Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 6, pp. 2365–2391
DOI : 10.47974/JDMSC-2627

A secure key exchange framework for smart grid network using elliptic curve cryptography

Akber Ali Khan [‡]
Department of Applied Sciences and Humanities
IIMT College of Engineering
Greater Noida 201310
Uttar Pradesh
India

Neeraj Kumar Gaud [†]
Swati Agarwal [§]
Institute of Applied Sciences
Mangalayatan University
Beswan
Aligarh 202146
Uttar Pradesh
India

Faisal Alanazi *
Department of Electrical Engineering
College of Engineering
Prince Sattam bin Abdulaziz University
Al-Kharj 11942
Saudi Arabia

Abstract

Traditional power systems rely on fossil fuels, which considerably harm the environment. A smart grid is an infrastructure that allows for the integration of renewable resources into the distribution system, as well as the possibility of establishing a two-way flow of energy and data between network management and subscribers to optimize energy consumption. However, attackers may use this data flow to compromise security and generate power network imbalances. With the ever-growing advancements in computing,

[‡] *E-mail:* cs.akberkhan@gmail.com
[†] *E-mail:* jr.neerajgaud@gmail.com
[§] *E-mail:* swati.agarwal@mangalayatan.edu.in
* *E-mail:* faisal.alanazi@psau.edu.sa (Corresponding Author)

© Taru Publications

cyber security concerns are also escalating, where a Smart grid (SG), being a networked distributed control system and exchanging electricity and consumption information with a high frequency, is also in jeopardy. Nevertheless, there is a notable lack of substantial efforts towards the development of practical secure solutions for power grids. This paper proposes a lightweight protocol designed to address security challenges in smart grid networks. The scheme integrates an authentication mechanism with ECC to securely generate session keys. The proposed protocol security has been evaluated through informal security analysis against a range of potential attacks in SG environment. The proposed protocol computational efficiency and suitability for smart grid devices are demonstrated by performance analysis.

Subject Classification: *11T71, 94A60, 94A62 .*

Keywords: *Elliptic curve cryptography, Smart grid, Key argument, Security and privacy, Mutual authentication, RoM.*

1. Introduction

A smart grid, functioning as a system of distributed and interoperable layers, bears the responsibility to exchange power and information across a networked grid structure integrated with information and communication technology (ICT) [1]. The predominance of ICT in smart grid has increased the space for possible security attacks over data communications [2]. In recent decades, the power grid infrastructure has not evolved at the same rate as industrial growth and social changes, creating a significant gap between electricity supply capabilities and rapidly increasing demand. Data analysis reveals that between 1950 and 2008, the United States experienced an approximate doubling in energy production while consumption nearly tripled [3]. In 2008, the highest electricity demands came primarily from three sectors: public and commercial services, industrial operations, and residential use. Meeting this growing demand presents a significant challenge: effectively coordinating diverse energy sources. This includes both conventional fossil fuels (petroleum, coal, and natural gas) and regenerable alternatives (such as solar and hydroelectric power) [4]. In response, the National Institute of Standards and Technology (NIST) launched nationwide initiatives to develop what now commonly known as the SG a modernized system of electrical power for the next generation [5]. Unlike traditional power systems, the Smart Grid incorporates high-speed, bidirectional communication technologies into millions of electrical devices, creating a dynamic, interactive infrastructure with enhanced energy management capabilities [6]. This including Advanced Metering Infrastructure (AMI)and strategies for responding to demand [7].This extensive reliance on information networks, however, exposes the Smart Grid to vulnerabilities inherent in communication systems. This paradoxically threatens the reliable and secure power delivery that represents the Smart Grid fundamental purpose. Research demonstrates that network security breaches could result in consequences ranging from unauthorized access to customer data to cascading system failures, potentially causing widespread power outages and infrastructure damage [8]. This reality drives our investigation into Smart Grid

cybersecurity issues a critical aspect of information network design that has been identified as among the highest priorities in Smart Grid development [9].

1.1 *Related Work*

The establishment of security policies and requirements is imperative for networks where nodes communicate within a communication infrastructure. The real-time transmission of data and its consequential role in decision-making regarding the control of power transmission equipment and the mitigation of energy distribution issues have increased the complexity of implementing network communication security. Therefore, in such a critical situation, security protocols that verify the identity of communication parties at various points during the protocol process and supply a secure encryption key for information can be instrumental in implementing network security policies. Researchers have proposed various authenticated key establishment mechanisms to ensure the secure communication of data.We briefly describe different studies and approaches introduced to enhance communication security. The public channel functions as a pathway for the SG network entities to communicate and exchange vital data required for SG network equipment. However, its inherent insecurity exposes exchanged data and information to an array of security vulnerabilities.

The literature contains numerous authentication and key agreement protocols specifically designed for SG communications. Fouda et al. [10] introduced a computational Diffie-Hellman protocol for Smart Meters (SM) and various SG equipment. Their approach, however, incurred significant computational burden due to the extensive use of exponential operations. Subsequent evaluation by Mahmood et al. [11] identified critical shortcomings in [10], including vulnerability to insider attacks, absence of perfect forward secrecy (PFS), and excessive computational and communication overhead. Additionally, Mood and Nikooghadam [12] noted that [10] failed to satisfy the CK security model [13] requirements and demonstrated public key infrastructure limitations. Chim et al. [14] developed a protocol prioritizing user privacy protection. However, [11] demonstrated that [14] lacked essential security features including key agreement, mutual authentication, and PFS, while remaining susceptible to impersonation and insider attacks. Nicanfar and Leung [15] contributed a key agreement protocol for SG networks, distinguished by its dual support for forward and backward confidentiality. Wu and Zhou [16] proposed an alternative key distribution framework for SG environments. Subsequently, Xia and Wang [17] revealed that [16] was vulnerable to man in the middle (MITM) attack and offered their improved protocol. Park, Kim, and Kwon [18] later demonstrated that the solution in [17] remained vulnerable to impersonation attacks and did not adequately address unknown key share attacks. Tsai and Lo [19] presented their key distribution protocol for SG contexts, while highlighting numerous security deficiencies in [16], including anonymity issues, replay attack vulnerabilities, PFS limitations, and impersonation attack susceptibilities. They further identified vulnerabilities in [17] related to SM-side impersonation, PFS, MITM and replay attacks. Odelu et al. [20] proposed an authentication scheme for SG systems while

demonstrating that [19] inadequately protected SM credentials and privacy, leaving it vulnerable to credential leakage and various security exploits. Sule, Katti, and Kavasseri, [21] designed an authentication and key exchange protocol with computational and communication requirements comparable to [10]. Mahmood et al. [22] presented an RSA-based authentication mechanism that achieved greater computational efficiency than both [10] and [21]. Li et al. [23] contributed a public key infrastructure-dependent authentication and key exchange framework for SG communication systems, though it required substantial computational and communication resources. In another approach, [11] developed a framework for SG environments utilizing Elliptic Curve Cryptography and hash functions. after that, Kumar, Khan, and Ahmad [24] evaluated this protocol and discovered several security vulnerabilities, particularly regarding insider and impersonation attacks. Abbasinezhad-Mood and Nikooghadam [25] proposed an ECC-based authentication and key agreement framework, which Braeken, Kumar, and Martin [26] demonstrated lacked session key security under the CK security model [13] and remained vulnerable to DoS attacks. Wazid et al. [27] introduced a 3-factor authentication scheme specifically for renewable energy-powered SG communications. Gope and Sikdar [28] designed an authenticated key agreement mechanism employing physically unclonable functions, claiming enhanced resistance against DoS and MITM attack. Further, Khan, Kumar, and Ahmad [29] developed a mutual authentication scheme incorporating fuzzy techniques with ECC for SG communications. Abbasinezhad and Nikooghadam [12] demonstrated that the protocol suggested in [11] is not secure in the CK-security model [13] and the adversary could obtain the session key by applying a random value reveal attack and proposed a new scheme.However, in [12], no one knows how to keep the smart meter identities secret, even with extensive calculation. Recently, Abbasinezhad-Mood, Ostad-Sharif, and Nikooghadam [30] designed a key agreement protocol based on password It is suitable for isolated SMs since it does not necessitate Internet-based connection with the ESP. Futher, Kumar et al. [31] suggested a key exchanged protocol in which the NAN gateway acts as the Trusted Authority to register SMs and provide registration phase settings. Their scheme employs ECC-based encryption, AES, and MAC. Baghestani, Moazami, and Tahavori [32] showed that their protocol does not maintain SMs anonymity, and a malicious meter may follow other SMs and determine whether the consecutive two parallel sessions belong to a SMs or not. Also, Baghestani, Moazami, and Tahavori [32] designed a new framework for SG based on ECC Which is secure in CK security based model [13] and it is also more optimal in terms of performance. Further research contributions to SG authentication include works [25, 32–38], which proposed various authentication and key exchange protocols for SG communication with IoT integration, smart metering systems, and related environments. Analysis reveals that the majority of existing protocols either contain security vulnerabilities or impose excessive computational and communication demands, rendering them suboptimal for practical SG implementation.

1.2 *Objectives and Needs for Security in SG environments*

To ensure the efficient operation of the SG communication system network, It is necessary to address critical concerns related to security and privacy protection. The proposed secure AKA protocol for the SG communication system network is designed to meet the following security requirements:

1.2.1 Privacy Protection

Smart grid devices gather detailed energy consumption data, which, if intercepted or accessed without authorization, could reveal sensitive information about users' behaviors or routines. Robust protocols are required to safeguard transmitted data, detect unauthorized access, and prevent data tampering or manipulation.

1.2.2 Authentication Mechanisms

Unauthorized access to smart grid devices or systems could enable malicious control or disruption. A mutual authentication protocol is vital to verify the identity of all communicating parties and block unauthorized access.

1.2.3 Rules for Access Control

Role-based access control (RBAC) mechanisms must be implemented to restrict users to data and functions relevant to their roles, thereby reducing the risk of unauthorized actions or data exposure.

1.2.4 Data Encryption

All communications within the smart grid must be encrypted to protect against eavesdropping and unauthorized data modifications, ensuring secure data exchange among participants.

1.2.5 Data Integrity Verification

Cryptographic techniques, such as Message Authentication Codes (MAC) or secure cryptographic hash functions must be employed to verify that data remains unaltered during transmission, ensuring its accuracy and reliability.

1.2.6 Confidentiality Assurance

Only authorized recipients should access and interpret encrypted communications. Information regarding system vulnerabilities or potential threats must be protected from malicious entities.

1.2.7 System Redundancy and Resilience

To ensure continuous availability, redundancy and fault-tolerant designs must be incorporated to manage hardware or communication failures effectively, maintaining uninterrupted system operation. Fulfilling these security standards, the proposed designed scheme improves SG networks system privacy and security, offering a robust and reliable solution.

1.3 *Adversary model*

In our scheme, we assume that an attacker in [13, 39] adversary model can eavesdrop on two-way wireless channels and also has the ability to inject new messages, forge messages for specific purposes and access a session key.In addition, the protocol is designed in such a way that when the attacker achieves random numbers, none of the important parameters of the session and the private parameters in the memory of the parties is revealed to the attacker and the proposed design is secure against to ephemeral secret leakage.

1.4 *Motivation and contribution*

In the smart grid network infrastructure, meeting network requirements such as throughput, reliability, operational efficiency, and overall security are essential. High-quality data transmission significantly contributes to fulfilling these objectives. Due to the substantial number of entities in the network, a considerable volume of data is exchanged through insecure channels, creating conditions that allow network attackers to carry out multiple attacks on the network. The implementation of encryption algorithms and authentication mechanisms stands out as a primary necessity in this network. Our objective in establishing a security protocol for the Smart Grid network is to ensure the enforcement of security policies necessary for the network. This ensures that network entities can transmit their data under secure conditions. Most of the methods proposed for securing and implementing authentication mechanisms in the Smart Grid network have weaknesses in terms of security or efficiency. These issues pose challenges to the implementation of smart grid security policies and requirements. Some related schemes have security problems, such as impersonation attacks, MITM attacks, and failure to provide anonymity of communication entities. In addition, the high time for the computation of protocol process operations can lead to excessive workload on low-power entities, which may cause significant problems in the smart grid network.Therefore, it is worthwhile to conduct further investigations into the smart grid infrastructure to devise more effective methods for privacy protection and an efficient authentication mechanism. In this paper, our primary focus is on addressing the security concerns related to smart grid communications, specifically concerning the compatibility of security protocols with resource-constrained entities and the enhancement of robust mutual authentication mechanisms. To meet the security requirements of the Smart Grid network, we propose a key agreement protocol and multiple authentication methods leveraging the advantages of ECDH and ECC theories. The choice of ECC in our protocol is aligned with recommendations from standards such as NIST SP 800-57, ANSI X9.62, and SECG SEC 2, where a 160-bit key provides a security level equivalent to 1024-bit RSA. ECC is known for offering strong security with significantly shorter key lengths compared to traditional public-key schemes such as RSA, which results in faster computations, reduced memory usage, and lower energy consumptionfeatures essential for smart grid devices. The use of 160-bit ECC keys provides a security level equivalent to 1024-bit RSA, which is currently

sufficient against classical attacks. This selection offers an efficient balance between security and performance, making it well-suited for resource-constrained smart grid devices. In addition, the modular design of the protocol, allows easy adoption of larger ECC key sizes (e.g., 224 or 256 bits per NIST SP 800-186) or integration with post-quantum algorithms in the future, ensuring long-term security against emerging threats.

The following summarizes this proposed design papers main contributions:

- To address the evolving security demands of the Smart Grid framework, we first conducted a thorough analysis of vulnerabilities present in existing protocols. Building on this foundation, we propose a key agreement and authentication protocol which is secure against various security attacks in the SG communication.
- Security of the suggested proposed protocol is formally validated through a rigorous analysis within the probability RoM. In addition, we also provided an informal security analysis of the proposed protocol that provide broader assurance of its robustness.
- We compared the performance of our proposed protocol with related protocols. The results indicate that the designed protocol achieves significantly lower communication and computing costs while maintaining robust security compared to other protocols.

This research presents a substantial advancement in key agreement and authentication mechanisms tailored for smart grid environments. By rigorously analyzing the shortcomings of existing protocols, we propose a novel scheme that delivers enhanced security guarantees, computational efficiency, and practical deploymentability. The proposed protocol addresses critical challenges in secure communication, thus facilitating robust and scalable implementations in real-world smart grid infrastructures.

1.5 *The structure of the paper*

The article is structured as follows for the remaining portions:

Section 2 goes over the preliminary steps. In Section 3, proposed protocol is presented. Section 4, of the paper outlines the formal and informal evaluation of the proposed protocol, aimed at determining its effectiveness in resisting common network attacks. The paper proceeds to describe in full an extensive overview of the performance evaluations of the proposed protocol in Section 5, focusing on comparisons with other similar security protocols regarding the complexity of protocol operation calculations and communication cost. Finally, The conclusion of the paper and future work is presented.

2. Preliminaries

This section provides an introduction to the key terms and symbols that will be used throughout the protocol design and security analysis.

2.1 Notations

Table 1 shows the associated helpful notations.

Table 1

Notations used

Notation	Explanation	Notation	Explanation
ECC	Elliptic curve cryptography	$Z_{q_1}^*$	Integer of order q_1 with a finite field
$\mathcal{A}$	Adversary	T_i	The i^{th} time
SID_i	The distinct identity of the entity i	q	Sufficiently large primes
$h(.)$	Secure one way hash function	$\cong$	Approximate number
SK_i	The session key of entities i	ε	Sufficiently small
$D_K(.)$	Symmetric decryption using the key K	SP	Service Provider
P	Concatenation operation	$\oplus$	The bitwise XoR operation
$E_K(.)$	Symmetric encryption using the key K	$\stackrel{?}{=}$	Whether equal or not
ΔT	Maximum transmission delay	g	Generator of G
G	ECC group under operation addition	SMU	Smart Meter User

2.2 *ECC over a Finite Prime Field*

A public key encryption technique is ECC that leverages the mathematics of elliptic curves. Compared to traditional systems like RSA, ECC provides equivalent security with smaller key sizes, resulting in faster and more efficient operations. ECC is used for generating public and private keys, encrypting and decrypting data, creating digital signatures, and performing other cryptographic tasks. The security of ECC relies on the difficulty of the **discrete logarithm problem** for elliptic curves, which involves finding a specific point on the curve. The following formula defines an elliptic curve over a prime finite field:

$$v^2 \equiv u^3 + ru + s(\text{mod } q) \tag{1}$$

where $r, s \in \mathbb{Z}_q^*$ (integers modulo q, excluding zero), and the curve is **non-singular** if:

$$4r^3 + 27s^2(\text{mod } q) \neq 0 \tag{2}$$

The **group** of points on the curve, denoted G, consists of all pairs (u, v) that satisfy the curve equation, plus a special point called the **identity** (or ``point at infinity"). The group uses an operation called **point addition**.

Scalar multiplication is defined in this group as repeatedly appending a point P to itself:

$$aP = P + P + \cdots + P \ (a \text{ instances}) \tag{3}$$

where $a \in \mathbb{Z}_q^*$. This operation is central to ECC cryptographic functions.

2.3 *One-Way Secure Hash Function*

An input string is passed to a **one-way secure hash function** of any length and produces a fixed-length output string. It is defined as:

$$h: \{0,1\}^* \rightarrow \{0,1\}^n \tag{4}$$

where the input x is any string of 0s and 1s, and the output $h(x)$ is a fixed-length string of n bits. The following some important characteristics of a secure hash function are present:

- It can process any input data.
- Every input produces a unique hash output (or nearly so).
- **Weak collision resistance**: Considering an input x_1, it is computationally challenging to determine a different input $x_2 \neq x_1$ such that $h(x_1) = h(x_2)$.
- **Strong collision resistance**: Finding it is really challenging any two different inputs $x_1 \neq x_2$ such that $h(x_1) = h(x_2)$.

The **advantage** of an attacker A trying to find a collision is:

$$\mathrm{Adv}_{\mathrm{HASH}}(\mathcal{A}) = \Pr[(x_1, x_2) \text{ from } \mathcal{A}: x_1 \neq x_2 \text{ and } h(x_1) = h(x_2)] \tag{5}$$

A secure cryptographic hash function is **collision-resistant** if this advantage is very small (less than some tiny ε).

2.4 *Elliptic Curve Discrete Logarithm Problem: (ECDLP)*

ECDLP is a computationally hard problem that underpins ECC security. Given a point V on the elliptic curve and another point cV (where cV is the result of adding V to itself c times), it is difficult to find the number c.

The **advantage** of an attacker $\mathcal{A}$ solving ECDLP is:

$$\mathrm{Adv}_{\mathrm{ECDLP}}(\mathcal{A}) = \Pr[\mathcal{A}(V, cV) = c: c \in \mathbb{Z}_q^*, V \in G] \tag{6}$$

For any practical attacker, this advantage is very small (less than some tiny ε) [40].

2.5 *Elliptic Curve Diffie-Hellman Problem: (ECDHP)*

The ECDHP is another hard problem used in ECC. Given two points cV and dV on the curve (where $c, d \in \mathbb{Z}_q^*$), it is computationally difficult to compute the point cdV.

The **advantage** of an attacker A solving ECDHP is:

$$\mathrm{Adv}_{\mathrm{ECDHP}}(\mathcal{A}) = \Pr[\mathcal{A}(cV, dV) = cdV: c, d \in \mathbb{Z}_q^*, V \in G] \tag{7}$$

This advantage is very small (less than some tiny ε) [40].

2.6 *Elliptic Curve Gap Diffie-Hellman Problem: (ECGDHP)*

The ECGDHP involves computing a point $\lambda\mu X$ given two points λX and μX on the curve, even with access to an ECGDHP oracle. The **advantage** of an attacker A solving this in polynomial time ζ is:

$$\mathrm{Adv}_{\mathrm{ECGDHP}}(\mathcal{A}, \beta) \leq \varepsilon \quad (8)$$

This advantage is very small, indicating that ECGDHP is a hard problem [41].

2.7 *Elliptic Curve Decisional Diffie-Hellman Problem: (ECDDHP)*

The ECDDHP involves determining whether a given point νX is equal to $\lambda\mu X$, given points λX, μX, and νX. The **advantage** of an attacker $\mathcal{A}$ solving this in polynomial time β is:

$$\mathrm{Adv}_{\mathrm{ECDDHP}}(\mathcal{A}, \beta) \leq \varepsilon \quad (9)$$

This advantage is negligibly small, meaning ECDDHP is also computationally difficult [41].

3. The proposed protocol

In order to facilitate SG communication, we suggested an authentication protocol. The three stages of the presented protocol are initialization, registration, login, and authentication.

3.1 *Initialization*

In this stage, the non-singular elliptic curve is selected by TA as $E_q(e, f)$ and g as the base point, and n as the largest prime order over $E_q(e, f)$. Further, TA selects $h(.)$ and generates its private key x_i. Finally, TA publishes the system parameter $\{E_q(e, f), g, q, h(.)\}$.

3.2 *Registration*

In this phase, SMU registered itself with SP performed the following steps and also are shown in Table 2.

Step 1: This phase elaborates the registration procedure.The SMU picks up his SID_i , T_1 Selects $x_i \in Z_q^*$ Compute $\alpha_i = h(x_i \mathrm{P} SID_i).g$ send it $M_1 = \{SID_i, \alpha_i, T_1\}$ to SP.

Step 2: Against registration request message from SPU, SP checks the freshness $T_2 - T_1 \leq \Delta T$ aborts if not fresh then Generates $y_i \in Z_q^*$ and Computes $X_i = y_i.g$ Computes $\beta_i = h(SID_i \mathrm{P} X_i \mathrm{P} \alpha_i)$ Computes $Z_i = x_i + K.\beta_i$ send to SMU $M_2 = \{Z_i, X_i\}$

Step 3: Now SMU verify $Z_i.g \stackrel{?}{=} X_i + (h(SID_i \mathrm{P} X_i)).PK$ Sets its public key $PK_i = Z_i.g$

Table 2

Registration phase of Proposed protocol

Smart-Meter user	Service Provider
Input SID_i, T_1 Selects $x_i \in Z_q^*$ Compute $\alpha_i = h(x_i \mathrm{P} SID_i).g$ $\xrightarrow{M_1=\{SID_i,\alpha_i,T_1\}}$	
	$T_2 - T_1 \leq \Delta T$ aborts if not fresh Generates $y_i \in Z_q^*$ Computes $X_i = y_i.g$ Computes $\beta_i = h(SID_i \mathrm{P} X_i \mathrm{P} \alpha_i)$ Computes $Z_i = x_i + K.\beta_i$ $\xleftarrow{M_2=\{Z_i,X_i\}}$
$Z_i.g \stackrel{?}{=} X_i + (h(SID_i \mathrm{P} X_i)).PK$ Sets its public key $PK_i = Z_i.g$	

3.3 *Login and authentication phase*

After successful mutual authentication between the SMU and the SP, both parties establish a exchanged session key to secure subsequent encrypted communications. The login and authentication phase of the designed proposed protocol consists of the completing the steps, as detailed in Table 3.

Step 1: In the authentication phase, SMU selects his/her SID'_i, calculates $\beta'_i = h(SID'_i \| X_i \| \alpha_i)$ and then verifies $\beta'_i \stackrel{?}{=} \beta'_i$. If this holds, then SMU produces a random nonce $a \in \mathbb{Z}_q^*$, computes $SSID'_i = SID_i \oplus h(a \cdot g \| PK \| T_1)$ and compute $K_i = h(|PK_j \| X_i \| T_1)$ and encrypts $E_1 = E_{K_1}(SSID_i, a \cdot g)$ with the help of key $K_i = h(PK_j \| X_i \| T_1)$. After that, SMU sends $\{E_1, T_1\}$ towards SP through a public channel.

Step 2: Once obtaining $\{E_1, T_1\}$, SP checks the timestamp condition as $T_2 - T_1 \leq \Delta T$. If this condition holds, then SP decrypts $(SID_i) = D_{K_1^*}(E_1)$ with the help of key $K_1^* = h(PK_j \| X_i \| T_1)$ and computes $SID_A^* = SID_i \oplus h(a \cdot g \| PK \| T_1)$. After that, SP generates a random value $b \in \mathbb{Z}_q^*$ and computes $Q = h(a \cdot g \mathrm{P} b \cdot g \| SID_i^* \| T_1)$, computes session key as $SK_j = h(SID_i^* \| ID_j \| Q \| a \cdot g \| b \cdot g \| T_3)$, $ID_j = ID_j \oplus h(a \cdot g \| b \cdot g | T_3 \|)$ and encrypts $E_2 = E_{K_2}(ID_j, Q, a \cdot g)$ with the help of $K_2 = h(a \cdot g \| T_3 \| SID_i^*)$. Further, SP sends $\{E_2, T_3\}$ towards SMU through a public communication channel.

Step 3: After obtaining $\{E_2, T_3\}$, SMU checks the timestamp condition $T_4 - T_3 \leq \Delta T$. If this condition holds, then SMU computes $K_2^* = h(a \cdot g \mathrm{P} T_3 \mathrm{P} SID_i)$ and decrypts $(ID_j, Q, b \cdot g) = D_{K_2^*}(E_2)$. Further, SMU computes the following: $ID_j^* = ID_j \oplus h(a \cdot g \| b \cdot g \| T_3)$, $Q^* = h(a \cdot g \| b \cdot g \| SID_i$

$\|ID_j^*\|T_1)$ and verifies $Q^* \stackrel{?}{=} Q$. If verification is successful, subsequently select its session key as $SK_i = h(SID_i\|ID_j^*\|Q^*\|a \cdot g\|b \cdot g\|T_3)$. Hence, $SK = SK_i = SK_j$.

Table 3

Login and authentication phase

Smart Meter User	Service Provider
Inputs SID_i'	
Computes $\beta_i' = h(SID_i'\|X_i\|\alpha_i)$	
Checks $\beta_i' \stackrel{?}{=} \beta_i$ hold then	
Generates $a \in Z_q^*$	
Computes $SSID_i = SID_i \oplus h(a.g\|PK\|T_1)$	
Computes $K_i = h(PK_j\|X_i\|T_1)$	
Encrypted $E_1 = E_{K_1}(SSID_i, a.g)$	Verifies $T_2 - T_1 \leq \Delta T$
Sends $M_1 = \{E_1, T_1\}$	Computes $K_1^* = h(PK_j\|X_i\|T_1)$
……→	Decrypted $(SID_i, a.g) = D_{K_1^*}(E_1)$
(via public channel)	Computes$SID_i^* = SID_i \oplus h(a.g\|PK\|T_1)$
	Generates $b \in Z_q^*$
	Computes $Q = h(a.g\|b.g\|SID_i^*\|T_1)$
	Computes
	$SK_j = h(SID_i^*\|ID_j\|Q\|a.g\|b.g\|T_3)$
	Computes $ID_j = ID_j \oplus h(a.g\|b.g\|T_3)$
	Computes $K_2 = h(a.g\|T_3\|SID_i^*)$
	Encrypted $E_2 = E_{K_2}(ID_j, Q, b.g)$
	Sends $M_2 = \{E_2, T_3\}$
	←……
	(via public channel)
Checks $T_4 - T_3 \leq \Delta T$	
Computes $K_2^* = h(a.g\text{PT}_3\text{P}SID_i)$	
Decrypted $(ID_j, Q, b.g) = D_{K_2^*}(E_2)$	
Computes $ID_j^* = ID_j \oplus h(a.g\|b.g\|T_3)$	
Computes $Q^* = h(a.g\|b.g\|SID_i\|ID_j^*\|T_1)$	
Verify $Q^* \stackrel{?}{=} Q$	
Computes	
$SK_i = h(SID_i\|ID_j^*\|Q^*\|a.g\|b.g\|T_3)$	
Hence $SK = SK_i = SK_j$	

4. Security analysis

In this section, we conduct both informal and formal security analyses of the proposed framework to assess its robustness against potential threats and to rigorously verify its security properties.

4.1 Informal security analysis

This section includes a demonstration of how the designed protocol is protected against every known security threat and manages security properties for SG communication environments.

Proposition 1: The proposed protocol is secure against Replay attacks.

Proof: All messages exchanged in each session are different from other sessions because of the use of time stamps and random values in all steps. In addition to using the timestamp, each party to the protocol obtains and uses a random value generated by the other party. Therefore, replaying old messages fails the authentication process.

Proposition 2: The proposed protocol is secure against Man-in-the-Middle Attacks.

Proof: The proposed protocol employs secure hash verification and timestamp conditions, ensuring $T_i - T_j \leq T$ at each phase. If an adversary A attempts to interfere during the login or key exchange phase by violating the timestamp condition, the protocol verifies hash conditions $\beta_i' \stackrel{?}{=} \beta_i$ and $Q^* \stackrel{?}{=} Q$. Since the hash function is cryptographically secure, A cannot forge these values, rendering man-in-the-middle attacks infeasible.

Proposition 3: Impersonation attacks cannot be carried out in proposed protocol.

Proof: To impersonate user SMU, an adversary must compute $E_1 = E_{K_1}(\mathrm{SSID}_i, a.g)$, where $K_i = h(\mathrm{PK}_j \| X_i \| T_1)$, $\mathrm{SSID}_i = \mathrm{SID}_i \oplus h(a.g \| \mathrm{PK} \| T_1)$,. The inclusion of secret parameters makes forging E_1 computationally infeasible, ensuring proposed resistance to impersonation attacks.

Proposition 4: The protocol is secure contrary to Privileged Insider Attacks.

Proof: The protocol avoids maintaining a verifier table at either the communicating parties or a third party. Authentication relies on secret keys held by the participants, mitigating risks from stolen verifiers or insider threats.

Proposition 5: The proposed protocol is secure against Eavesdropping Attack.

Proof: Proposed protects against eavesdropping by securing all parameters with hash functions and fresh random nonces in each authentication round. This ensures that adversaries cannot extract sensitive information, such as user identities or session keys.

Proposition 6: Security against DoS Attack.

Proof: During mutual authentication, SMU submits its identity SID' and verifies legitimacy via $\beta_i' \stackrel{?}{=} \beta_i$, where $\beta_i' = h(SID_i' \| X_i \| \alpha_i)$. If the verification fails, the system identifies the user as unauthorized and terminates communication, effectively preventing DoS attacks.

Proposition 7: Security against Ephemeral Secret Leakage Attack.

Proof: Proposed remains secure against ephemeral secret leakage attacks. An adversary A cannot compute the session key SK without both long-term and ephemeral secrets. Whether A lacks temporary or permanent secrets, the protocols security holds due to its reliance on secure hash functions and cryptographic hardness assumptions.

Proposition 8: The proposed protocol is manages Key freshness.

Proof: Key freshness is most likely related to the generation of new keys, which prevents future connectivity from being distorted even if the security of the older keys is compromised. Due to this, it is usually advisable to choose a random integer and a time stamp when determining the freshness of keys in cryptography. In the designed protocol, we select a new random integer as well as a new timestamp at each step. Thus, our suggested technique preserves the freshness of the key.

Proposition 9: The proposed protocol maintain Perfect forward secrecy

Proof: A property of key agreement protocols called perfect forward secrecy ensures that session keys won't be leaked even in the event that long-term secrets utilized in the protocol are compromised. Establishing this feature assuming that all private parameters are exposed, is not possible in schemes based on hash functions and symmetric cryptography. According to the definition of this property, if we assume that the private keys of the communication parties are exposed and not necessarily all the private parameters. since attacker does not have access to SID_i and SID_j, calculation of session key is impossible and our proposed scheme meets perfect forward secrecy feature.

Proposition 10: The proposed protocol maintain Forward/backward secrecy.

Proof: In this scheme, the session key will be a combination of long-term values and values specific to the same session, such as $Q^*, a, g, b,$ which will ensure that the session key of one session is not compromised by session key leakage from other session.

Proposition 11: The proposed protocol is secure against Desynchronization attack

Proof: Protocols that require both parties to update their internal values at the same time can be exposed to de-synchronization attacks. In such scenarios, an attacker may disrupt communication after one party completes its update, preventing the other from doing so and causing a state mismatch. In the scheme presented, the values a_i and b_i are updated at the conclusion of each session; however, only the smart meter user retains these updated values. The service provider neither updates nor stores them. Instead, at the start of each new session, the SM sends the current a_i and b_i values to the SP gateway. This design

ensures synchronization between the parties and protects the protocol from de-synchronization attacks.

Proposition 12: Anonymity and untraceability.

Proof: The protocol incorporates feature to protect against identity disclosure by eavesdroppers on unsecured communication channels. It also ensures session unlinkability, meaning that an adversary cannot relate multiple protocol runs to the same smart meter. To preserve anonymity, the SMs real identity is never directly transmitted, instead it is concealed using a secure one-way hash function before being sent. The real identities are further merged with random values, resulting in unique outputs for every session. However, to prevent the tracking of SMs, it is essential that all messages exchanged in each session differ from those in any other session. Taking into account the messages $M_1 = \{E_1, T_1\}$ that is sent by the SMU user in the first step, we see that the E_1 where $E_1 = E_{K_1}(SSID_i, a.g)$, $K_i = h(PK_j \| X_i \| T_1)$ and also in the designed scheme the random values a_i and b_i are updated at the conclusion of every session. Likewise, it is evident that message $M_2 = \{E_2, T_3\}$ which, in the subsequent second stage, are transmitted from SP to the SMU, likewise employ new random values and will vary for every session.

Proposition 13: The proposed protocol is resistance against Ephemeral secret leakage attack

Proof: In this scheme, with the help of hash function and XOR operation, important security parameters which disclosure of them will disturb the security of the scheme, are protected in such a way that even the leakage of all temporary values at the same time does not cause the leakage of these parameters.

4.2 *Formal Security Analysis*

The detailed security evaluation of the suggested protocol is conducted under the Random Oracle Model (RoM), as defined in the security model referenced earlier. The protocol involves two entities: SMU and SP. In the random oracle model, oracles respond to adversarial queries with one of three outcomes: "accept," "reject," or "⊥" (denoting an undefined or failed state), based on the adversary.

4.2.1 Security Definitions

To analyze the semantic security of the proposed protocol, we define the following terms and assumptions.

- **Partnering:** Oracles SMU and SP are partners if they have mutually authenticated and share a common session key.
- **Fresh Session:** A session is fresh if neither it nor its partnered session has been compromised (e.g., via SKReveal queries).

- **Adversary Advantage:** The adversary advantage in breaking proposed semantic security is defined as $Adv_{\text{proposed}}(\mathcal{A}) = |2 \cdot \Pr(\text{succ}) - 1|$, where $\Pr(\text{succ})$ is the probability of the adversary distinguishing the session key from a random value.
- **Security Condition:** proposed is secure if $Adv_{\text{proposed}}(\mathcal{A}) \leq \varepsilon$, where ε is a negligible probability, as per the adversarial model in [39].
- **Reveal:** Given an input string x, the oracle outputs $y = h(x)$, where h is a random function in the ROM.
- **h(messages):** Models a hash function, returning a random hash value $hv \in \{0,1\}^L$, where L is the hash output length, for a given message input.
- **Execute(SMU, SP):** Simulates passive attacks, allowing the adversary to eavesdrop on messages exchanged between SMU and SP during protocol execution.
- **Send(β, messages):** Represents active attacks, where the adversary sends messages to oracle β, receiving responses as specified by the protocol.
- **EsReveal(β):** Discloses the ephemeral secret held by oracle β to the adversary.
- **SKReveal(β):** Reveals the session key established by oracle β to the adversary.
- **Corrupt(β):** Grants the adversary access to the long-term secrets of oracle β.
- **Expire(β):** Deletes the session key of the session managed by oracle β.
- **TestID(β):** Tests the anonymity feature of proposed. On receiving a TestID(β) query, a fair coin c is flipped. If $c = 1$, the true identity of β is returned; otherwise, a random value is returned.
- **TestSK(β):** Evaluates the session key semantic security. Upon obtaining a TestSK(β), a fair coin c is flipped. The true session key is returned if $c = 1$; If not, an equivalently lengthed random value is returned.
- **Difference Lemma:**

  Let x denote an error event distinguishing two probability distributions x_1 and x_2, such that the events satisfy $x_1 \wedge \neg x \Leftarrow x_2 \wedge \neg x$. Then, the absolute difference in their probabilities is bounded by the probability of the error event:

$$|\Pr[x_1] - \Pr[x_2]| \leq \Pr[x].$$

Theorem 1: *Semantic Security of proposed protocol: Let us consider an adversary or attacker $\mathcal{A}$ with polynomial-time capabilities attacking the semantic security of the proposed protocol. Let q_e denote the number of Execute queries, q_s the number of Send queries, and q_H the number of hash queries. Let $|D|$ represent the cardinality of a uniformly distributed set, and let $Adv_{EK}^{SE}(A)$ denote the adversary's advantage against the encryption or decryption scheme*

Enc_K. The following equation provides the advantage of A in breaching the suggested semantic security:

$$Adv_{\text{proposed}}(\mathcal{A}) \leq \frac{q_H^2+q_s}{2^{L+1}} + 2Adv_{EK}^{SE}(\mathcal{A}) + \frac{(q_s+q_e)^2}{2q} + \frac{2q_s}{|D|}$$
$$+2q_H \max\{Adv_{\text{ECDLP}}(\mathcal{A}), Adv_{\text{ECDHP}}(\mathcal{A})\},$$

where L is the hash function output length, q is a security parameter, $Adv_{ECDLP}(\mathcal{A})$ is the adversary's advantage in solving the Elliptic Curve Discrete Logarithm Problem (ECDLP), and $Adv_{ECDHP}(A)$ is the advantage in solving the Elliptic Curve Diffie-Hellman Problem (ECDHP).

Proof: To illustrate the suggested semantic security, we employ a sequence of games, starting from $\mathcal{GM}_0$ (the real attack) and ending at $\mathcal{GM}_6$, where the adversary has no advantage. Each game modifies the protocol simulation to bound the adversary's success probability.

- $\mathcal{GM}_0$**:** This game represents the real attack in the Random Oracle Model (ROM), where the adversary $\mathcal{A}$ interacts with the proposed protocol. The adversary's advantage is defined as:

$$Adv_{\text{proposed}}(\mathcal{A}) = |2\Pr[\text{Succ}_0] - 1|.$$

- $\mathcal{GM}_1$**:** This game simulates $\mathcal{GM}_0$, but oracles store responses to distinct queries in lists to ensure consistency. Since the simulation is indistinguishable from $\mathcal{GM}_0$, we have:

$$\Pr[\text{Succ}_1] = \Pr[\text{Succ}_0].$$

- $\mathcal{GM}_2$**:** This game extends $\mathcal{GM}_1$, but terminates if a collision occurs in hash queries or protocol transcripts. By the birthday paradox, the probability of a hash collision is at most $\frac{q_H^2}{2^{L+1}}$, and the probability of a transcript collision is at most $\frac{(q_s+q_e)^2}{2q_1}$. Applying the Difference Lemma, we obtain:

$$|\Pr[\text{Succ}_2] - \Pr[\text{Succ}_1]| \leq \frac{q_H^2}{2^{L+1}} + \frac{(q_s+q_e)^2}{2q_1}.$$

- $\mathcal{GM}_3$**:** This game is similar to $\mathcal{GM}_2$, but terminates if the adversary correctly guesses the verifier's conditions without querying the oracle. The games are identical unless the adversary forges a valid authentication value, yielding:

$$|\Pr[\text{Succ}_3] - \Pr[\text{Succ}_2]| \leq \frac{q_s}{2^L}.$$

- $\mathcal{GM}_4$**:** This game assesses the suggested session key security by making sure the attacker is unable to calculate the session key. Two situations are taken into account:
  - **Scenario a (Perfect Forward Secrecy):** The adversary issues Corrupt(SMU) and Corrupt(SP) queries to obtain the long-term secret keys of SMU and SP(e.g., (a, b)) and the session key SK, but not their ephemeral secrets.

- **Scenario b (Known Session-Specific Temporary Information):** The adversary issues EsReveal(SMU) and EsReveal(SP) queries to obtain the ephemeral secrets of SMU and SP, but not their long-term secrets.

In both scenarios, computing the session key SK requires solving the ECDLP or ECDHP, or obtaining a secure hash value. As long as ECDLP and ECDHP remain hard, the games are indistinguishable, giving:

$$|\Pr[\text{Succ}_4] - \Pr[\text{Succ}_3]| \leq q_H \max\{Adv_{\text{ECDLP}}(\mathcal{A}), Adv_{\text{ECDHP}}(\mathcal{A})\}.$$

- $\mathcal{GM}_5$**:** This game differs from GM_4 by terminating if the adversary issues a TestID query to reveal the true identity or a query to obtain the password. Thus:

$$|\Pr[\text{Succ}_5] - \Pr[\text{Succ}_4]| \leq \frac{q_s}{|D|} + Adv_{EK}^{SE}(\mathcal{A}).$$

- $\mathcal{GM}_6$**:** This game is identical to GM_5, except it terminates if the adversary issues a TestSK query with a hash query of the form $h(SID_i \| ID_j^* \| Q^* \| a.g \| b.g \| T_3)$. The probability of the adversary guessing the correct hash input is at most $\frac{Q_H^2}{2^{L+1}}$, yielding:

$$|\Pr[\text{Succ}_6] - \Pr[\text{Succ}_5]| \leq \frac{q_H^2}{2^{L+1}}.$$

In $\mathcal{GM}_6$, without a correct hash query, the adversary cannot distinguish the real session key from a random value, so $\Pr[\text{Succ}_6] = \frac{1}{2}$. Summing the probability differences across all games and combining with the initial advantage definition, we derive:

$$Adv_{\text{proposed}}(\mathcal{A}) \leq \frac{q_H^2 + q_s}{2^{L+1}} + 2Adv_{EK}^{SE}(\mathcal{A}) + \frac{(q_s + q_e)^2}{2q_1} + \frac{2q_s}{|D|} + 2q_H \max\{Adv_{\text{ECDLP}}(\mathcal{A}), Adv_{\text{ECDHP}}(\mathcal{A})\}.$$

This completes the proof of Theorem.

5. Performance Analysis

This subsection evaluates the proposed protocol against other authentication schemes by examining theoretical computation costs (operations), the duration of each operation, and their respective pros and cons. Given that smart meters are resource-constrained devices, any security protocol implemented on them must not only fulfill security requirements but also be efficient in terms of computational cost, communication expenses, and data overhead. Also, Table 4, 6, 7 and Figures 1, 2 provide a visual illustration of performance analysis.

5.1 A comparison of features related to functionality and security

This section presents a comparative analysis of the security aspects of various proposed protocols. Table 4 assesses the security strengths and potential vulnerabilities of each protocol. The security and functional capabilities of the proposed framework are compared with those of existing authentication schemes

referenced in [11, 12, 19, 20, 23, 25, 30-33, 35-38]. The proposed solution is designed to meet all key security requirements within smart grid environments, offering robust protection against known attacks. As illustrated in Table 4, the proposed system demonstrates notable improvements in both security and functionality. It ensures secure key establishment and mutual authentication among entities, while also providing forward secrecy, maintaining user privacy, and supporting fairness during session key negotiation.

Table 4

Comparison security and properties

Security Attack	[19]	[36]	[35]	[38]	[20]	[11]	[12]	[25]	[31]	[23]	[30]	[33]	[37]	[32]	Proposed
Impersonation attack	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	×	✓	✓	✓
Man in the middle attack	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	×	✓	✓	✓
Denial of service attack	×	✓	×	✓	×	✓	✓	×	✓	×	✓	✓	✓	✓	✓
Replay attack	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓
Insider attack	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓	✓	✓
Mutual authentication	✓	✓	×	✓	✓	×	✓	×	✓	×	×	✓	×	×	✓
User Anonymity	×	×	✓	✓	×	×	×	×	×	✓	×	✓	✓	✓	✓
Data confidentiality	✓	✓	×	✓	✓	×	✓	×	✓	×	×	✓	×	×	✓
Perfect forward security	✓	✓	✓	✓	✓	×	✓	×	✓	✓	✓	✓	✓	✓	✓
Untraceability	×	×	✓	✓	×	×	×	×	×	✓	×	✓	✓	✓	✓
Private key privacy	✓	×	×	✓	✓	×	✓	×	✓	×	×	✓	✓	×	✓
Session key agreement	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Session key security	✓	×	×	✓	×	×	✓	×	✓	×	×	✓	×	×	✓
Stolen verifier attack	×	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Security key leakage	×	×	✓	✓	✓	×	✓	✓	×	✓	✓	✓	×	✓	✓

Observation ⇒ ×: does not prevent the attack or does not provide the security property. : Prevents the attack or provides the security property.

5.2 *Evaluation of the computation cost*

We evaluated the time complexity of the proposed protocols under consistent testing conditions, using two different systems. A Philips HiPersmart card and a Pentium IV computer, with maximum clock speeds of 3 GHz and 36 MHz, respectively, were used to perform cryptographic pairing operations. The Philips HiPersmart card features a 32-bit RISC MIPS-based processor with a 2 KB of instruction cache, five-stage pipeline, 16 KB of RAM and 256 KB of flash memory. In comparison, the Pentium IV system runs Windows XP, is equipped with 512 MB of RAM, and operates at a peak clock speed of 3 GHz.

Table 5

Execution time of cryptographic primitives in millisecond

Operations	Description	SMU	SP
T_h	Hash function	1ms	0.01ms
T_{s_e}	Symmetric encryption/decryption	2ms	0.02ms
T_{a_e}	Asymmetric encryption/decryption	100ms	1ms
T_e	Exponential	100ms	1ms
$T_{mp/ecc}$	ECC point wise multiplication	130ms	1.17ms
$T_{ecc/add}$	ECC Addition	10ms	0.1ms
T_b	Bilinear pairing	380ms	3.16ms
T_{mac}	MAC operation	1ms	0.01ms

Table 6

Evaluation of computation cost

Protocols	SMU	SP	Total Cost (ms)
Tsai and Lo [19]	$4T_{mp} + T_e + 5T_h \approx 625$	$3T_{mp} + T_e + 2T_b + 5T_h \approx 10.88$	≈ 635.88
He et al.[36]	$4T_{mp} + 1T_{add} + 5T_h \approx 535$	$6T_{mp} + 2T_{add} + 6T_h \approx 7.28$	≈ 542.28
Chen et al. [35]	$2T_{mp} + T_e + 5T_h \approx 365$	$3T_{mp} + T_e + T_b + 5T_h \approx 7.72$	≈ 372.72
Zhang et al. [38]	$T_{se} + 7T_h \approx 9$	$3T_{se} + 9T_h \approx 0.0054$	≈ 9.0054
Odelu et al. [20]	$3T_{mp} + T_e + 6T_h \approx 496$	$2T_{mp} + T_e + 2T_b + 6T_h \approx 9.72$	≈ 505.72
Mahmood et al. [11]	$5T_{mp} + 2T_{add} + 4T_h \approx 674$	$5T_{mp} + 2T_{add} + 4T_h \approx 6.09$	≈ 680.09
Abbasinezhad and Nikooghadam [12]	$4T_{mp} + 2T_{add} + 4T_h \approx 544$	$4T_{mp} + 2T_{add} + 4T_h \approx 4.92$	≈ 548.92
Abbasinezhad and Nikooghadam [25]	$4T_{mp} + 1T_{add} + 5T_h \approx 535$	$4T_{mp} + T_{add} + 5T_h \approx 4.83$	≈ 539.83
Kumar et al. [31]	$3T_{mp} + 2T_m + 4T_h + 2T_{se} \approx 400$	$3T_{mp} + 3T_m + 5T_h + 2T_{se} \approx 3.63$	≈ 403.63
Li et al. [23]	$3T_e + 3T_h \approx 303$	$4T_e + 3T_h \approx 4.03$	≈ 307.03
Abbasinezhad-Mood, Ostad-Sharif, and Nikooghadam [30]	$2T_{mp} + 4T_h \approx 264$	$2T_{mp} + 2T_{se} + 8T_h \approx 2.3432$	≈ 266.3432
Baghestani, Moazami, and Tahavori [33]	$3T_{mp} + 1T_{add} + 7T_h \approx 407$	$3T_{mp} + T_{add} + 7T_h \approx 3.68$	≈ 410.68
Qi and Chen [37]	$3T_{mp} + 1T_{add} + 4T_h \approx 404$	$4T_{mp} + 2T_{add} + 6T_h \approx 4.94$	≈ 408.94
Baghestani, Moazami, and Tahavori [32]	$2T_{mp} + 5T_h \approx 265$	$2T_{mp} + 11T_h \approx 2.45$	≈ 267.45
Proposed	$2T_h + 2T_{se} \approx 8$	$2T_{se} + 4T_h \approx 0.08$	≈ 8.08

For implementation, the pairing operations on the Philips HiPersmart card were developed using the C language within the Ashling integrated development environment. Meanwhile, the Pentium IV computer executed the same operations using standard C. Both platforms utilized the Multi-precision Integer and Rational Arithmetic C/C++ Library (MIRACL) for computations. To assess computational costs, the client-side performance was evaluated on the Philips HiPersmart card, while the server-side evaluation was conducted on the Pentium IV computer. To assess the efficiency of the proposed protocol, we analyze the computational requirements and compare them with existing approaches in Table 5, referencing prior work from [42]. The findings indicate that authentication and key exchange mechanisms built on elliptic curve cryptography typically require less computational effort than those based on bilinear pairings or conventional public key cryptosystems. However, operations like elliptic curve point multiplication still take considerably longer to execute than faster cryptographic primitives such as hash functions. Table 6 compares the computational costs of suggested protocol with those of current for SG applications. The results demonstrate that our method achieves greater efficiency than comparable hash-based schemes currently used in SG environments.

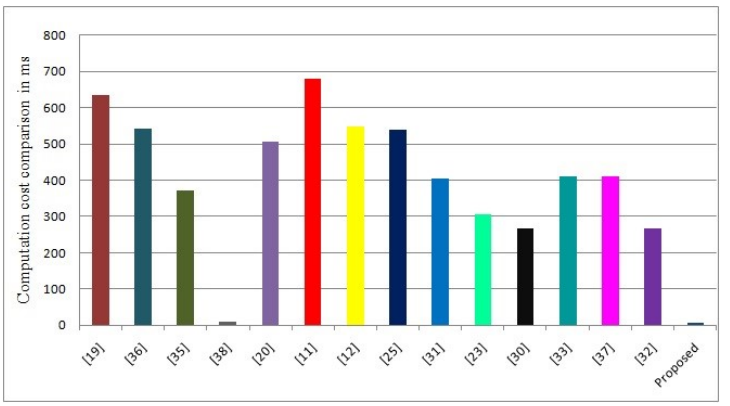


Figure 1
Comparison of the computation cost

5.3 *Communication cost comparison*

The communication overhead of an authentication protocol is determined by both the protocol rounds and the total data transmitted. Our scheme achieves optimal efficiency by requiring only two communication exchanges to complete mutual authentication and key establishment, fewer than comparable protocols in this category. To analyze the communication costs during the authentication phase, we assume the following bit lengths for key components: identity, message digest (using the SHA-1 hash algorithm), and random number/nonce are each 160 bits, while the timestamp is 32 bits. Since the security of a 160-bit ECC system is considered equivalent to that of a 1024-bit RSA system [42], an elliptic curve

bits, while the timestamp is 32 bits. Since the security of a 160-bit ECC system is considered equivalent to that of a 1024-bit RSA system [42], an elliptic curve point requires a total of 320 bits (160 bits for each coordinate). To determine the message size in bits, we use the values in Table 7. This table 7 presents a comprehensive evaluation comparing our protocol with existing protocols.

Table 7

A comparison of the cost of communication

	Communication cost in bits	Number of messages
Tsai and Lo [19]	3456	3
He et al.[36]	1856	3
Chen et al. [35]	2720	3
Zhang et al. [38]	1408	4
Odelu et al. [20]	3712	3
Mahmood et al. [11]	1856	2
Abbasinezhad and Nikooghadam [12]	1856	3
Abbasinezhad and Nikooghadam [25]	1344	3
Kumar et al. [31]	2112	2
Li et al. [23]	2624	2
Abbasinezhad-Mood, Ostad-Sharif, and Nikooghadam [30]	1536	3
Baghestani, Moazami, and Tahavori [33]	1760	3
Qi and Chen [37]	1600	2
Baghestani et al. [32]	1696	2
Proposed	704	2

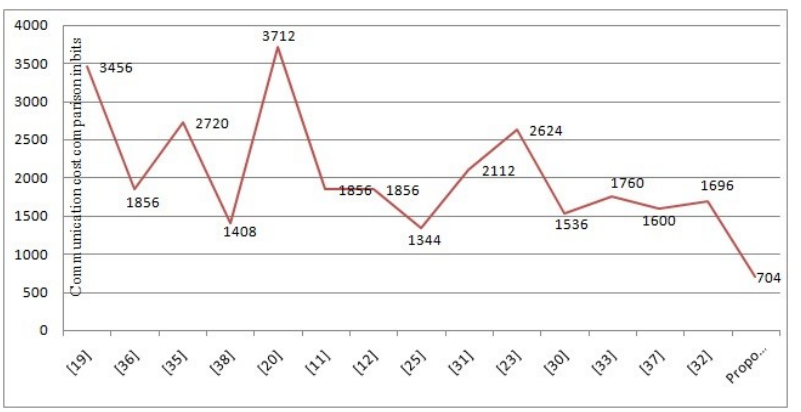


Figure 2

Comparison of the communication cost

6. Conclusion

This research presents an innovative solution to Smart Grid security challenges through an optimized elliptic curve cryptography implementation. Our

cryptographic techniques to establish a secure data transmission mechanism for power consumption information. The solution capitalizes on ECC ability to provide robust protection with compact key sizes, significantly improving computational efficiency a crucial consideration for resource limited Smart Grid infrastructure. We incorporate cryptographically strong random number generation to ensure key unpredictability, substantially enhancing system security. These contributions advance cybersecurity in critical infrastructure while demonstrating ECC viability as an effective security solution. Through comprehensive security analysis, we validate that our protocol meets all critical security requirements while outperforming existing solutions in key metrics, memory efficiency, computational overhead, and communication efficiency. The results confirm its suitability for practical Smart Grid deployment. Future research directions include refining ECC-based protocol efficiency and developing specialized cryptographic components specifically designed for Smart Grid operational requirements.

Acknowledgements

This study is supported via funding from Prince sattam bin Abdulaziz University project number (PSAU/2026/R/1447)

References

[1] C. Peng, H. Sun, M. Yang, and Y.-L. Wang, "A survey on security communication and control for smart grids under malicious cyber attacks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 8, pp. 1554–1569 (2019).

[2] C.-W. Ten, G. Manimaran, and C.-C. Liu, "Cybersecurity for critical infrastructures: Attack and defense modeling," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans,* vol. 40, no. 4, pp. 853–865 (2010).

[3] G. Lu, D. De, and W.-Z. Song, "SmartGridLab: A laboratory-based smart grid testbed," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, pp. 143–148 (2010).

[4] A. Q. Huang, M. L. Crow, G. T. Heydt, J. P. Zheng, and S. J. Dale, "The future renewable electric energy delivery and management (FREEDM) system: The energy internet," *Proc. IEEE*, vol. 99, no. 1, pp. 133–148 (2011).

[5] G. W. Arnold, D. A. Wollman, G. FitzPatrick, D. Prochaska, D. Holmberg, D. H. Su, A. R. Hefner Jr., N. T. Golmie, T. L. Brewer, M. Bello, et al., "NIST framework and roadmap for smart grid interoperability

standards, release 1.0," *Nat. Inst. Standards Technol. (NIST)*, Gaithersburg, MD, USA, Rep., (2010).

[6] A. Aggarwal, S. Kunta, and P. K. Verma, "A proposed communications infrastructure for the smart grid," in *Proc. IEEE Innovative Smart Grid Technol. (ISGT)*, pp. 1–5 (2010).

[7] H. Sui, H. Wang, M.-S. Lu, and W.-J. Lee, "An AMI system for the deregulated electricity markets," *IEEE Trans. Ind. Appl.*, vol. 45, no. 6, pp. 2104–2108 (2009).

[8] A. R. Metke and R. L. Ekl, "Smart grid security technology," in *Proc. IEEE Innovative Smart Grid Technol. (ISGT)*, pp. 1–7 (2010).

[9] G. N. Ericsson, "Cyber security and power system communication—Essential parts of a smart grid infrastructure," *IEEE Trans. Power Del.*, vol. 25, no. 3, pp. 1501–1507 (2010).

[10] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. S. Shen, "A lightweight message authentication scheme for smart grid communications," *IEEE Trans. Smart Grid,* vol. 2, no. 4, pp. 675–685 (2011).

[11] K. Mahmood, S. A. Chaudhry, H. Naqvi, S. Kumari, X. Li, and A. K. Sangaiah, "An elliptic curve cryptography based lightweight authentication scheme for smart grid communication," *Future Gener. Comput. Syst.,* vol. 81, pp. 557–565 (2018).

[12] D. Abbasinezhad-Mood and M. Nikooghadam, "Design and hardware implementation of a security-enhanced elliptic curve cryptography based lightweight authentication scheme for smart grid communications," *Future Gener. Comput. Syst.,* vol. 84, pp. 47–57 (2018).

[13] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in Advances in Cryptology—EUROCRYPT 2001, *Lecture Notes in Computer Science*, vol. 2045, Berlin, Germany: Springer, pp. 453–474 (2001).

[14] T. W. Chim, S.-M. Yiu, L. C. Hui, and V. O. K. Li, "PASS: Privacy-preserving authentication scheme for smart grid network," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, pp. 196–201 (2011).

[15] H. Nicanfar and V. C. M. Leung, "Password-authenticated cluster-based group key agreement for smart grid communication," *Secur. Commun. Netw.*, vol. 7, no. 1, pp. 221–233 (2014).

[16] D. Wu and C. Zhou, "Fault-tolerant and scalable key management for smart grid," *IEEE Trans. Smart Grid,* vol. 2, no. 2, pp. 375–381 (2011).

[17] J. Xia and Y. Wang, "Secure key distribution for the smart grid," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1437–1443 (2012).

[18] J. H. Park, M. Kim, and D. Kwon, "Security weakness in the smart grid key distribution scheme proposed by Xia and Wang," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1613–1614 (2013).

[19] J.-L. Tsai and N.-W. Lo, "Secure anonymous key distribution scheme for smart grid," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 906–914 (2016).

[20] V. Odelu, A. K. Das, M. Wazid, and M. Conti, "Provably secure authenticated key agreement scheme for smart grid," *IEEE Trans. Smart Grid,* vol. 9, no. 3, pp. 1900–1910 (2018).

[21] R. Sule, R. S. Katti, and R. G. Kavasseri, "A variable length fast message authentication code for secure communication in smart grids," in *Proc. IEEE Power Energy Soc. General Meeting (PESGM*), pp. 1–6 (2012).

[22] K. Mahmood, S. A. Chaudhry, H. Naqvi, T. Shon, and H. F. Ahmad, "A lightweight message authentication scheme for smart grid communications in power sector," *Comput. Electr. Eng.*, vol. 52, pp. 114–124 (2016).

[23] X. Li, F. Wu, S. Kumari, L. Xu, A. K. Sangaiah, and K.-K. R. Choo, "A provably secure and anonymous message authentication scheme for smart grids," *J. Parallel Distrib. Comput.,* vol. 132, pp. 242–249 (2019).

[24] V. Kumar, A. A. Khan, and M. Ahmad, "Design flaws and cryptanalysis of elliptic curve cryptography-based lightweight authentication scheme for smart grid communication," in *Advances in Data Sciences, Security and Applications: Proc. ICDSSA 2019*. Singapore: Springer, pp. 169–179 (2019).

[25] D. Abbasinezhad-Mood and M. Nikooghadam, "An anonymous ECC-based self-certified key distribution scheme for the smart grid," *IEEE Trans. Ind. Electron*., vol. 65, no. 10, pp. 7996–8004 (2018).

[26] A. Braeken, P. Kumar, and A. Martin, "Efficient and provably secure key agreement for modern smart metering communications," *Energies*, vol. 11, no. 10, Art. no. 2662 (2018).

[27] M. Wazid, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Secure three-factor user authentication scheme for renewable-energy-based smart grid environment," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3144–3153 (2017).

[28] P. Gope and B. Sikdar, "Privacy-aware authenticated key agreement scheme for secure smart grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3953–3962 (2019).

[29] A. A. Khan, V. Kumar, and M. Ahmad, "An elliptic curve cryptography based mutual authentication scheme for smart grid communications using biometric approach," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 3, pp. 698–705 (2022).

[30] D. Abbasinezhad-Mood, A. Ostad-Sharif, and M. Nikooghadam, "Novel anonymous key establishment protocol for isolated smart meters," *IEEE Trans. Ind. Electron.*, vol. 67, no. 4, pp. 2844–2851 (2020).

[31] P. Kumar, A. Gurtov, M. Sain, A. Martin, and P. H. Ha, "Lightweight authentication and key agreement for smart metering in smart energy networks," *IEEE Trans. Smart Grid,* vol. 10, no. 4, pp. 4349–4359 (2019).

[32] S. H. Baghestani, F. Moazami, and M. Tahavori, "Lightweight authenticated key agreement for smart metering in smart grid," *IEEE Syst. J.*, vol. 16, no. 3, pp. 4983–4991 (2022).

[33] J. Srinivas, A. K. Das, X. Li, M. K. Khan, and M. Jo, "Designing anonymous signature-based authenticated key exchange scheme for Internet of Things-enabled smart grid systems," *IEEE Trans. Ind. Informat.,* vol. 17, no. 7, pp. 4425–4436 (2021).

[34] Y. Li, "An improved lightweight and privacy preserving authentication scheme for smart grid communication," *J. Syst. Archit.*, vol. 152, Art. no. 103176 (2024).

[35] Y. Chen, J.-F. Martínez, P. Castillejo, and L. López, "An anonymous authentication and key establish scheme for smart grid: FAuth," *Energies,* vol. 10, no. 9, Art. no. 1354 (2017).

[36] D. He, H. Wang, M. K. Khan, and L. Wang, "Lightweight anonymous key distribution scheme for smart grid using elliptic curve cryptography," *IET Commun.*, vol. 10, no. 14, pp. 1795–1802 (2016).

[37] M. Qi and J. Chen, "Two-pass privacy preserving authenticated key agreement scheme for smart grid," *IEEE Syst. J.,* vol. 15, no. 3, pp. 3201–3207 (2021).

[38] L. Zhang, L. Zhao, S. Yin, C.-H. Chi, R. Liu, and Y. Zhang, "A lightweight authentication scheme with privacy protection for smart grid communications," *Future Gener. Comput. Syst.,* vol. 100, pp. 770–778 (2019).

[39] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208 (1983).

[40] A. A. Khan, V. Kumar, M. Ahmad, and S. Rana, "LAKAF: Lightweight authentication and key agreement framework for smart grid network," *J. Syst. Archit.*, vol. 116, Art. no. 102053 (2021).

[41] A. A. Khan, V. Kumar, R. Prasad, and M. J. Idrisi, "SGAK: A robust ECC-based authenticated key exchange protocol for smart grid networks," *IEEE Access* (2024).

[42] N. Kumar, G. S. Aujla, A. K. Das, and M. Conti, "ECCAuth: A secure authentication protocol for demand response management in a smart grid system," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6572–6582 (2019).

Received July, 2025