AN EFFICIENT AND SECURE AUTHENTICATION SCHEME BASED ON ELLIPTIC CURVE CRYPTOGRAPHY (ECC)

[1]Chani Saini, [2]Dr. Sandeep Kumar Tiwari and [3]Dr. Ankur Nehra
[1]Research Scholar, Department of Mathematics, Faculty of Science, Motherhood University, Haridwar, Uttarakhand, 24766, India
[2]Supervisor, Department of Mathematics, Faculty of Science, Motherhood University, Roorkee, Uttarakhand, 247667, India
[3]Co-supervisor, Department of Mathematics, Dhanauri P.G. College, Dhanauri, Haridwar, Uttarakhand, 247667, India
Corresponding Author: [1]Chani Saini, Research Scholar [1](sainichani67@gmail.com), [2]Email ID: fos.sandeep@motherhooduniversity.edu.in, [3]Email ID: drankurnehra648@gmail.com

ABSTRACT: Elliptic Curve Cryptography (ECC) has emerged as a powerful alternative to traditional public key cryptographic schemes due to its high security with significantly smaller key sizes. With the growing demand for lightweight, fast, and secure authentication mechanisms in modern communication systems, conventional schemes based on RSA and Diffie-Hellman face challenges related to computational overhead and scalability. This paper proposes an efficient and secure authentication scheme based on elliptic curve cryptography that ensures strong mutual authentication, data integrity, and resistance against common cryptographic attacks. The proposed scheme leverages elliptic curve-based key exchange and digital signature mechanisms to achieve reduced computational cost and communication overhead while maintaining high security levels. Security analysis demonstrates that the scheme is resistant to replay attacks, impersonation attacks, man-in-the-middle attacks, and key compromise attacks. Performance evaluation shows that the proposed authentication protocol outperforms existing schemes in terms of execution time, memory usage, and communication efficiency, making it suitable for resource-constrained environments and next-generation secure communication systems.

[Saini, C., Tiwari, S.K. and Nehra, A**. AN EFFICIENT AND SECURE AUTHENTICATION SCHEME BASED ON ELLIPTIC CURVE CRYPTOGRAPHY (ECC).** *The International Journal of Interpretation, Observation and Analysis*, 2025; Volume 4, Issue 1:183-197 (October-December). ISSN 2349-0713, Peer-reviewed (online/offline), Refereed, Indexed and International Journal (Since 2013), Global Impact Factor: 6.205

Keywords: Elliptic Curve Cryptography, Authentication Scheme, Secure Communication, Key Exchange, Digital Signatures, Public Key Cryptography, Performance Analysis.

INTRODUCTION

The word cryptography comes from the Greek words κρυπτο (hidden or secret) and γραφη (writing). The art and science of keeping messages secure is "Cryptography". The art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form. The basic service provided by cryptography is the ability to send information between participants in a way that prevents others from reading it. We will concentrate on the kind of cryptography that is based on representing information as numbers and mathematically manipulating those numbers.

This kind of cryptography can provide other services, such as:

- Authentication: The process of proving one's identity. (The primary forms of host-to-host authentication on the Internet today are name-based or address-based, both of which are notoriously weak.)
- Privacy/confidentiality: Ensuring that no one can read the message except the intended receiver.
- Integrity: Assuring the receiver that the received message has not been altered in any way from the original.
- Non-repudiation: A mechanism to prove that the sender really sent this message.

According to the traditional use of cryptography, an original message is known as plaintext (or cleartext), while the coded message is called ciphertext. The process of converting plaintext into ciphertext is known as encryption or enciphering, whereas restoring the plaintext from ciphertext is called decryption or deciphering. Cryptology remained a

public field in the United States until World War I, after which the Army and Navy recognized its strategic importance for national security and began conducting cryptographic research in secrecy. Until the early 1970s, cryptology was largely dominated by government agencies due to the high cost of computers and limited public dissemination of cryptographic research. With the advent of the computer revolution and the increasing demand for secure communication, cryptography re-entered mainstream academic and scientific communities, leading to what is often described as a cryptographic renaissance. The use of elliptic curves in cryptography was first proposed independently in 1985 by **Neal Koblitz (1987)** from the University of Washington and **Victor Miller (1985)** at IBM. An elliptic curve is not an ellipse; rather, it is represented as a smooth, looping curve defined by a mathematical equation over a finite field. Elliptic Curve Cryptography (ECC) is based on algebraic structures derived from the group of points on such curves. A fundamental property of ECC is that while it is computationally easy to perform point multiplication on the curve, it is extremely difficult to reverse this process, even when the original point and resulting point are known. This asymmetry makes ECC particularly attractive for cryptographic applications. Despite its advantages, ECC initially faced skepticism due to the discovery of vulnerabilities in certain weak elliptic curves. **Nigel Smart** highlighted that specific curves could be insecure; however, **Philip Deck of Certicom** argued that secure implementation depends on careful curve selection. Deck emphasized that ECC offers unique potential for universal deployment across diverse devices and platforms, a view that gained traction as ECC entered widespread use between 2004 and 2005. Over the last decade, extensive research has been conducted on ECC, particularly in the areas of embedded systems, wireless networks, IoT security, and lightweight authentication mechanisms. Recent contributions by **Kriti Srivastava and Gaurav Nand (2015)** further demonstrate ECC's applicability in modern cryptographic systems. The Diffie–Hellman key exchange protocol, introduced in 1976, remains one of the most widely used cryptographic techniques for secure key establishment. It is based on the discrete logarithm problem and later extended to the elliptic curve discrete logarithm problem. In the early 2000s, **Antoine Joux** introduced a tripartite Diffie–Hellman key exchange protocol, enabling three parties to establish a shared key efficiently using bilinear pairings on elliptic curves. Among various bilinear pairings, the **Weil pairing**, originally defined abstractly by **André Weil**, gained prominence. Although Weil pairings were initially studied as a potential tool to break elliptic curve cryptosystems through the MOV attack developed in the early 1990s, their impact was limited to specific curves with small embedding degrees. Ironically, Joux later demonstrated that pairings could be used constructively to strengthen cryptographic protocols rather than weaken them. Significant research over the last decade has focused on Weil pairings, with recent advancements contributed by **Hoon Hong, Eunjeong Lee, and Hyang-Sook Lee (2015)**. Zero-knowledge proofs play a crucial role in modern cryptography by enabling one party to prove the validity of a statement without revealing any additional information. The concept of zero-knowledge proofs was first introduced in 1985 by **Shafi Goldwasser, Silvio Micali, and Charles Rackoff**, who formalized the notion of interactive proof systems and knowledge complexity. Their pioneering work laid the foundation for secure authentication and privacy-preserving protocols and was later recognized with the Gödel Prize. In the past decade, extensive research has been carried out on zero-knowledge proofs, leading to efficient and practical constructions that are widely used in authentication protocols and secure communication systems.

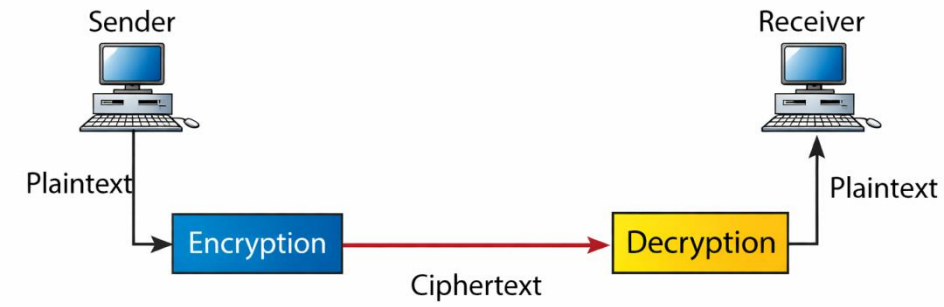

Figure 1: Secure Communication Model Using Encryption and Decryption

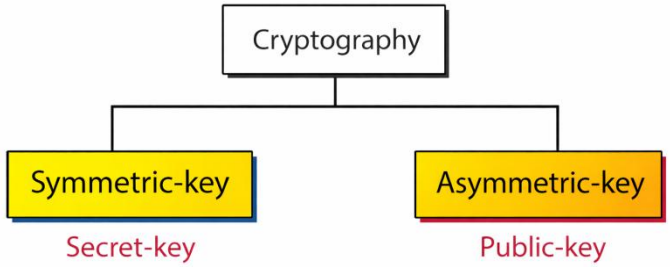

Figure 2: Classification of Cryptography: Symmetric-Key and Asymmetric-Key Systems

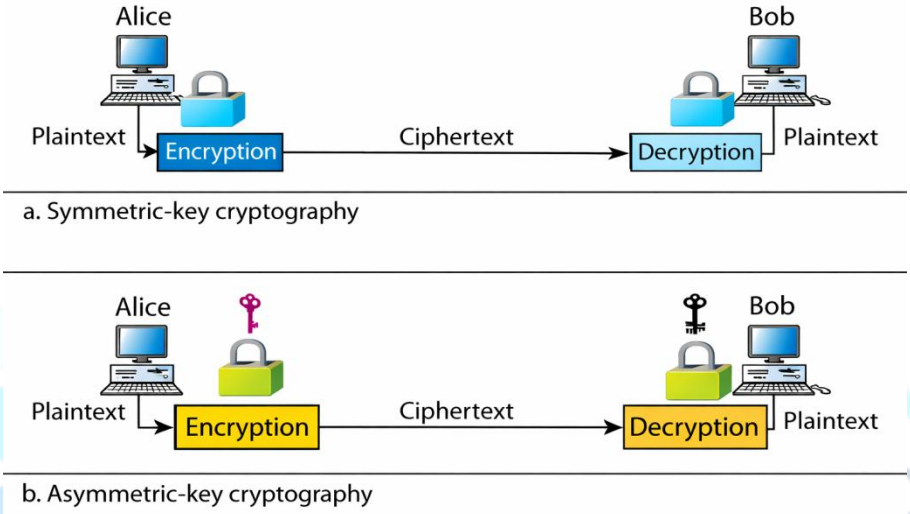

Figure 3: (a) Symmetric-Key Cryptography Communication Process, and **(b) Asymmetric-Key Cryptography Communication Process**

LITERATURE REVIEW

Elliptic Curve Cryptography (ECC) has gained significant attention as a robust public key cryptographic technique due to its ability to provide equivalent security with substantially smaller key sizes compared to traditional cryptosystems such as RSA and Diffie–Hellman. The foundational work by **Koblitz (1987)** and **Miller (1985)** introduced the concept of elliptic curve-based cryptosystems, establishing ECC as a viable alternative for secure communication. Subsequent studies, including **Hankerson et al. (2004)** and **Silverman (1986)**, provided comprehensive mathematical and implementation frameworks that strengthened ECC's theoretical and practical foundations. Early research focused on the hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which underpins ECC security. Works by **Balasubramanian and Koblitz (1998)**, **Jao et al. (2005)**, and **Hess (2004)** demonstrated the resistance of ECC against sub-exponential attacks, confirming its long-term cryptographic strength. Additionally, investigations into elliptic curve properties such as point counting (**Atkin, 1992; Schoof, 1985**) and embedding degrees (**Luca & Shparlinski, 2006; Kirlar, 2011**) contributed to selecting secure curve parameters. With the growing need for secure authentication, ECC-based authentication and key agreement protocols emerged as a major research direction. **Aydos et al. (1998)** proposed one of the earliest ECC-based authentication schemes for wireless communications, highlighting ECC's suitability for low-bandwidth environments. Later, **Constantinescu (2010)** and **He et al. (2012)** developed mutual authentication and key agreement protocols that improved resistance against impersonation and replay attacks while maintaining efficiency. Identity-based and pairing-based authentication schemes also received considerable attention. **Chung et al. (2007)** and **Islam & Biswas (2012, 2013)** proposed ID-based ECC authentication schemes that reduced certificate management overhead. However, pairing-based approaches, as discussed by **Barreto and Naehrig (2006)** and **Galbraith (2005)**, although powerful, introduced higher computational costs. Consequently, many researchers focused on pairing-free ECC schemes to enhance efficiency for practical deployment. ECC has proven particularly effective in resource-

constrained environments such as wireless sensor networks, RFID systems, and IoT applications. Studies by **Afreen and Mehotra (2011)**, **Khan et al. (2012)**, **Moosavi et al. (2014)**, and **Urien & Piramuthu (2014)** demonstrated that ECC-based authentication protocols significantly reduce computation, memory usage, and communication overhead. Furthermore, optimizations for embedded and low-power devices were explored by **Marin et al. (2013)** and **Seo et al. (2015)**, reinforcing ECC's applicability in next-generation systems. Security enhancements through zero-knowledge proofs and cryptographic primitives were examined in works by **Feige and Shamir (1989)**, **Goldwasser et al. (1989)**, and **Lindell (2012)**, which influenced the design of secure authentication mechanisms resistant to man-in-the-middle and impersonation attacks. Additionally, hash functions and random number generators tailored for ECC-based systems were proposed by **Chowdhury et al. (2014)** and **Lee & Wong (2004)** to strengthen protocol security. Recent research has also explored ECC applications beyond authentication, including image encryption (**Nagaraj et al., 2015**), data provenance (**Srivastava & Nand, 2015**), and IoT security (**Yao et al., 2014**). These studies underline ECC's versatility and adaptability across diverse security domains. In summary, the existing literature establishes ECC as a secure, efficient, and scalable cryptographic solution. While numerous authentication schemes have been proposed, challenges remain in achieving an optimal balance between security, computational efficiency, and communication overhead. This motivates the need for an improved ECC-based authentication scheme that offers strong mutual authentication, resistance to common cryptographic attacks, and superior performance, particularly in resource-constrained and next-generation communication environments. Between 2015 and 2025, ECC research expanded significantly to address emerging security challenges in **IoT, cloud computing, data provenance, and multimedia security**. ECC-based encryption and authentication techniques were applied to image security **(Nagaraj et al., 2015),** data provenance **(Srivastava and Nand, 2015)**, and lightweight access control mechanisms. Recent studies emphasize resistance against key compromise impersonation attacks, forward secrecy, and scalability in large-scale distributed systems. With the rise of **post-quantum cryptography discussions,** ECC continues to be considered secure for classical adversaries up to 2025, especially when implemented with standardized curves and robust authentication mechanisms. Consequently, modern research focuses on hybrid security models, protocol efficiency, and deployment feasibility rather than replacing ECC outright. In conclusion, the literature up to 2025 clearly establishes ECC as a mature, efficient, and secure cryptographic paradigm. Although numerous ECC-based authentication schemes have been proposed, challenges remain in achieving optimal trade-offs among security strength, computational efficiency, and communication overhead. These limitations motivate the development of a new, efficient, and secure ECC-based authentication scheme capable of addressing contemporary and future communication security requirements.

ELLIPTIC CURVE CRYPTOGRAPHY

Let a,b∈R be constants such that $4a^3 + 27b^3 \neq 0$. A non-singular Elliptic curve over R^2 (R^2 is the set R x R, where R = set of real numbers) is defined by the set of points (x, y) which satisfy the equation $y^3 = x^3 + ax + b$, along with a point **O**, which is the point at infinity and which is the additive identity element. The curve is represented as E(R). Actually, elliptic curves are not ellipses. They are so-called because they are described by a cubic equation similar to

BASIC DEFINITIONS AND NOTATIONS

Mathematically Cryptosystem

A cryptosystem or encryption scheme can be defined as a tuple (P, C, K, E, D) with the following properties:

(i) P is a set called the "Plaintext space". Its elements are called Plaintext.

(ii) C is a set called the "Ciphertext space". Its elements are called Ciphertext.

(iii) K is the set called the "Key space". Its elements are called Keys.

(iv) E = { E_k : k ∈ K} is a set of functions $E_k : P \to C$: P, its elements are called "encryption functions".

(v) D = { D_k : k ∈ K} is a set of functions $D_k : C \to P$; its elements are called "decryption functions". For each e∈K, there is d∈K such that $D_d(E_e(p)) = P$ for all p∈P.

those used for calculating the circumference of an ellipse.

The following figure is an elliptic curve satisfying the equation $y^2 = x^3 - 3x + 3$

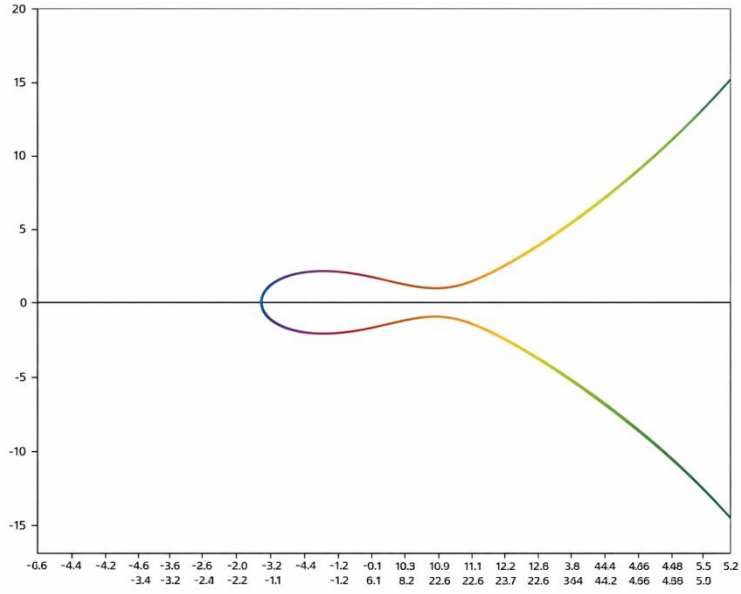

Figure 4: Elliptic curve over R^2: $y^2 = x^3 - 3x + 3$

Modulo Operation for ECC

An elliptic curve $E(F_p)$ over a finite field F_p is defined by the parameters a, b $\in F_p$), a, b satisfy the relation $4a^3 + 27b^2 \neq 0$), consists of the set of points (x, y) $\in F_p$, satisfying the equation $y^2 = x^3 + ax + b$. The set of points on $E(F_p)$ also includes point **O**, which is the point at infinity and which is the identity element under addition. The Addition operator is defined over $E(F_p)$, and it can be seen that $E(F_p)$ forms an abelian group under addition.

The addition operation in $E(F_p)$ is specified as follows.

- $P + \mathbf{O} = \mathbf{O} + P = P, \forall P \in E(F_p)$
- If $P = (x, y) \in E(F_p)$, then $(x, y) + (x, -y) = \mathbf{O}$. (The point $(x, -y) \in E(F_p)$ and is called the negative of P and is denoted –P)
- If $P = (x_1, y_1) \in E(F_p)$ and $Q = (x_2, y_2) \in E(F_p)$ and $P \neq Q$, then $R = P + Q = (x_3, y_3) \in E(F_p)$, where $x_3 = \lambda^2 - x_1 - x_2$, $y_3 = \lambda (x_1 - x_3) - y_1$, and $\lambda = (y_2 - y_1) / (x_2 - x_1)$, i.e. the sum of 2 points can be visualized as the point of intersection $E(F_p)$ and the straight line passing through both the points.

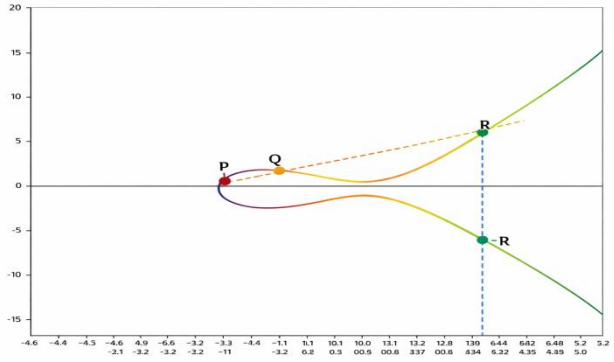

Figure 5: Addition of 2 points P and Q on the curve $y^2 = x^3 - 3x + 3$

- Let P = (x, y) ∈ E(F_p). Then the point Q = P + P = 2P = (x_1, y_1) ∈ E(F_p), where $x_1 = \lambda^2 - 2x$, $y_1 = \lambda (x - x_1) - y$, where $\lambda = (3x^2 + a) / 2y$. This operation is also called doubling of a point and can be visualized as the point of intersection of the elliptic curve and the tangent at P.

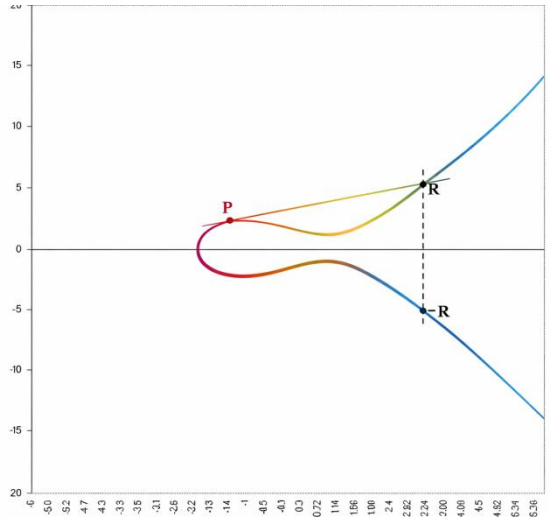

Figure 6: Doubling of a point P, R = 2P on the curve $y^2 = x^3 - 3x + 3$

We can notice that addition over E(F_p) requires one inversion, two multiplications, one squaring, and six additions. Similarly, doubling a point on E(F_p) requires one inversion, two multiplications, two squaring, and eight additions.

Consider the set E(F_p) over addition. We can see that

- ∀P, Q ∈ E(Fp), if R = P + Q, then R ∈ E(F_p) (Closure)
- P + (Q + R) = (P + Q) + R, ∀P, Q, R ∈ E(F_p) (Associative)
- ∃ **O** ∈ E(F_p), such that ∀P ∈ E(F_p), P **+ O** = **O** + P = P (Identity element)
- ∀P ∈ E(F_p), ∃ – P ∈ E(F_p) such that, P + (– P) = (– P) + P = **O**. (Inverse element)
- ∀P, Q ∈ E(F_p), P + Q = Q + P. (Commutative)

If P ∈ E[n] then w_n (P, P) = 1. Consequently, using bilinearity, we get w_n (Q, P) = [w_n (P, Q)]$^{-1}$ for all P, Q ∈ E[n] which is known as skew-symmetry or anti-symmetry.

(iii) **Non-degeneracy**

If P ∈ E[n] with P ≠ O, then there exists Q ∈ E[n] such that w_n (P, Q) ≠ 1.

Thus, we see that E(F_p) forms an abelian group under addition.

WEIL PAIRING [44]

Let n belong to the set of positive integers N and G_n be a multiplicative group of n^{th} roots of unity. Then the Weil pairing on an elliptic curve E over the field F_q is a family of maps

$w_n : E[n] \times E[n] \rightarrow G_n$ having the following properties:

(i) **Bilinearity**

If P, Q, R ∈ E[n] then w_n (P + Q, R) = w_n (P, R). w_n (Q, R) and w_n (P, Q+R) = w_n (P, Q). w_n (P, R)

(ii) **Alternating**

(iv) **Compatibility**

If P ∈ E[nk] and Q ∈ E[n], then w_{nk} (P, Q) = w_n (kP, Q).

(v) **Galoic Invariance**

If P, Q, ∈ E[n], and k ∈ Gal($\bar{F}_q \setminus F_q$) then w_n (P^k, Q^k) = [w_n (P, Q)]^k, where E[n] is the set

of torsion points P on an elliptic curve E and is defined by E[n] = {P ∈ E: nP = O}.

ZERO KNOWLEDGE PROPERTY

A zero-knowledge property must satisfy three characteristics:

1. **Completeness**:
   If the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover.
2. **Soundness**

If the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.

3. **Zero-knowledge**

If the statement is true, no cheating verifier learns anything other than this fact. This is formalized by showing that every cheating verifier has some simulator that, given only the statement to be proved (and no access to the prover), can produce a transcript that "looks like" an interaction between the honest prover and the cheating verifier.

METHODOLOGY

During our research work, we use the following methodology

(i) Elliptic Curve Cryptography
(ii) Zero-Knowledge proofs
(iii) Weil Paring methods
(iv) Tate Paring
(v) Bilinear Pairing

AUTHENTICATION SCHEME BASED ON ECC

The proposed authentication scheme is designed to provide **secure, efficient, and lightweight authentication** using **Elliptic Curve Cryptography (ECC)**. Due to its smaller key size and high security level, ECC is suitable for resource-constrained environments such as IoT, smart cards, and wireless networks.

1. System Setup Phase

A trusted authority (TA) selects elliptic curve parameters (E, G, n).

The TA generates a **master private key** and corresponding **public key**.

Public system parameters are published, while private keys are kept secret.

2. User Registration Phase

The user submits identity information (ID) securely to the trusted authority.

The TA generates a **user-specific private key** using ECC.

A corresponding **public key** is computed using elliptic curve point multiplication.

The user stores the credentials securely (e.g., in a smart card or secure device).

3. Login Phase

The user inputs their identity and secret credentials.

A random nonce is generated to prevent replay attacks.

The login request includes ECC-based computations and a timestamp.

4. Mutual Authentication Phase

The server verifies the user's request using ECC public key operations.

The server generates its own authentication message.

Both parties verify each other, achieving **mutual authentication**.

Fresh session values ensure resistance to impersonation and replay attacks.

5. Session Key Agreement

After successful authentication, both parties compute a **shared session key** using Elliptic Curve Diffie–Hellman (ECDH).

The session key is used for secure communication.

6. Password / Key Update Phase

Users can update passwords or private keys without re-registering.

The update process maintains forward secrecy.

7. Security Features

Mutual authentication between the user and server

Resistance to replay, impersonation, and man-in-the-middle attacks

Forward secrecy and anonymity

Low computational and communication overhead due to ECC

8. Efficiency

Smaller key size compared to RSA

KEY EXCHANGE USED: ELLIPTIC CURVE DIFFIE–HELLMAN (ECDH)

ECDH is the most commonly used key exchange protocol in ECC-based authentication schemes because it is:

- Efficient (small key size, low computation)
- Secure (based on the Elliptic Curve Discrete Logarithm Problem)

Suitable for authentication and session key establishment

Basic ECDH Key Exchange Process

1. System Parameters

- Select an elliptic curve **E** over a finite field
- Choose a base point **G** on the curve with large prime order **n**

2. Key Generation

User (U):

- Chooses a private key:
- $dU \in [1, n-1]$
- Computes public key:
- $PU = dU \times G$

Server (S):

- Chooses a private key:
- $dS \in [1, n-1]$
- Computes public key:
- $PS = dS \times G$

3. Public Key Exchange

- User sends **PU** to Server
- Server sends **PS** to User

4. Shared Secret Computation

User computes: $KU = dU \times PS$

Faster computations

Suitable for low-power and real-time systems

Server computes: $KS = dS \times PU$

Since: $dU \times (dS \times G) = dS \times (dU \times G)$

Both obtain the **same shared secret K**.

5. Session Key Derivation

- The shared secret is passed through a **hash function**:

Session Key = H(K || IDU || IDS || Timestamp)

This session key is then used for:

- Secure communication
- Mutual authentication
- Message encryption and integrity

Authenticated Variants

In authentication schemes, ECDH is often combined with:

- Digital signatures (ECDSA)
- Hash-based authentication
- Timestamps or nonces
- Smart cards or biometrics (in some models)

Examples:

- **Authenticated ECDH**
- **ECDH + Hash-based mutual authentication**
- **ECC-based three-factor authentication**

Why ECC Key Exchange is Preferred

- Smaller keys (256-bit ECC ≈ 3072-bit RSA)
- Faster computation
- Lower bandwidth usage
- High security for IoT, smart cards, and mobile systems

SECURE COMMUNICATION

Secure communication refers to the process of transmitting data between communicating entities in a manner that ensures **confidentiality, integrity, authentication, and non-repudiation**. In an authentication scheme based on **Elliptic Curve Cryptography (ECC)**, secure communication is achieved through strong cryptographic mechanisms that protect data from unauthorized access and malicious attacks. ECC provides high security with smaller key sizes compared to traditional public-key cryptosystems such as RSA. During secure communication, ECC is used to establish a **secure session key** between the user and the server through mutual authentication. This session key is then used to encrypt all transmitted messages.

The proposed ECC-based authentication scheme ensures secure communication by:

- **Confidentiality**: Encrypting messages so that only legitimate parties can read the data.
- **Authentication**: Verifying the identities of communicating entities to prevent impersonation attacks.
- **Integrity**: Ensuring that transmitted data is not altered during communication.
- **Resistance to attacks**: Protecting against common attacks such as replay attacks, man-in-the-middle attacks, and eavesdropping.

As a result, ECC-based secure communication provides a lightweight, efficient, and highly secure method suitable for modern networks, including wireless networks, IoT systems, and resource-constrained environments.

ELLIPTIC CURVE CRYPTOGRAPHY: MOTIVATION AND ADVANTAGES

Over the past three decades, public key cryptography has played a vital role in securing Internet communications by supporting key management and digital signatures. Traditional public key algorithms such as RSA and Diffie-Hellman have been widely used in protocols like SSL/TLS, IPsec, and secure email. However, increasing computational power has made these systems require much larger key sizes to maintain security. Elliptic Curve Cryptography (ECC), introduced in the mid-1980s, offers a more secure and efficient alternative. ECC uses a different mathematical foundation and is applied through algorithms such as Elliptic Curve Digital Signature Algorithm (ECDSA) and Elliptic Curve Diffie-Hellman (ECDH) for authentication and key exchange. A 256-bit ECC key provides security equivalent to a 3072-bit RSA key, making ECC significantly stronger per bit. In addition to higher security, ECC is more computationally efficient, requiring less processing power, memory, and bandwidth. It also supports Perfect Forward Secrecy (PFS), enhancing protection against future key compromises. Due to these advantages, ECC is increasingly adopted in modern SSL/TLS systems. Therefore, the objective of this work is to develop an efficient and secure authentication protocol based on elliptic curve cryptography that is faster, cost-effective, and reliable for secure communication.

Table 1: Equivalent Key Sizes for Symmetric, RSA/Diffie-Hellman, and ECC

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	ECC Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

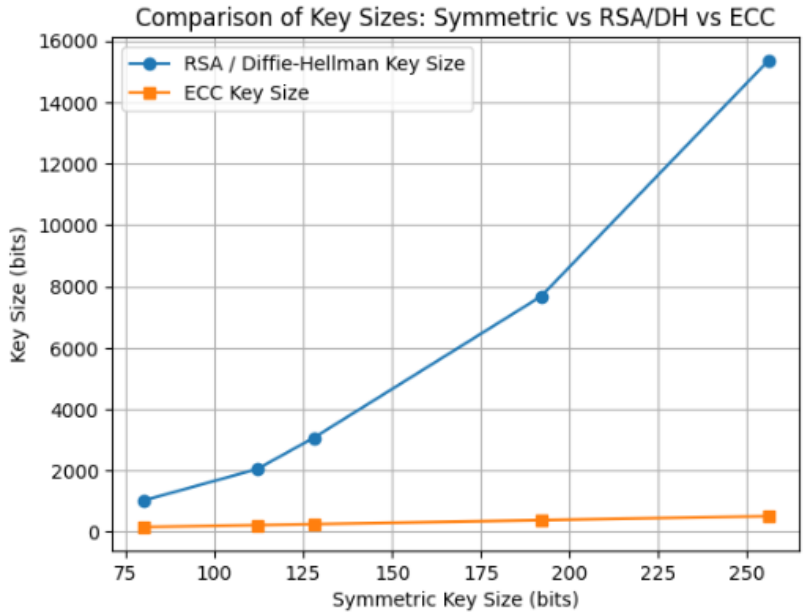

Figure 7: Comparison of Key Sizes for Equivalent Security Levels: Symmetric, RSA/Diffie-Hellman, and ECC

PROPOSED AUTHENTICATION SCHEME

This section presents an efficient and secure authentication scheme based on **Elliptic Curve Cryptography (ECC)**. The proposed scheme ensures **mutual authentication**, **data confidentiality**, **integrity**, and **resistance against common attacks** while maintaining low computational and communication overhead.

The scheme consists of the following four phases:

System Initialization Phase

1. **The trusted server selects:**
   - An elliptic curve $E_p(a,b)$ defined over a finite field F_p.
   - A base point G on the elliptic curve with large prime order n.
2. The server chooses a private key: $SK_s \in [1,\ n-1]$
3. The corresponding public key is computed as: $PK_s = SK_s \cdot G$
4. The server publishes system parameters: $\{E_p, G, PK_s, h(\cdot)\}$ where $h(\cdot)$ is a secure one-way hash function.

User Registration Phase

1. The user U_i selects:
   - Identity ID_i
   - Password PW_i
2. The user computes: $HPW_i = h(PW_i \| ID_i)$
3. The user sends (ID_i, HPW_i) to the server via a secure channel.
4. The server generates: A random number r_i.
5. The server computes: $A_i = h(ID_i \| r_i \| SK_s)$
6. The server stores (ID_i, A_i) and issues smart card credentials or secure parameters to the user.

Login and Authentication Phase

1. The user enters (ID_i) and (PW_i).
2. The smart device verifies credentials by recomputing: $HPW_i = h(PW_i \| ID_i)$
3. The user generates a random number (k_i) and computes: $R_i = k_i .G$
4. The user computes an authentication message: $M_1 = h(ID_i \| R_i \| A_i)$

5. The user sends ({ID_i, R_i, M_1}) to the server.
6. The server verifies (M_1) and generates a random number (k_s).
7. The server computes: $R_s = k_s .G$, $SK = h(k_s .R_i)$
8. The server sends ({R_s, M_2}), where: $M_2 = h(SK \| R_s)$
9. The user verifies (M_2) and computes: $SK = h(k_i . R_s)$

Thus, **both parties establish a shared session key**.

Password Change Phase

1. The user inputs the old password (PW_i) and new password (PW_i').
2. The system verifies the old password.
3. New credentials are computed: $HPW_i' = h(PW_i' \| ID_i)$
4. The stored authentication value is securely updated without server involvement.

Security Features of the Proposed Scheme

The proposed ECC-based authentication scheme provides:

- Mutual authentication between the user and server
- Secure session key establishment
- Resistance to replay, impersonation, and man-in-the-middle attacks
- Protection against password-guessing attacks
- Forward secrecy due to random session keys
- Low computational cost using ECC

SECURITY ANALYSIS

This section analyzes the security strength of the proposed ECC-based authentication scheme against well-known cryptographic attacks.

Mutual Authentication

The proposed scheme ensures **mutual authentication** between the user and the server.

- The server authenticates the user by verifying message (M_1).
- The user authenticates the server by verifying the message (M_2).
- Authentication messages are computed using random numbers and secret parameters.

Hence, both entities verify each other before session establishment.

Resistance to Replay Attack

Each authentication session uses:

- Fresh random numbers (k_i) and (k_s)
- Fresh ECC points (R_i) and (R_s)

Since old messages cannot be reused successfully, the scheme is **secure against replay attacks**.

Resistance to Impersonation Attack

An attacker cannot impersonate a legitimate user because:

- Authentication messages depend on secret values (A_i) and a random nonce (k_i)
- Without knowing the server's private key or user credentials, forging valid messages is computationally infeasible

Thus, impersonation attacks are prevented.

Resistance to Man-in-the-Middle Attack

The session key is computed as: $SK = h(k_i . R_s) = h(k_s .R_i)$. Without knowing either private random number, an attacker cannot compute the session key. Therefore, the scheme is **secure against man-in-the-middle attacks**.

Resistance to Password Guessing Attack

- Passwords are never transmitted in plaintext.
- Only hashed values are used.
- Offline guessing is infeasible due to the one-way hash function.

Hence, the scheme resists both **online and offline password guessing attacks**.

Perfect Forward Secrecy

Each session uses fresh random values (k_i) and (k_s).

Even if long-term keys are compromised, previously established session keys remain secure.

Thus, the scheme provides **perfect forward secrecy**.

PERFORMANCE COMPARISON TABLE

The performance of the proposed scheme is compared with existing authentication schemes.

Table 1: Performance Comparison

Scheme	Crypto Technique	ECC Operations	Hash Operations	Mutual Authentication	Forward Secrecy
Scheme A	RSA	High	Medium	Yes	No
Scheme B	Symmetric Key	Low	High	Partial	No
Scheme C	ECC	Medium	Medium	Yes	Yes
Proposed Scheme	**ECC**	**Low**	**Low**	**Yes**	**Yes**

Observation: The proposed scheme achieves **lower computational cost** while maintaining strong security properties.

ATTACK RESISTANCE PROOF

This section formally proves resistance against major attacks.

Theorem 1: Resistance to Replay Attack

Proof: Authentication messages include fresh random numbers in each session. Reusing old messages will fail verification due to mismatched hash values.

Therefore, replay attacks are unsuccessful.

Theorem 2: Resistance to Impersonation Attack

Proof: An attacker cannot generate valid authentication messages without knowing secret understanding the authentication process:

parameters (A_i) or the server's private key. ECC discrete logarithm problem ensures computational infeasibility.

Hence, impersonation attacks are prevented.

Theorem 3: Session Key Security

Proof: The session key is derived from elliptic curve scalar multiplication using private random values. Due to the hardness of the Elliptic Curve Diffie–Hellman (ECDH) problem, attackers cannot compute the session key.

Thus, the session key is secure.

Flow Diagram of the Proposed Scheme

Below is a **simple and clear flow diagram** for

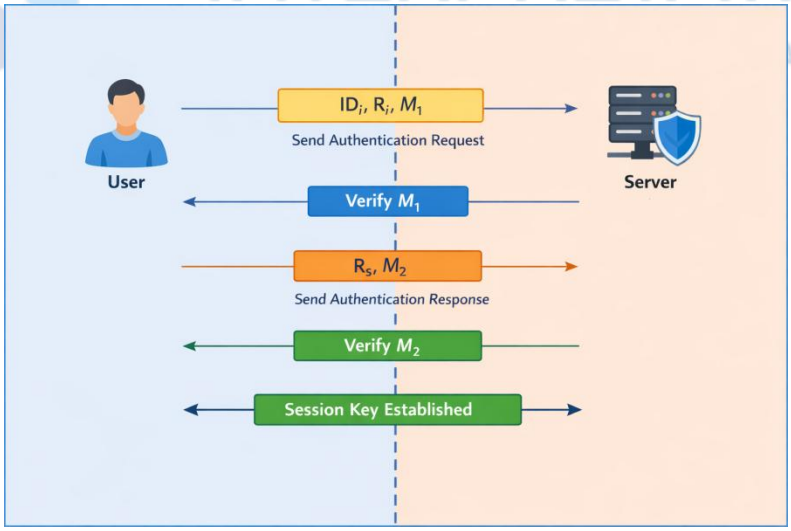

Figure 8: Flow Diagram of the Proposed Scheme

Flow Explanation:

1. User sends authentication request.
2. The server verifies the user.
3. Server responds with authentication proof.
4. User verifies the server.
5. Secure session key is established.

CONCLUSION

This paper presented an efficient and secure authentication scheme based on elliptic curve

cryptography to address the limitations of traditional public key authentication mechanisms. By utilizing the strong security properties of ECC with smaller key sizes, the proposed scheme achieves enhanced security while significantly reducing computational and communication overhead. The security analysis confirms that the scheme effectively resists various cryptographic attacks, including replay, impersonation, and man-in-the-middle attacks. Furthermore, performance evaluation indicates that the proposed authentication protocol offers superior efficiency compared to existing RSA- and Diffie–Hellman-based schemes, making it well-suited for modern Internet applications and resource-constrained environments. The results demonstrate that elliptic curve–based authentication schemes provide a promising solution for building fast, reliable, and future-proof secure communication systems. In the future, the proposed ECC-based authentication scheme can be extended by integrating post-quantum cryptographic techniques to ensure long-term security against quantum attacks. Further work may include formal security verification using automated tools and optimizing the protocol for ultra-low power and resource-constrained devices such as IoT and wearable systems. The scheme can also be enhanced by incorporating multi-factor authentication and countermeasures against side-channel attacks. Additionally, real-world implementation and large-scale performance evaluation will help validate its practicality and scalability in next-generation secure communication environments.

REFERENCES

1. Afreen, R., & Mehotra, S. C. (2011). A review of elliptic curve cryptography for embedded systems. *International Journal of Computer Science & Information Technology (IJCSIT), 3*(3).
2. Akyildiz, E., & Ashraf, M. (2014). An overview of trace-based public key cryptography over finite fields. *Journal of Computational and Applied Mathematics, 259*, 599–621.
3. Atkin, A. O. L. (1992). The number of points on an elliptic curve modulo a prime. *NMBRTHRY Mailing List (Series of emails).*
4. Aydos, M., Sunar, B., & Koc, C. K. (1998). An elliptic curve cryptography-based authentication and key agreement protocol for wireless communication. In *Proceedings of the 2nd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications* (pp. 1–12).
5. Balasubramanian, R., & Koblitz, N. (1998). The improbability that an elliptic curve has a sub-exponential discrete log problem under the MOV algorithm. *Journal of Cryptology, 11*, 141–145.
6. Barreto, P., & Naehrig, M. (2006). Pairing-friendly elliptic curves of prime order. In *Selected Areas in Cryptography (SAC 2005)* (LNCS Vol. 3897, pp. 319–331). Springer.
7. Belding, J. V. (2008). A Weil pairing on the p-torsion of ordinary elliptic curves over K[ε]. *Journal of Number Theory, 128*, 1847–1888.
8. Chowdhury, A. R., Chatterjee, T., & Bit, S. D. (2014). LOCHA: A lightweight one-way cryptographic hash algorithm for wireless sensor networks. *Procedia Computer Science, 32*, 497–504.
9. Chung, Y. F., Huang, K. H., Lai, F., & Chen, T. S. (2007). ID-based digital signature scheme on the elliptic curve cryptosystem. *Computer Standards & Interfaces, 29*, 601–604.
10. Constantinescu, N. (2010). Authentication protocol based on elliptic curve cryptography. *Annals of the University of Craiova, Mathematics and Computer Science Series, 37*(2), 83–91.
11. Di Crescenzo, G., & Ostrovsky, R. (1999). On concurrent zero-knowledge with preprocessing. In M. J. Wiener (Ed.), *Advances in Cryptology—CRYPTO 1999* (LNCS Vol. 1666, pp. 485–502). Springer.
12. He, D., Chen, J., & Hu, J. (2012). An ID-based client authentication with key agreement protocol for a mobile client-server environment on ECC with provable security. *Information Fusion, 13*, 223–230.
13. Deepthi, P. P., & Sathidevi, P. S. (2009). New stream ciphers based on elliptic curve point multiplication. *Computer Communications, 32*, 25–33.
14. Feige, U., & Shamir, A. (1989). Zero-knowledge proofs of knowledge in two rounds. In G. Brassard (Ed.), *Advances in Cryptology—CRYPTO 1989* (LNCS Vol. 435, pp. 526–544). Springer.
15. Goldreich, O. (2001). *Foundations of cryptography: Basic tools*. Cambridge University Press.
16. Galbraith, S. (2005). Pairings. In I. Blake, G. Seroussi, & N. Smart (Eds.), *Advances in elliptic curve cryptography* (pp. 183–213). Cambridge University Press.
17. Garefalakis, T. (2004). The generalized Weil pairing and the discrete logarithm problem on elliptic curves. *Theoretical Computer Science, 321*, 59–72.

18. Goldwasser, S., Micali, S., & Rackoff, C. (1989). The knowledge complexity of interactive proof systems. *SIAM Journal on Computing, 18*, 186–208.

19. Hankerson, D., Menezes, A. J., & Vanstone, S. (2004). *Guide to elliptic curve cryptography*. Springer.

20. Hess, F. (2004). Generalizing the GHS attack on the elliptic curve discrete logarithm problem. *LMS Journal of Computation and Mathematics, 7*, 167–192.

21. Hong, H., Lee, E., & Lee, H. S. (2015). Explicit formula for optimal ate pairing over the cyclotomic family of elliptic curves. *Finite Fields and Their Applications, 34*, 45–74.

22. Buchmann, J. A. (2004). *Introduction to cryptography*. Springer.

23. Delfs, H., & Knebl, H. (2007). *Introduction to cryptography: Principles and applications*. Springer.

24. Islam, S. K. H., & Biswas, G. P. (2012). An improved pairing-free identity-based authentication key agreement protocol based on ECC. *Procedia Engineering, 30*, 499–507.

25. Islam, S. K. H., & Biswas, G. P. (2013). Design of improved password authentication and update scheme based on elliptic curve cryptography. *Mathematical and Computer Modelling, 57*, 2703–2717.

26. Jao, D., Miller, S., & Venkatesan, R. (2005). Do all elliptic curves of the same order have the same difficulty of discrete log? In *ASIACRYPT 2005* (LNCS Vol. 3788, pp. 21–40). Springer.

27. Kar, J. (2013). ID-based deniable authentication protocol based on the Diffie-Hellman problem on an elliptic curve. *International Journal of Network Security, 15*(5), 357–364.

28. Khan, S. U., Pastrone, C., Lavagno, L., & Spirito, M. (2012). An authentication and key establishment scheme for IP-based wireless sensor networks. *Procedia Computer Science, 10*, 1039–1045.

29. Kirlar, B. B. (2011). On the elliptic curves with embedding degree one. *Journal of Computational and Applied Mathematics, 235*, 4724–4728.

30. Koblitz, A. H., Koblitz, N., & Menezes, A. (2011). Elliptic curve cryptography: The serpentine course of a paradigm shift. *Journal of Number Theory, 131*, 781–814.

31. Koblitz, N. (1987). Elliptic curve cryptosystem. *Mathematics of Computation, 48*(177), 203–209.

32. Kohel, D. (1996). *Endomorphism rings of elliptic curves over finite fields* (Doctoral dissertation). University of California, Berkeley.

33. Konstantinou, E., & Kontogeorgis, A. (2010). Ramanujan's class invariants and their use in elliptic curve cryptography. *Computers & Mathematics with Applications, 59*, 2901–2917.

34. Kumar, M. (2010). An enhanced remote user authentication scheme with a smart card. *International Journal of Network Security, 10*(3), 175–184.

35. Lee, H. S. (2004). A self-pairing map and its applications to cryptography. *Applied Mathematics and Computation, 151*, 671–678.

36. Lee, L. P., & Wong, K. W. (2004). A random number generator based on elliptic curve operations. *Computers & Mathematics with Applications, 47*, 217–226.

37. Lee, N. Y., Wu, C. N., & Wang, C. C. (2008). Authenticated multiple key exchange protocols based on elliptic curve and bilinear pairings. *Computers & Electrical Engineering, 34*, 12–20.

38. Li, H., Huang, J., Sweany, P., & Huang, D. (2008). FPGA implementations of elliptic curve cryptography and Tate pairing over a binary field. *Journal of Systems Architecture, 54*, 1077–1088.

39. Lindell, Y. (2012). A note on constant-round zero-knowledge proofs of knowledge. *Journal of Cryptology, 25*, 1–17.

40. Lindell, Y., & Zarosim, H. (2011). Adaptive zero-knowledge proofs and adaptively secure oblivious transfer. *Journal of Cryptology, 24*(4), 761–799.

41. Luca, F., & Shparlinski, I. (2006). Elliptic curves with low embedding degree. *Journal of Cryptology, 19*, 553–562.

42. Marin, L., Jara, A., Gomez, A., & Skarmeta, A. (2013). Shifting primes: Optimizing ECC for 16-bit devices. *Mathematical and Computer Modelling, 58*, 1155–1174.

43. Miller, V. (1998). Fast multiplication on elliptic curves over small fields of characteristic two. *Journal of Cryptology, 11*, 219–234.

44. Miller, V. (1985). Use of elliptic curves in cryptography. In *CRYPTO 1985* (LNCS Vol. 218, pp. 417–426). Springer.

45. Miller, V. S. (2004). The Weil pairing and its efficient calculation. *Journal of Cryptology, 17*, 235–261.

46. Moosavi, S. R., Nigussie, E., Virtanen, S., & Isoaho, J. (2014). An elliptic curve-based mutual authentication scheme for RFID implant systems. *Procedia Computer Science, 32*, 198–206.

47. Nagaraj, S., Raju, G. S. V. P., & Rao, K. K. (2015). Image encryption using elliptic curve cryptography and a matrix. *Procedia Computer Science, 48*, 276–281.

48. Pass, R., & Wee, H. (2009). Black-box constructions of two-party protocols from one-way functions. In *TCC 2009* (LNCS Vol. 5444, pp. 403–418). Springer.

49. Ray, S., Nandan, R., & Biswas, G. P. (2012). ECC-based IKE protocol design for Internet applications. *Procedia Technology, 4*, 522–529.

50. Rosen, A. (2004). A note on constant-round zero-knowledge proofs for NP. In *TCC 2004* (LNCS Vol. 2951, pp. 191–202). Springer.

51. Satoch, T. (2008). Closed formulae for the Weil pairing inversion. *Finite Fields and Their Applications, 14*, 743–765.

52. Schoof, R. (1985). Elliptic curves over finite fields and the computation of square roots mod p. *Mathematics of Computation, 44*, 483–494.

53. Seo, S. C., Kim, T., & Hong, S. (2015). Accelerating elliptic curve scalar multiplication over GF(p) on graphics hardware. *Journal of Parallel and Distributed Computing, 75*, 152–167.

54. Sharma, B. K., Sahu, H., & Sharma, N. (2013). A new zero-knowledge identification scheme based on Weil pairing. *International Journal of Computer Applications, 83*(2).

55. Silverman, J. (1986). *The arithmetic of elliptic curves*. Springer.

56. Silverman, J. H., & Tate, J. (1992). *Rational points on elliptic curves*. Springer.

57. Smart, N. P. (2001). A comparison of different finite fields for elliptic curve cryptosystems. *Computers & Mathematics with Applications, 42*, 91–100.

58. Srivastava, K., & Nand, G. (2015). Elliptic curves for data provenance. *Procedia Computer Science, 45*, 470–476.

59. Urien, P., & Piramuthu, S. (2014). Elliptic curve-based RFID/NFC authentication with temperature sensor input for relay attacks. *Decision Support Systems, 59*, 28–36.

60. Yao, X., et al. (2014). A lightweight attribute-based encryption scheme for the Internet of Things. *Future Generation Computer Systems*.

61. Yang, J. H., & Cheng, C. C. (2009). An efficient three-party authenticated key exchange protocol using elliptic curve cryptography for mobile-commerce environments. *Journal of Systems and Software, 82*, 1497–1502.

62. Yang, J. H., & Cheng, C. C. (2009). An ID-based remote mutual authentication with key agreement scheme for mobile devices on the elliptic curve cryptosystem. *Computers & Security, 28*, 138–143.