addition – Let $Q, R \in F_q$ be the two points on elliptic curve$E$, where $Q = (x_1, y_1)$ and $R = (x_2, y_2)$ and$Q \neq \pm R$. The addition of $QandR$ is defined as$Q + R = (x_3, y_3)$, where x_3 and y_3 are given by:
$x_3 = \lambda^2 - x_1 - x_2andy_3 = \lambda(x_1 - x_3) - y_1$
with$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$ if $Q \neq R$ and $\lambda = \frac{3{x_1}^2 + A}{2y_1}$ if $Q = R$
- Point Multiplication – Let $Q \in E(F_q)$ and an integer$k$. The multiplication of point Q with k is defined by$kQ = Q + Q + \cdots + Q(ktimes)$.
- Negative – Let $Q = (x, y) \in E(F_q)$ then the negative of point Q is defined as $-Q = (x, -y)$ and$Q + (-Q) = O$. Moreover, $-O = O$.

4.2. Strength of Elliptic Curve Cryptography

The strength of the elliptic curve-based cryptosystem is ensured by the three computationally hard problems given below. An elliptic curve $E(F_q)$ has been considered in the definition of these problems.

1. Elliptic Curve Based Discrete Logarithmic Problem (ECDLP) – For known two points$Q, R \in E(F_q)$, it is computationally infeasible to get an integer$k$ so that$R = kQ$ [17].
2. Elliptic Curve Based Diffie-Hellman Problem (ECDHP) – Given a point$Q \in E(F_q)$, and consider two other points $R = aQ$ and $S = bQ$ on the same elliptic curve$E(F_q)$, where$a, b \in Integer$. Determining a point $T = abQ$ is computationally hard [18].
3. Elliptic Curve Based Decision Diffie-Hellman Problem (ECDDHP) - Given a point$Q \in E(F_q)$, and consider three other points$R = aQ$,$S = bQ$ and$T = cQ$. It is computationally infeasible to conclude that if$T = abQ$ [19].

4.3. Overview of Signcryption

Signcryption which integrates confidentiality and authentication in a single-phase logically was proposed by Y. Zheng [20]. Zheng showed that encryption consumes 50% less time in computation and 85% less bandwidth than the signature-then-encryption process which is traditionally followed. Y. Zheng and H. Imai [21] applied elliptic curves in signcryption and proposed the first signcryption mechanism based on the elliptic curve. They proved that elliptic curve signcryption consumes 58% less time and 40% less communication bandwidth than the signature-then-encryption mechanism based on the elliptic curve. For low computing power devices (LCPDs) it is wise to use elliptic curve signcryption schemes, since it saves a huge amount of computational time and communication bandwidth, while also providing many security attributes including authentication, secure key establishment, confidentiality, non-repudiation, integrity, unforgeability, and forward security [4]. The elliptic curve signcryption scheme proposed by Y. Zheng and H. Imai [21] has been publicized in Figure 3 to provide a glimpse that how elliptic curves can be used in designing signcryption schemes. The process of signcryption is carried out in three phases – first is the initialization phase, second is the signcryption phase and, last is the un-signcryption phase. In the initialization phase, the global public parameters and key pairs are selected. Signcryption phase implements confidentiality and signature functionality. In the un-signcryption phase decryption and signature verification is carried out. In Figure 4 the sender is Alice and the receiver is Bob, Msg is the message sent by the Alice to the Bob, and SECDSS is Shortened Elliptic Curve Digital Signature Standard.

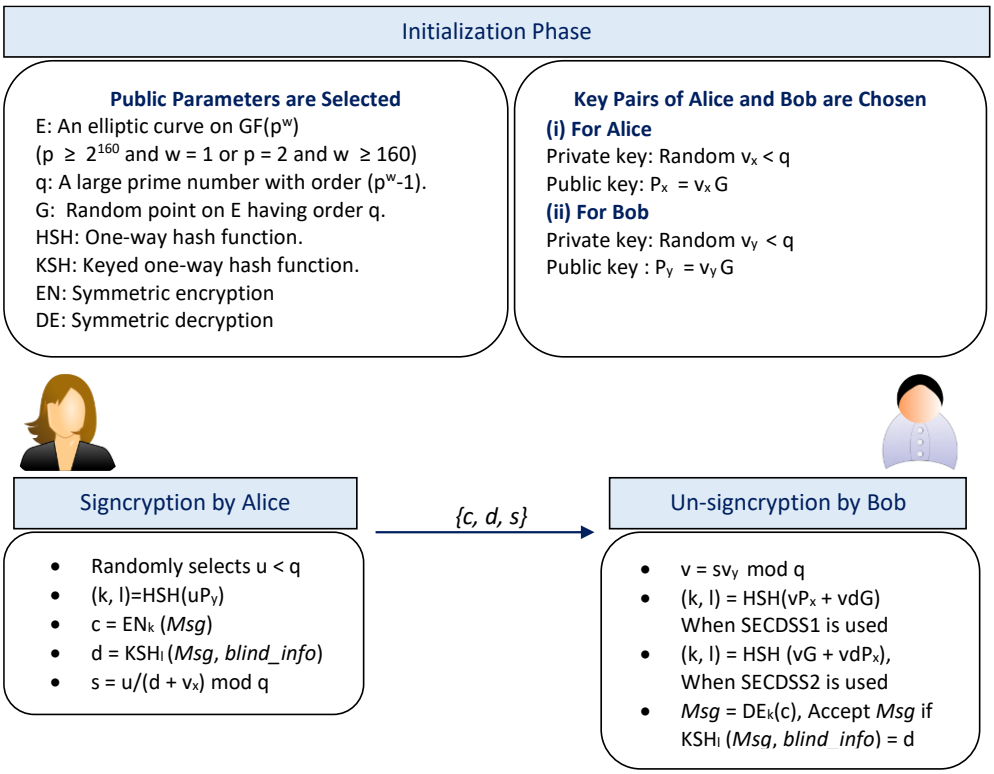

Figure 3. Elliptic Curve based Signcryption by Zheng and Imai [21]

5. PROPOSED PROTOCOL

In this section, a novel elliptic curve signcryption based security protocol for wireless sensor networks has been proposed and elucidated in detail. The proposed security protocol presented here has three phases – first is setup phase, second is the registration phase, and the third is the signcryption and key-establishment phase. The symbols and notations utilized in the proposed protocol are mentioned in Table 1.

5.1. Setup Phase

In the setup phase, global parameters for the system are selected by the gateway. The gateway also generates its private and public keys in this phase. The steps of the setup phase are:

1. The gateway selects an elliptic curve $E: y^2 = x^3 + Ax + B$over the finite field $F_q$with curve parameters $\{q, A, B, G, n\}$ satisfying $4A^3 + 27B^2 \neq 0$ and having point at infinity$O$.
2. The gateway selects a private key $v_G \in Z_n$ and generates its public key$P_G = v_G G$.
3. The gateway also selects the hash function$H: \{0,1\}^* \rightarrow \{0,1\}^l$.
4. All the public parameters $\{Fq, E(Fq), q, A, B, G, n, P, G, H\}$ are made available to all the parties in the WSN.

Table 1. Notations and symbols used in proposed protocol.

Symbol	Notation
F_q	Finite prime field of size q
E	Elliptic curve over F_q
A,B	Curve parameters for E
G	Generator of E with order n
$q.n$	Two large prime numbers
ID_U	User identity
ID_G	Gateway identity
PW_U	Password of the user
H	Hash computation
T_G	Time stamp of the gateway
T_{Si}	Time stamp of the sensor node
$\oplus$	Exclusive OR
K	Established shared key
T	Current timestamp
t	Average transmission delay

5.2. Registration Phase

A user willing to collect the data from a sensor node S_i, has to register itself to the gateway. Moreover, the sensor node S_i has also to register with the gateway node. The registration of the user to the gateway has been shown in Figure 4. All the messages in the following steps of the registration phase are sent using a secure channel.

1. The user selects its identity and password as $\{ID_u, PW_u\}$.
2. User computes $P_u = ID_u G$ and transmits the message $\{P_u\}$to the gateway.
3. On receiving the public key $\{P_u\}$ from the user, the gateway computes the following:
   - Generates the key $K_{GU} = H(v_G P_u)$
   - Creates the ciphertext$c_1 = E_{KGU}(ID_G)$
   - Calculates the intermediate value$r_1 = H(c_1 \oplus K_{GU})$
   - Calculates another intermediate value $w_1 = v_G / r_1$
   - Computes $T_1 = r_1 G$

   The gateway sends the signcrypted text $\{c_1, T_1, P_G\}$ to the user.
4. Upon receiving $\{c_1, T_1, P_G\}$from the gateway, the user computes${K_{GU}}^* = H(ID_u P_G)$,$d_1 = D_{K_{GU}{}^*}(c_1)$,${r_1}^* = H(c_1 \oplus {K_{GU}}^*)$, and${T_1}^* = {r_1}^* G$. If ${T_1}^* = T_1$then the gateway is successfully authenticated by the user, and then the user computes $H(PWu)$and$c_r = d_1 + H(PWu)$. Finally the user saves the credentialc_r.

A sensor node S_i willing to register itself to the gateway sends the request containingID_{Si} to the gateway. On receiving the request from sensor Si gateway computes a secret key $K_{GS_i} = H(ID_{Si}, ID_G, v_G)$and send $K_{GS_i}$to the sensor node.

5.3. Signcryption and Key Establishment Phase

In this phase mutual authentication, confidentiality, and key establishment functionalities are implemented. The user, gateway, and the sensor node authenticate each other. After the successful execution of all the steps of this phase, a secret session key is generated and distributed securely between the sensor node and the user. Signcryption and key establishment phase has been demonstrated in Figure 5. The steps are given below.

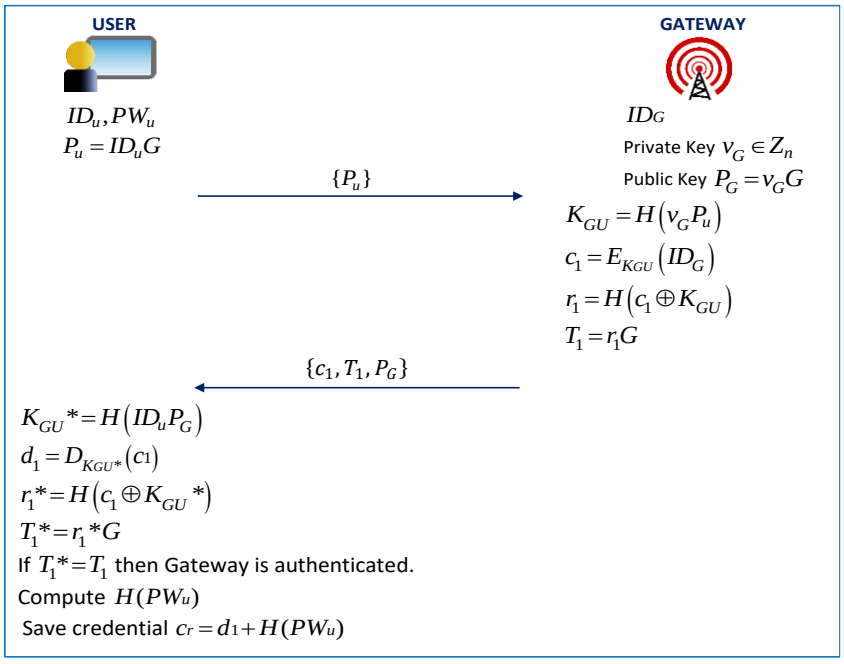

Figure 4. Registration of user with the gateway.

1. The user selects a private number $x \in Z_n$ and computes$X = xG$. It also denote the session label$L = (ID_G, IDs_i)$. The user then performs the following computations -
   - Retrieve $d_1 = c_r - H(PW_u)$
   - Compute the key $K_{UG} = H(ID_U P_G)$
   - Compute the ciphertext$c_2 = E_{KUG}(d1)$
   - Compute $r_2 = H(c_2 \oplus K_{UG})$
   - Calculate$w_2 = ID_U / r_2$, and$T_2 = r_2 G$.

The user combines the signcrypted text $\{c_2, T_2, w_2\}$with$\{L, X, P_U\}$ and sends the message $\{L, X, P_U, c_2, T_2, w_2\}$ to the gateway.

2. Upon receiving the message $\{, X, P_U, c_2, T_2, w_2\}$from the user, the gateway first generates the key as $K_{UG}{}^* = H(v_G w_2 P_U)$ and decrypt c_2 as$c_2{}^* = D_{K_{UG}{}^*}(c_2)$. The gateway checks whether $c_2{}^* = ID_G$ or not. If not then it terminates the session and if yes then it computes $r_2{}^* = H(c_2 \oplus K_{UG}{}^*)andT_2{}^* = r_2{}^* G$. If $T_2 = T_2{}^*$ then the user is authenticated by the gateway. The gateway then computes the hash code of the secret key as$A_{GS} = H(K_{GS_i})$, records the timestampT_G and sends the message $\{L, X, T_G, A_{GS}\}$to the sensor node.

3. Upon receiving the message $\{L, X, T_G, A_{GS}\}$from the gateway, the sensor node $S_i$checks if$T - T_G \leq t$, where T is the present time stamp and t is the average transmission delay. If it is true then node $S_i$first verifies the correctness of A_{GS} by computing the hash code$H(K_{GS_i})$. If $A_{GS}$is correct then it selects private number$y \in Z_n$, computes$Y = yG$ and records the current timestampT_{S_i}. It also computes$A_{SU} = H(L, K_{GS_i}, X, Y, T_G, T_{S_i})$, the shared secret key $K = yX$ with the user, and the session key$S_k = H(L, X, Y, K)$. The nodeS_i sends the message $\{ID_{S_i}, Y, T_{S_i}, A_{SU}\}$ to the gateway.
4. When the message $\{ID_{S_i}, Y, T_{S_i}, A_{SU}\}$ is received by the gateway it checks if$T - T_{S_i} \leq t$, where T is the present time-stamp and t is the average transmission delay. If it is true then gateway verifies the correctness of A_{SU} by computing the hash code$H(L, K_{GS_i}, X, Y, T_G, T_{S_i})$, if A_{SU} is found correct then the gateway computes$c_3 = E_{K_{UG}^*}(T_2^*)$. The gateway then sends the message $\{Y, L, c_3\}$ to the user.
5. Upon receiving $\{Y, L, c_3\}$ from the gateway, the user computes $c_3^* = E_{K_{UG}}(T_2)$and if $c_3^* = c_3$ then it authenticates the gateway. It computes the shred secret key $K = xY$and session key$S_k = H(L, X, Y, K)$.

The established shared key K between the user and the sensor node S_i can be used for the upcoming communication.

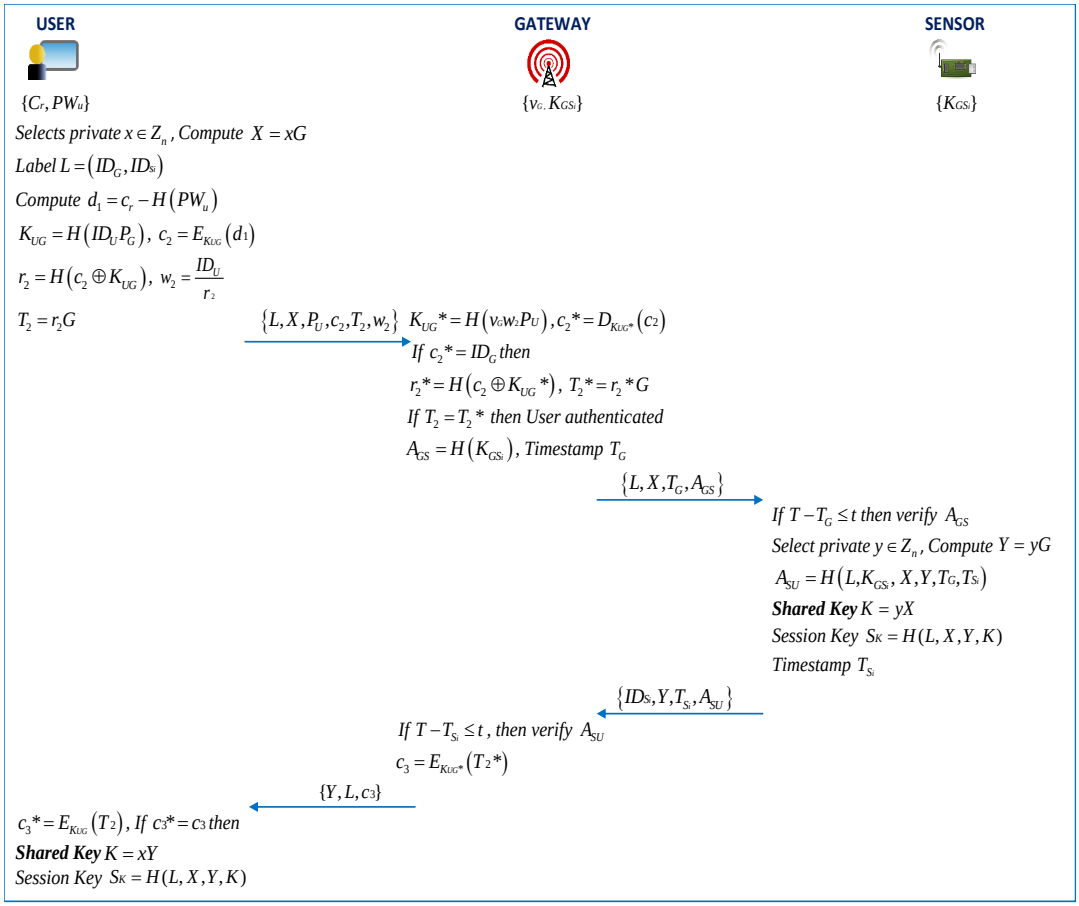

Figure 5. Signcryption and key establishment phase of the proposed protocol.

6. ANALYSIS OF SECURITY FUNCTIONS OF THE PROPOSED PROTOCOL

In this section of the paper, an analysis of the security functionalities provided by the proposed protocol has been carried out. The two dimensions of analyzing the security functionalities are, the security attributes satisfied by the proposed protocol and the resistance provided from different attacks. The following reasonable assumptions have been considered to sustain security analysis.

A1: Secure channel is used for registration of the sensor node as well as the user, to the gateway.
A2: An adversary can obtain common system parameters from a corrupted node.
A3: The private number $x$selected by the user and the private number y selected by the sensor node are fresh for every session.
A4: The encryption algorithm is strong enough that an adversary is unable to decode the ciphertext.
A5: Given R and$Q$, the adversary is unable to compute $p$in $R = pQ$, due to the strength of ECDLP.

6.1. Analysis of Security Attributes

The proposed elliptic curve signcryption protocol satisfies anonymity, confidentiality, secure key establishment, mutual authentication, key privacy, untraceability and forward security.

6.1.1.User Anonymity

User identity must be kept secret because if it is exposed then any unauthorized party can trace the login pattern of the user [12]. In the proposed elliptic curve signcryption protocol, the identity of the user is kept secret and not transmitted in any of the messages. The user's public key P_U is transmitted, and according to assumption A5, due to the strength of ECDLP the adversary cannot find ID_U given P_U and G. Therefore, the proposed protocol provides strong user anonymity.

6.1.2.Confidentiality

The four messages have been exchanged in the signcryption and key establishment phase of our protocol. The very first message is $\{L, X, P_U, c_2, T_2, w_2\}$ in which the components $X, P_U, c_2, T_2 and w_2$ contain the secret information. Retrieving secret values of $ID_U, x,$ and r_2 from $X, P_U,$ and T_2 respectively is infeasible due to the security of ECDLP, as mentioned in the assumption A5. The ciphertextc_2 cannot be decoded by an adversary without knowing the keyK_{UG}. Moreover, to deduceK_{UG}, the adversary needs identity $ID_U$of the user, which cannot be known as the proposed protocol provides user anonymity. The component w_2 is obtained by dividing the ID_U byr_2 which are privately generated. The second message is $\{L, X, T_G, A_{GS}\}$ in which X and A_{GS} covers secret values$x,$ and K_{GS_i} respectively. The secret x cannot be obtained from X due to the security of ECDLP and K_{GS_i} cannot be obtained fromA_{GS}, due to the property of random oracles. The third message is $\{ID_{Si}, Y, T_{Si}, A_{SU}\}$ in which the components Y and A_{SU} protects the secret y and K_{GS_i}respectively since, the secret y cannot be obtained from Y due to the strength of ECDLP and K_{GS_i} cannot be obtained form A_{SU} due to the property of random oracles. The fourth message is $\{Y, L, c_3\}$ which contains the components Y and c_3 protecting secret information. Again, the confidential information in Y and c_3 is secure as per assumptions A5 and A4 respectively. Therefore, the proposed protocol provides confidentiality of secret information.

6.1.3. Secure Key Establishment

In our protocol, after executing all the steps the key K is generated and shared securely between the sensor and the user. In establishing the secure key, the values X and Y are transmitted between the user and the sensor. As per assumption A5, an adversary cannot obtain private values x and y from X and Y respectively. Hence, the protocol successfully achieves a secure key establishment between the sensor node and the user.

6.1.4. Key Privacy

The private keys x and y of the user and the gateway respectively along with the shared key K established in the protocol, are kept secret and adversary cannot access them. As per assumption A5, an adversary cannot obtain private values x and y from X and Y respectively, and in turn cannot generate key$K$. Thus, the proposed signcryption based protocol provides key privacy.

6.1.5. Mutual Authentication

The proposed signcryption based protocol implements mutual authentication between the two pair of parties, first the user and the gateway, second the gateway and the sensor node.

In mutual authentication between the gateway and the user, the user is authenticated by the gateway if$T_2 = {T_2}^*$, where $T_2$is the authentication information sent by the user and ${T_2}^*$ is computed by the gateway. Similarly, the gateway is authenticated by the user if${c_3}^* = c_3$, where${c_3}^*$ is computed by the gateway and c_3 is computed by the user.

In mutual authentication between the sensor node and the gateway, the gateway computes $A_{GS} = H(K_{GSi})$ and sends A_{GS} to the sensor node. Upon receiving A_{GS} from the gateway, the sensor node computes${A_{GS}}^*$ which is the hash code of the shared key K_{GSi} stored with it, and if${A_{GS}}^* = A_{GS}$then the gateway is successfully authenticated by the sensor node. Similarly, the sensor node sends $A_{SU} = H(L, K_{GSi}, X, Y, T_G, T_{S_i})$ to gateway, and upon receiving $A_{SU}$the gateway then verifies the correctness of A_{SU} by computing the hash code of $\{L, K_{GS_i}, X, Y, T_G, T_{S_i}\}$ and then authenticates the sensor node. In this manner the protocol achieves mutual authentication between the two pair of parties.

Furthermore, the authentication data $T_2, c_3, A_{GS}$and A_{SU}generated in the process of mutual authentication is unforgeable. The authentication dataT_2 depends upon r_2 which in turn depends upon ID_U which is kept secret. In order to forgec_3 the adversary needs ${K_{UG}}^*$ which depends upon random private secret v_G of the gateway. Finally,$A_{GS}$and A_{SU} are the hash codes of the key K_{GSi} which is shared between the sensor and the gateway over a protected channel. Therefore, the authentication data generated in all the messages of the protocol is unforgeable.

6.1.6. Forward Secrecy

Even if the adversary somehow obtains the secret key$K$, it cannot get the messages sent in the past sessions since the private values of x and y selected randomly by the user and the sensor respectively are fresh for every session. Moreover, if a sensor node joins the network in place of some other one then it cannot get the past messages due to unavailability of private values x and y of past sessions. Thus the proposed protocol provides forward security.

6.1.7. Untraceability

It is the assurance that an opponent cannot trace the sessions of the user by analyzing the messages in the protocol. In the proposed protocol private random number x is used, which is freshly generated in every session. This makes the user to use different values of X, c_2, T_2 and w_2 for different sessions in its messages. Thus, the proposed protocol satisfies untraceability.

6.2. Analysis of Resistance from Attacks

The security protocol for WSNs must be able to thwart the attacks attempted over the WSN system. In this subsection, the strength of the proposed WSN protocol from different attacks has been analyzed. The following adversary model given by Wang et al. [14] has been considered in this analysis.

1. An $Adversary\ A$ has the capability to intercept, modify, resend, and delete the message after eavesdropping the open communication channel.
2. An $AdversaryA$ can obtain the long term session key.
3. An $AdversaryA$ can get the password of the user or its parameters, but not both.
4. An $AdversaryA$ is capable of getting the data from an unattended sensor node.

6.2.1 Resistance from Replay Attack

In replay attack, an $AdversaryA$ records the legitimate message from a party and replays it later to the other party to produce an unauthorized effect. The analysis of the replay attack for the proposed protocol can be broken into two parts. First is the analysis of the replay attack between the user and the gateway, and the second is the analysis of the replay attack between the gateway and the sensor node.

If an $AdversaryA$ replays the past recorded message $\{L, X, P_U, c_2, T_2, w_2\}$ to the gateway then, the gateway performs the computations in step2 of signcryption and key establishment phase and sends the message $\{L, X, T_G, A_{GS}\}$ to the sensor node, which in turn performs the computations mentioned in step 3 of this phase and sends the message $\{Y, L, c_3\}$ to the $AdversaryA$. But, $AdversaryA$ cannot generate the shared key $K = xY$ correctly since, it does not know the private random number$x$ of the user. If an $AdversaryA$ replays the past message $\{Y, L, c_3\}$ to the user then also the shared key generated by the user will not match with the key generated by the sensor node since the fresh value of private random number$x$ will be used by the user in generating the shared key$K = xY$. Thus in both these cases, the shared key of the sensor node and the user will not match and the attack will fail.

When an $AdversaryA$ sends the previous recorded message $\{L, X, T_G, A_{GS}\}$to the sensor node then the sensor node will ignore the it, since time stamp has been used by the protocol to thwart the replay attack. Similarly, if an $AdversaryA$ tries to befool the gateway by sending the message $\{ID_{S_i}, Y, T_{Si}, A_{SU}\}$ then also this message will be rejected as the timestamp used in this messages is the older one.

In this way, the proposed protocol successfully thwart replay attack.

6.2.2 Resistance from Offline Dictionary Attack

Even if an attacker somehow acquires the password PW_u of the user, it is not able to create correct credential c_r to authenticate itself to the gateway and gateway will terminate the session. The credential $c_r$depends upon the identity ID_G and the private key $v_G$of the gateway, which cannot be obtained by the$AdversaryA$. So, our protocol is secure from an offline dictionary attack.

6.3.3 Resistance from Insider Attack

The user sends the message $\{L, X, P_U, c_2, T_2, w_2\}$ to the gateway. From this message, the gateway cannot extract any secret information, especially the password of the user. Therefore the proposed protocol can counter insider attack.

6.3.4. Resistance from Stolen Verifier Attack

The gateway stores the verifier table which does not reveal sensitive information i.e. even if an attacker obtains this table it cannot make any attack [16]. Hence, our protocol is safe against stolen-verifier attack.

6.3.5. Resistance from Impersonation Attack

In impersonation, an opponent pretends to be a legitimate party in to obtain confidential information from the other genuine party. In the proposed protocol for WSNs, an $Adversary\ A$ is unable to impersonate the user to the gateway, because to authenticate itself to the gateway it requires the identity of the user ID_Uwhich is kept secret. Similarly, an attacker is unable to impersonate the sensor node to the gateway since it cannot access the keyK_{GS_i}. Moreover, the attacker fails in impersonating the gateway to the user and gateway to the sensor node, since it cannot obtain v_G and K_{GS_i} respectively. So, the proposed protocol can counter impersonation attacks.

7. PERFORMANCE ANALYSIS

In this section, the performance of the proposed signcryption based WSN security protocol has been analyzed by measuring computational cost and the communication bandwidth required for the protocol. Furthermore, a comparison of costs and security functionalities has been made to show that the proposed security protocol is more efficient to the computational time as compared to the related protocols mentioned in [5, 7, 11, 14, 15 and 16]. For all the protocols it has been assumed that 160 bit ECC has been used by all the parties in the communication. In addition to this, it is presumed that the proposed protocol uses AES-128 algorithm for encryption/decryption and SHA-1 algorithm for producing the hash code of the input. The two main reasons for choosing AES-128 algorithm for encryption/decryption are – first 128-bit key will not put more computational load on the wireless sensors which is a low computing power device and second, the cryptographic support for implementing AES-128 is available in wireless sensors [4].

7.1. Analysis of Computational Time and Communication Cost

The computational time consumed by the protocol can be calculated by counting the key operations and then multiplying this count with the time taken by a single operation. On a 64-bit 2.5 GHz i7 processor having 8 GB RAM, a single elliptic curve point multiplication (ECPM),one

hash computation, and one encryption/decryption take 0.427576 ms, 0.005174 ms, and, 0.0214835 ms respectively [7]. The time consumed by other operations is very less and therefore has been ignored in the analysis. It can be observed that the time consumed by a single ECPM operation is highest in comparison to the other operations. Based on this fact, the computational time for each protocol has been calculated for all the three parties in the communication and is demonstrated in Table 2. The total computational times of all the protocols have been shown in Table 3. The graphical representation of this comparison of computational time has been shown in Figure 6 (a). The bandwidth consumed by each protocol has been computed by calculating the size of messages sent by the three parties – the user, the gateway, and the sensor, and then adding them. The comparison of bandwidth consumed by each protocol has been shown in Table 3, and a graphical representation of the same has been shown in Figure 6 (b).

Table 2. Comparison of computational time consumed by the user, the sensor, and the gateway.

Protocol	No. of operations performed												Time (ms)		
	User			Gateway			Sensor			Total					
	e	*m*	*h*	*e*	*m*	*h*	*e*	*m*	*h*	*e*	*m*	*h*	T_U	T_G	T_S
[5]	0	3	9	5	0	1	0	2	6	5	5	16	1.32929	0.11259	0.88619
[7]	1	2	11	2	0	11	1	2	4	4	4	26	0.93354	0.09988	0.89733
[11]	0	1	8	0	1	12	0	0	5	0	2	25	0.46896	0.48966	0.02587
[14]	0	2	8	1	2	11	1	2	11	2	6	30	0.89654	0.93354	0.93355
[15]	0	2	8	0	1	9	0	0	4	0	3	21	0.89654	0.47414	0.02069
[16]	0	4	4	0	4	5	0	2	1	0	10	10	1.73100	1.73617	0.86032
Proposed	***2***	***4***	***4***	***2***	***2***	***4***	***0***	***2***	***3***	***4***	***8***	***11***	***1.77396***	***0.91881***	***0.87067***

e-Encryption/Decryption, m-Elliptic Curve Point Multiplication, h-Hash Computation, N-Number of rounds, T_U-Time consumed by the user, T_G- Time consumed by the gateway, T_S- Time consumed by the sensor node

7.2. Comparison of Security Functionalities

As discussed in section 5, the proposed protocol provides mutual authentication, anonymity, confidentiality, secure key establishment, key privacy, untraceability, and forward security at the same time providing resistance against replay attack, insider attack, offline dictionary attack, stolen verifier attack, and impersonation attack. A comparative analysis of the security functions of the proposed signcryption based protocol with the protocols mentioned in [5, 7, 11, 14, 15, 16] has been shown in Table 4.

Table 3. Comparison of total computational time and bandwidth.

Protocol	**Total Time (ms)**	**Bandwidth (bits)**
[5]	2.32808	3072
[7]	1.93076	3168
[11]	0.98450	1856
[14]	2.76364	3968
[15]	1.39138	2912
[16]	4.32750	2976
Proposed	***3.56345***	***3136***

8. DISCUSSION

In this section, a brief discussion of the comparisons and results mentioned in section 6 has been made. The proposed elliptic curve signcryption protocol for WSNs has been compared with the protocols in [5, 7, 11, 14, 15 and 16]. From Table 4 it can be observed that the proposed protocol and the protocol given by Zhang et al. [16] are the only two protocols which provide all the necessary security functionalities. And from Table 3 it has been revealed that the computational time consumed by the proposed signcryption-based protocol is 3.56345 ms while the time taken by Zhang's protocol is 4.32750ms. Furthermore, the number of ECPM operations on the gateway in the proposed protocol is 2, while in the Zhang's protocol 4 ECPM operations are executed on the gateway. Due to this, the time consumed at the gateway in the proposed protocol is 0.91881 ms and the time consumed at the gateway in Zhang's protocol is 1.73617 ms. Therefore, the proposed protocol puts the less computational load on the gateway which makes it better for the WSNs. The bandwidth of the proposed protocol is slightly more than Zhang's protocol. It can be concluded that the proposed protocol is more computational time efficient as compared to all the other protocols mentioned in [5, 7, 11, 14, 15 and 16] at the same time providing a same or greater level of security. The novelty of the proposed signcryption-based security protocol is projected from the fact that it consumes least computational time at the same time satisfying all the required security functionalities.

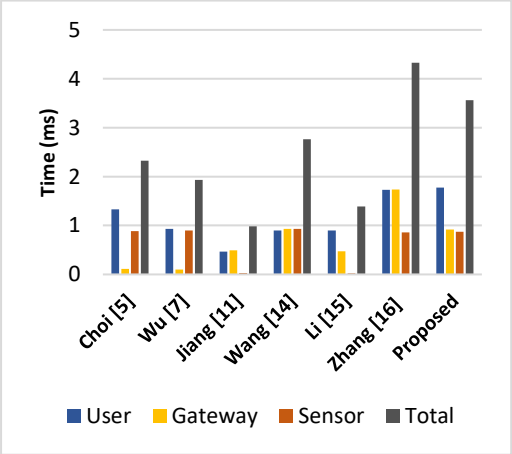

Figure 6 (a). Comparison of computational time.

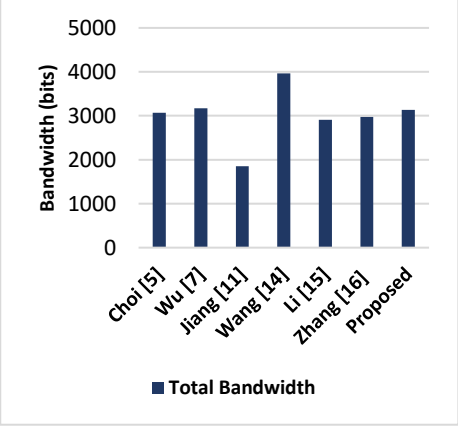

Figure 6 (b). Comparison of bandwidth.

Table 4: Comparison of security functions of different protocols.

Protocol		Security features						Resistance against attacks					
	ANY	**CNF**	**FWS**	**SKE**	**KEP**	**MUA**	**UNT**	**RPL**	**USI**	**STV**	**SNI**	**ODY**	**INS**
[5]	×	✓	✓	✓	✓	✓	×	✓	✓	×	✓	×	×
[7]	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
[11]	×	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
[14]	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
[15]	×	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
[16]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

ANY-Anonymity, CNF-Confidentiality, FWS-Forward Secrecy, SKE-Secure key establishment, KEP-Key Privacy, MUA-Mutual authentication, UNT-Untraceability, RPL-Replay attack, USI- User impersonation, STV-Stolen verifier attack, SNI-Sensor node impersonation, ODY-Offline dictionary attack, INS-Insider attack, ✓-Fulfilled, × - Not fulfilled.

9. CONCLUSION

WSNs are used widely in many critical applications, and therefore securing WSNs has been on priority for the research community. In this article, a novel elliptic curve signcryption based security protocol for WSNs has been presented which successfully provides user anonymity, confidentiality, mutual authentication, and secure key establishment at the same time taking less computational time than the other related schemes. It has been revealed that the proposed protocol also provides security from an offline dictionary attack, insider attack, impersonation attack, replay attack, and stolen verifier attack. In addition to providing the required security functionalities, our signcryption based protocol consumes least computational time for the gateway in comparison to the other protocols while providing same or higher security level, which makes it suitable to be used for security and privacy critical applications of WSNs.

REFERENCES

[1] M. Kocakulak, and I. Butun, "An overview of Wireless Sensor Networks towards internet of things,"in Proc. of the IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, USA, Dec 2016, pp. 1-6.

[2] J. Lopez, R. Roman, and C. Alcaraz, "Analysis of Security Threats, Requirements, Technologies and Standards in Wireless Sensor Networks," in Foundations of Security Analysis and Design V. FOSAD 2009, FOSAD 2007, FOSAD 2008. Lecture Notes in Computer Science Volume 5705,A. Aldini,G. Barthe, R. Gorrieri, Eds., Springer, Berlin, Heidelberg, 2009, pp. 289-338.

[3] A. Dhakne, and P. Chatur, "Detailed Survey on Attacks in Wireless Sensor Network," inProc. of the International Conference on Data Engineering and Communication Technology. Advances in Intelligent Systems and Computing, Singapore, 2017, pp. 319-331.

[4] A.K. Singh, and B.D.K.Patro, "Security of Low Computing Power Devices: A Survey of Requirements, Challenges & Possible Solutions," Cybernetics and Information Technologies, Vol. 19, No. 1, 2019, pp. 133-164.

[5] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security Enhanced User Authentication Protocol for Wireless Sensor Networks Using Elliptic Curves Cryptography," Sensors, Vol. 14, 2014, pp. 10081-10106.

[6] W. Shi, and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography,"International Journal of Distributed Sensor Networks, 2013pp. 1-7. doi. 10.1155/2013/730831.

[7] F. Wu, L. Xu, S. Kumari, X. Li, A.K. Das, M.K. Khan, M. Karuppiah, and R. Baliyan, R,"A novel and provably secure authentication and key agreement scheme with user anonymity for global mobility networks,"Security and Communication Networks, Vol. 9, 2016, pp. 3527-3542.

[8] R. Amin, S.K.H. Islam, G.P. Biswas, M.K. Khan, L. Leng, and N. Kumar, "Design of anonymity preserving three-factor authenticated key exchange protocol for wireless sensor network,"Computer Networks,Vol.2016, 2016, pp. 1-22.

[9] M.S. Farash, M. Turkanovic´, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment,"Ad Hoc Networks, Vol. 36, 2016, pp. 152-176.

[10]Y. Park, and Y.H. Park, "Three-Factor User Authentication and Key Agreement Using Elliptic Curve Cryptosystem in Wireless Sensor Networks," Sensors, Vol.16, No. 2123, 2016, pp. 1-17.

[11] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three factor authentication and key agreement protocol for internet integrated wireless sensor networks,"IEEE Access, Vol. 5, 2017,pp. 3376–3392.

[12] J. Jung, J. Moon, D. Lee, and D. Won, "Efficient and Security Enhanced Anonymous Authentication with Key Agreement Scheme in Wireless Sensor Networks," Sensors, Vol. 17, No. 644, 2017, pp. 1-21.

[13] I.P. Chang, T.F. Lee, T.H. Lin, and C.M. Liu, "Enhanced Two-Factor Authentication and Key Agreement Using Dynamic Identities in Wireless Sensor Networks," Sensors, Vol.15,2015, pp. 29841-29854.

[14] C. Wang, G. Xu, and J. Sun, "An Enhanced Three-Factor User Authentication Scheme Using Elliptic Curve Cryptosystem for Wireless Sensor Networks," Sensors, Vol.17, No. 2946, 2017, pp.

[15] X. Li, J. Niu, S. Kumari, F. Wu, A.K. Sangaiah, K. Kwang, and R. Choo, "Three-factor Anonymous Authentication Scheme for Wireless Sensor Networks in Internet of Things Environments,"Journal of Network and Computer Applications, Vol. 103, 2017, pp. 194-204.

[16] K. Zhang, K. Xu, andF. Wei, "A Provably Secure Anonymous Authenticated Key Exchange Protocol Based on ECC for Wireless Sensor Networks,"Wireless Communications and Mobile Computing, Vol. 1028, 2018, pp. 1-9.

[17] K.E. Lauter, and K.E. Stange, "The elliptic curve discrete logarithm problem and equivalent hard problems for elliptic divisibility sequences," in Selected Areas in Cryptography, Springer, , 2009, pp. 309-327.

[18] I. Shparlinski, "Computational Diffie-Hellman Problem," in Encyclopedia of Cryptography and Security,H.C.A.Van Tilborg, and S. Jajodia, Eds., Springer, 2011.

[19] D. Boneh, "The Decision Diffie-Hellman problem," in Algorithmic Number Theory, ANTS. Lecture Notes in Computer Science, Volume 1423, J.P. Buhler, Ed., Springer, 1998.

[20] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption),"in Advances in Cryptology —CRYPTO 1997, Lecture Notes in Computer Science, Volume 1294, B.S. Kaliski , Ed., Springer, 1997.

[21] Y. Zheng, and H. Imai, "How to Construct Efficient Signcryption Schemes on Elliptic Curves,"Information Processing Letters, Vol. 68, No 5,1998, pp. 227 – 233.

AUTHORS

Anuj Kumar Singh is pursuing Ph.D. in Computer Science and Engineering from Dr. A.P.J.Abdul Kalam Technical University, Lucknow (India). He is also working as Assistant Professor in the Department of Computer Science & Engineering at Amity University Haryana, Gurgaon (India). He passed M.Tech degree with honors from Panjab University, Chandigarh. He has more than 15 years of teaching experience in technical education. He has published 23 research papers in journals and conferences.

Dr.B.D.K.Patro earned Ph.D. degree in Computer Science from Institute of Computer and Information Sciences, Dr.B.R.Ambedkar University, Agra. He is an Associate Professor of Computer Science & Engineering at RajkiyaEngineeirng College, Kannauj (India). He has more than 24 years of experience to teach the undergraduate and postgraduate courses. He has guided 02 Ph.D, guiding 03 Ph.D candidates and he supervised 12 M.Tech and many Undergraduate projects. He has published more than 30 research papers in journals and conferences.