International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

# Diffie-Hellman Key Exchange Based on Block Matrices Combined with Elliptic Curves

**Hiba Hilal Hadi and Ammar Ali Neamah**

**Submitted:** 24/01/2023 **Revised:** 13/03/2023 **Accepted:** 04/04/2023

**Abstract:** Various techniques were used to improve and propose cryptographic systems based on elliptic curves. In particular, the Diffie-Hellman key exchange protocol is widely used in most of these systems. In this work, we introduce an efficient approach based on the block matrices integrated with elliptic curves to increase the security of this protocol. With the new system, we can reduce key size without expanding the underlying elliptic curve. Therefore, the proposed protocol security will be more intractable since it will need to solve the Elliptic Curve Discrete Logarithm Problem more than once based on the chosen block matrices compared with the original protocol.

***Keyword:*** *Diffe-Hellman Key Exchange, Elliptic Curves, ECDLP*

## 1. Introduction

Information needs to be transmitted safely over insecure channels more than ever in the information era. The science of safely transferring and retrieving data over an unsecured route is known as cryptography [1]. The method used to distribute keys between communicating parties is significant in contemporary cryptography. Either secret or public key cryptography can be employed to distribute keys [2]. Secure communication relies on public key cryptography as its underlying technology. The Diffie-Hellman (DH) key exchange was introduced by Diffie and Hellman in 1976 as the first public key cryptographic scheme to secretly distribute keys [3]. This protocol employs a combination of keys (private and public keys). For instance, if Ali wishes to communicate with Ban, he uses his private key and Ban's public key to encrypt his communication. Ban decrypts the communication at the receiving end using her private key and Ali's public key [1]. The Discrete Logarithm Problem (DLP) is the foundation for the DH key exchange algorithm, which is founded on the challenge of computing logarithmic functions of prime exponents [4].

Elliptic curve cryptographic schemes (ECCs), which are based on the idea of public key cryptography, were independently introduced by Miller, 1986 [5]; Koblitz, 1987 [6]. This marked the beginning of an ongoing study into the application of ECCs for public-key protocols like the Diffie-Hellman key agreement. These systems are more computationally effective and provide greater security with reduced key sizes [7]. As a result, ECCs are an ideal cryptographic method for limited environments such as smart cards and wireless networks. The security of these systems depends on how difficult it is to solve the discrete logarithm on elliptic curves problem (ECDLP). The implementation of ECC with security and efficiency has been the subject of extensive research. Finite groups dependent on elliptic curves are very attractive because, despite numerous efforts, the best methods for solving ECDLP have exponential running times [8] [16].

In [9], J.-J. Climent et al. introduce a new mathematical problem that applies to public key cryptography. In particular, they developed a DH key exchange algorithm that combined elements of matrix algebra and the addition of

*Department of Mathematics, Faculty of Computer Science and Mathematics, University of Kufa, Najaf, Iraq*

elliptic curve points. F. Amounas et al. suggested a novel mapping technique using matrices and the elliptic curve [10]. They employed the properties of invertible matrices combined with elliptical curve points to provide a new mapping technique for encrypting/decrypting processes. A. Chillali et al. [11] recently designed public key exchange protocols over a noncommutative ring, specifically over the ring of the "elliptic" matrix, whose security is based on ECDLP [17][18].

The main idea of this work is to introduce a new group of block matrices dependent on elliptic curve points. In particular, we develop the ECDH key exchange protocol by employing the block matrices whose security will be based on ECDLP. This development can efficiently increase the security level of the protocol because the key size will be large enough to resist brute-force attacks. For example, if one selects the size of base matrix $2 \times 3$, the attacker needs to solve the ECDLP six times instead of once. This means the security of the new approach for ECDH key exchange will increase based on the size of the base matrix [19][20].

The remainder of this paper is structured as follows: Section 2 provides preliminaries of this study. We describe the development of ECDH key exchange protocol in the next section. In Section 4, the security of the suggested protocol is provided. Section 5 ends with a summary of the complete work.

## 2. Preliminaries

### 2.1 Elliptic Curve Cryptography (ECC)

The equation $\boldsymbol{y^2 = x^3 + ax + b}$, where $\boldsymbol{4a^3 + 27b^2}$ mod $\boldsymbol{p} \neq 0$, can be used to define elliptic curves over the prime field $\mathbf{F_p}$. By altering the values of the curves' $\boldsymbol{a}$ and $\boldsymbol{b}$, which belong to $\mathbf{F_p}$, different curves can be produced. A group, designated by the symbol $\mathbf{E(F_p)}$, is made up of all the points $(\boldsymbol{x}, \boldsymbol{y})$ that fulfill above equation when the addition operation is applied. For elliptic curves, this operation ($\oplus$) can form a group with a specific point $\boldsymbol{O}_\infty$, where this point is known as the point at infinity or the identity (*i.e.* ($\mathbf{E(F_p)}$,$\oplus$) be an group) [18]. If $\boldsymbol{P} = (\boldsymbol{x_1}, \boldsymbol{y_1})$ and $\boldsymbol{Q} = (\boldsymbol{x_2}, \boldsymbol{y_2})$ belongs to $\mathbf{E(F_p)}$, then $\boldsymbol{P} \oplus \boldsymbol{Q} = (\boldsymbol{x_3}, \boldsymbol{y_3})$ can be defined as follows:

- $\boldsymbol{P} \oplus \boldsymbol{Q} = \boldsymbol{O}_\infty$ if $\boldsymbol{P} = \boldsymbol{O}_\infty$ and $\boldsymbol{Q} = \boldsymbol{O}_\infty$.
- $\boldsymbol{P} \oplus \boldsymbol{Q} = \boldsymbol{P}$ or $\boldsymbol{Q}$ if $\boldsymbol{Q} = \mathbf{0}_\infty$ or $\boldsymbol{P} = \boldsymbol{O}_\infty$.
- $\boldsymbol{P} \oplus (\ominus \boldsymbol{P}) = \boldsymbol{O}_\infty$ if $\boldsymbol{Q} = \ominus \boldsymbol{P} = (\boldsymbol{x_1}, -\boldsymbol{y_1})$.
- Otherwise $\boldsymbol{P} \oplus \boldsymbol{Q} = (\boldsymbol{x_3}, \boldsymbol{y_3})$ where:

$$\boldsymbol{x_3} = \lambda^{\,2} - \boldsymbol{x_1} - \boldsymbol{x_2}, \quad \boldsymbol{y_3} = \lambda\,(\boldsymbol{x_1} - \boldsymbol{x_3}) - \boldsymbol{y_1},$$

and

$$\lambda = \begin{cases} \frac{3x_1^2}{2y_1} & \text{if } \boldsymbol{P} = \boldsymbol{Q} \\ \frac{y_2 - y_1}{x_2 - x_1} & \text{if } \mathbf{P} \neq \mathbf{Q} \end{cases}.$$

In ECCs, the public key, which has an order $\boldsymbol{n}$ point $\boldsymbol{P}$, is created by multiplying the secret key $\boldsymbol{k} \in [\mathbf{1}, \boldsymbol{n} - \mathbf{1}]$ that is an arbitrary integer. The point $\boldsymbol{P}$ is called the generator of a cyclic group $< \boldsymbol{P} > = \{\boldsymbol{O}_\infty, \boldsymbol{P}, [\mathbf{2}]\boldsymbol{P}, \ldots, [\boldsymbol{n} - \mathbf{1}]\boldsymbol{P}\}$ and its order is described as the smallest positive integer $\boldsymbol{n}$ such that $[\boldsymbol{n}]\boldsymbol{P} = \boldsymbol{O}_\infty$ which denoted by $\boldsymbol{ord}\ (\boldsymbol{P})$. The ECDLP is crucial to the ECCs' protection. This issue is as follows: If $\boldsymbol{P} \in \mathbf{E(E_p)}$ has order $\boldsymbol{n}$ and a point $\mathbf{Q} \in < \mathbf{P} >$, then finding $\boldsymbol{k} \in [1, \boldsymbol{n} - 1]$ such that $\boldsymbol{Q} = \boldsymbol{kP} = \underbrace{\boldsymbol{P} + \boldsymbol{P} + \boldsymbol{P} + \cdots + \boldsymbol{P}}_{\text{k times}}$ is known as the discrete logarithm of $\boldsymbol{Q}$ to the base $\boldsymbol{P}$ and is represented by $\boldsymbol{k} = \boldsymbol{log_P Q}$. Because $\boldsymbol{k}$ can be derived from $\boldsymbol{Q}$ if the ECDLP is simple, the ECDLP's complexity is critical to the security of these cryptographic systems [12] [21][22][23].

### 2.2 Elliptic Curve Diffie Hellman Key Exchange Protocol

In this paper, we are interested in Elliptic Curve Diffie Hellman (ECDH) key exchange protocol which differs from the standard DH in that it depends on ECDLP rather than DLP. ECDH is a key agreement protocol that enables two parties, A and B, who each have an elliptic curve private-public key pairs, to create a shared secret key over an unsecured channel [13]. Initially, the following domain parameters are chosen to share keys between two participants using ECDH: $\boldsymbol{D} = (\mathbf{F_q}, \mathbf{E}(\mathbf{E_p}), \boldsymbol{P}, \boldsymbol{n})$, where $\mathbf{E}(\mathbf{E}_q)$ is elliptic curve with parameters $\boldsymbol{a}, \boldsymbol{b}$, and $\mathbf{F_q}$ is finite field such that $\boldsymbol{q}$ is a prime or an integer of the form $\mathbf{2}^{\boldsymbol{m}}$, and $\boldsymbol{P}$ is a base point ($\boldsymbol{P}$ has prime order) on the elliptic curve with an order $\boldsymbol{n}$ [14]. Following is a description of an ECDH method for calculating a shared key:

- A randomly selects $\boldsymbol{d} \in [\mathbf{1}, \boldsymbol{n}-\mathbf{1}]$ and calculates key $\boldsymbol{dQ_B}$ = $(\boldsymbol{x_k}, \boldsymbol{y_k})$, where $\boldsymbol{d}$ is A's private key and $\boldsymbol{Q_B} = \boldsymbol{eP}$ is B's public key.
- B randomly selects $\boldsymbol{e} \in [\mathbf{1}, \boldsymbol{n}-\mathbf{1}]$ and calculates key $\boldsymbol{eQ_A}$ = $(\boldsymbol{x_k}, \boldsymbol{y_k})$, where $\boldsymbol{e}$ is B's private key, and $\boldsymbol{Q_A} = \boldsymbol{dP}$ is A's public key.
- Since $\boldsymbol{dQ_B} = \boldsymbol{deP} = \boldsymbol{edP} = \boldsymbol{eQ_A}$, then the shared secret computed by both sides is equal.

### 2.3 The Group $\mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}(\mathbf{E}(\mathbf{F_p}))$

In this section, we introduce the theoretical idea for the suggested development of the ECDH key exchange protocol using the matrix-group, in the following form:

$$\mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}(\mathbf{E}(\mathbf{F_p})) = \left\{\begin{bmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} & \cdots & \boldsymbol{P}_{1n} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} & \cdots & \boldsymbol{P}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} & \boldsymbol{P}_{m2} & \cdots & \boldsymbol{P}_{mn} \end{bmatrix} \middle| \boldsymbol{P_{ij}} \in \mathbf{E}(\mathbf{F_p}), \boldsymbol{i} = \mathbf{1}, \ldots, \boldsymbol{m}, \boldsymbol{j} = \mathbf{1}, \ldots, \boldsymbol{n} \right\}.$$

We can define the on $\mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}(\mathbf{E}(\mathbf{F_p}))$ internal law called addition "+" as follows, let $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} & \cdots & \boldsymbol{P}_{1n} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} & \cdots & \boldsymbol{P}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} & \boldsymbol{P}_{m2} & \cdots & \boldsymbol{P}_{mn} \end{bmatrix}$ and $\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} & \cdots & \boldsymbol{Q}_{1n} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} & \cdots & \boldsymbol{Q}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{Q}_{m1} & \boldsymbol{Q}_{m2} & \cdots & \boldsymbol{Q}_{mn} \end{bmatrix}$ be two elements in $\mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}(\mathbf{E}(\mathbf{F_p}))$, then $\boldsymbol{Z} = \boldsymbol{X} + \boldsymbol{Y} =$

$$\begin{bmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} & \cdots & \boldsymbol{P}_{1n} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} & \cdots & \boldsymbol{P}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} & \boldsymbol{P}_{m2} & \cdots & \boldsymbol{P}_{mn} \end{bmatrix} + \begin{bmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} & \cdots & \boldsymbol{Q}_{1n} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} & \cdots & \boldsymbol{Q}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{Q}_{m1} & \boldsymbol{Q}_{m2} & \cdots & \boldsymbol{Q}_{mn} \end{bmatrix} =$$

$$\begin{bmatrix} \boldsymbol{P}_{11} \oplus \boldsymbol{Q}_{11} & \boldsymbol{P}_{12} \oplus \boldsymbol{Q}_{12} & \cdots & \boldsymbol{P}_{1n} \oplus \boldsymbol{Q}_{1n} \\ \boldsymbol{P}_{21} \oplus \boldsymbol{Q}_{21} & \boldsymbol{P}_{22} \oplus \boldsymbol{Q}_{22} & \cdots & \boldsymbol{P}_{2n} \oplus \boldsymbol{Q}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} \oplus \boldsymbol{Q}_{m1} & \boldsymbol{P}_{m2} \oplus \boldsymbol{Q}_{m2} & \cdots & \boldsymbol{P}_{mn} \oplus \boldsymbol{Q}_{mn} \end{bmatrix}.$$

We can also define $SA$, which is the outcome of the matrix scalar multiplication that can be formed by multiplying each point of $\boldsymbol{X}$ by $S$ as follows:

$$S\boldsymbol{X} = (S\boldsymbol{P_{ij}}) = S\begin{bmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} & \cdots & \boldsymbol{P}_{1n} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} & \cdots & \boldsymbol{P}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} & \boldsymbol{P}_{m2} & \cdots & \boldsymbol{P}_{mn} \end{bmatrix} = \begin{bmatrix} S\boldsymbol{P}_{11} & S\boldsymbol{P}_{12} & \cdots & S\boldsymbol{P}_{1n} \\ S\boldsymbol{P}_{21} & S\boldsymbol{P}_{22} & \cdots & S\boldsymbol{P}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ S\boldsymbol{P}_{m1} & S\boldsymbol{P}_{m2} & \cdots & S\boldsymbol{P}_{mn} \end{bmatrix},$$

where $S$ be an integer number.

**Proposition 2.1:**

The set of all matrices over the group $\mathbf{E}(\mathbf{F_p})$ with addition "+" (*i.e.,* $(\boldsymbol{M}_{\boldsymbol{m}\times\boldsymbol{n}}(\boldsymbol{E}(\boldsymbol{F_p})), +)$) is a unitary abelian group with identity,

$$\boldsymbol{O} = \begin{bmatrix} \boldsymbol{O}_\infty & \boldsymbol{O}_\infty & \cdots & \boldsymbol{O}_\infty \\ \boldsymbol{O}_\infty & \boldsymbol{O}_\infty & \cdots & \boldsymbol{O}_\infty \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O}_\infty & \boldsymbol{O}_\infty & \cdots & \boldsymbol{O}_\infty \end{bmatrix}.$$

**Proof:**

One can easily show that the proposition is true by checking component by component. ■

We adopted the Hadamard product of matrices in [15] to extract the definition below, which is used later to develop ECDH key exchange protocol [24][25].

**Definition 2.2:**

Let $\boldsymbol{W}$ be a matrix whose entries are integer (*i.e.,* $\boldsymbol{k_{ij}} \in \mathbf{Z}$ for $\boldsymbol{i} = \mathbf{1}, \ldots, \boldsymbol{m}, \boldsymbol{j} = \mathbf{1}, \ldots, \boldsymbol{n}$.) and $\boldsymbol{X}$ as defined above. Then the new Hadamard product is defined as follows:

$$W \odot X = \begin{bmatrix} k_{11} & k_{12} \cdots & k_{1n} \\ k_{21} & k_{22} \cdots & k_{2n} \\ \vdots & \vdots & \vdots \\ k_{m1} & k_{m2} \cdots & k_{mn} \end{bmatrix} \odot \begin{bmatrix} P_{11} & P_{12} \cdots & P_{1n} \\ P_{21} & P_{22} \cdots & P_{2n} \\ \vdots & \vdots & \vdots \\ P_{m1} & P_{m2} \cdots & P_{mn} \end{bmatrix} = \begin{bmatrix} kP_{11} & kP_{12} \cdots & kP_{1n} \\ kP_{21} & kP_{22} \cdots & kP_{2n} \\ \vdots & \vdots & \vdots \\ kP_{m1} & kP_{m2} \cdots & kP_{mn} \end{bmatrix}.$$

## 3. The Development of ECDH Key Exchange Protocol

In this section, we will introduce developments of the ECDH key exchange protocol using the above concepts.

### 3.1 The First Proposed Development of ECDH Protocol

A cryptographic procedure known as Diffie-Hellman key exchange enables two parties to generate a shared private key over an insecure communication channel. It was developed by Diffie and Hellman in 1976 and is considered one of the oldest and most important public key cryptographic protocols. It provides a way to exchange a shared secret between two parties without transferring the private key itself, making it difficult for attackers to intercept and crack the communication. The security of the Diffie-Hellman protocol depends on the difficulty of computing discrete logarithms, which is currently thought to be a computationally infeasible problem [26].

We improve the protocol by mean of using the matrices concept combined with the elliptic curve points. Suppose two users Ban and Ali want to exchange keys via the ECDH protocol, the procedure goes like this:

**1. Initialization**

• Ban and Ali publicly select a base matrix $\boldsymbol{B}$ (all $\boldsymbol{P_{ij}}$ have prime order) over the group $\mathbf{E}(\mathbf{F_p})$ with $\boldsymbol{m}$ rows and $\boldsymbol{n}$ columns (*i.e.*, $\boldsymbol{B} \in \mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}\,(\mathbf{E}(\mathbf{F_p}))$).

**2. Key generation**

• Ban selects a private positive integer $\boldsymbol{e}$ and calculates $\boldsymbol{eB} \in \mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}\,(\mathbf{E}(\mathbf{F_p}))$.

• Ali selects a private positive integer $\boldsymbol{d}$ and calculates $\boldsymbol{dB} \in \mathbf{M}_{\boldsymbol{m}\times\boldsymbol{n}}\,(\mathbf{E}(\mathbf{F_p}))$.

• Keep $\boldsymbol{e}$ and $\boldsymbol{d}$ private and make $\boldsymbol{eB}$ and $\boldsymbol{dB}$ public.

**3. Calculation of the private key matrix $\boldsymbol{deB}$**

• Ban calculates the private shared key $\boldsymbol{edB} = \boldsymbol{e}(\boldsymbol{dB})$.

• Ali calculates the private shared key $\boldsymbol{deB} = \boldsymbol{e}(\boldsymbol{dB})$.

The only information intercepted by the eavesdropper Ahmed is the matrix over the group $\mathbf{E}(\mathbf{F_p})$ with $m$ rows and $n$ columns, as well as the matrices $\boldsymbol{B}$, $\boldsymbol{eB}$, and $\boldsymbol{dB}$. As a result, Ahmed's issue is that he must use only $\boldsymbol{B}$, $\boldsymbol{eB}$, and $\boldsymbol{dB}$ to calculate $\boldsymbol{edB}$. Ahmed could use $\boldsymbol{B}$ and $\boldsymbol{eB}$ to discover Ban's private key $\boldsymbol{e}$ if he could solve the discrete logarithm over $\mathbf{E}(\mathbf{F_p})$. Ahmed could then compute $\boldsymbol{e}(\boldsymbol{dB})$ to obtain $\boldsymbol{edB}$. However, in order to solve this issue, you must be familiar with solving discrete logarithms many times based on the size of the base matrix $\boldsymbol{B}$, which was thought to be intractable to solve over $\mathbf{E}(\mathbf{F_p})$.

**Example 3.1:**

Consider an elliptic curve $\boldsymbol{E}$ define over $\mathbf{F_{41}}$ with parameters $\boldsymbol{a} = 1, \boldsymbol{b} = 5$ where $4\boldsymbol{a}^{\mathbf{3}} + 27\boldsymbol{b}^{\mathbf{2}} \bmod \boldsymbol{p} = 23 \neq 0$ and $\boldsymbol{B} = \begin{bmatrix} (4{,}14) & (12{,}8) \\ (33{,}31) & (9{,}28) \end{bmatrix}$. Suppose Ban and Ali agree upon an $\mathbf{E}(\mathbf{F_{41}})$ and a base matrix $\boldsymbol{B}$. How can they exchange the key?

**Solution:**

- Ban selects a private positive integer $\boldsymbol{e} = 11$.

$$\boldsymbol{eB} = 11\begin{bmatrix} (4{,}14) & (12{,}8) \\ (33{,}31) & (9{,}28) \end{bmatrix} = \begin{bmatrix} 11(4{,}14) & 11(12{,}8) \\ 11(33{,}31) & 11(9{,}28) \end{bmatrix} = \begin{bmatrix} (9{,}28) & (39{,}35) \\ (14{,}37) & (12{,}8) \end{bmatrix},$$

and send $\begin{bmatrix} (9{,}28) & (39{,}35) \\ (14{,}37) & (12{,}8) \end{bmatrix}$ to Ali.

- Ali selects a private positive integer $\boldsymbol{d} = 35$.

$$\boldsymbol{dB} = 35\begin{bmatrix}(4,14) & (12,8)\\(33,31) & (9,28)\end{bmatrix} = \begin{bmatrix}35(4,14) & 35(12,8)\\35(33,31) & 35(9,28)\end{bmatrix}$$

$= \begin{bmatrix}(26,10) & (32,28)\\(10,21) & (28,38)\end{bmatrix}$, and send $\begin{bmatrix}(26,10) & (32,28)\\(10,21) & (28,38)\end{bmatrix}$ to Ban.

- Ban calculates the private key $\boldsymbol{e(dB)}$:

$$\boldsymbol{e(dB)} = 11\begin{bmatrix}(26,10) & (32,28)\\(10,21) & (28,38)\end{bmatrix} = \begin{bmatrix}11(26,10) & 11(32,28)\\11(10,21) & 11(28,38)\end{bmatrix}$$

$$= \begin{bmatrix}(28,38) & (8,19)\\(0,13) & (32,28)\end{bmatrix}.$$

- Ali calculates the private key $\boldsymbol{d(eB)}$:

$$\boldsymbol{d(eB)} = 35\begin{bmatrix}(9,28) & (39,35)\\(14,37) & (12,8)\end{bmatrix} = \begin{bmatrix}35(9,28) & 35(39,35)\\35(14,37) & 35(12,8)\end{bmatrix}$$

$$= \begin{bmatrix}(28,38) & (8,19)\\(0,13) & (32,28)\end{bmatrix}.$$

Now, Ban and Ali have the same matrix key $\boldsymbol{edB} = \begin{bmatrix}(28,38) & (8,19)\\(0,13) & (32,28)\end{bmatrix}$.

### 3.2 The Second Proposed Development of ECDH Protocol

- Ban and Ali agree upon an $\mathbf{E(F_p)}$ and a base matrix $\boldsymbol{B} \in \mathbf{M}_{\boldsymbol{m\times n}}\,(\mathbf{E(F_p)})$ (all $\boldsymbol{P_{ij}}$ have prime order).
- (Ali) selects a random and secret matrix $\boldsymbol{E(D)}$ whose coefficients are integers with same size of the matrix $\boldsymbol{B}$, and computes $\boldsymbol{E \odot B}$ $(\boldsymbol{D \odot B})$ and sends it to Ban (Ali). Then $\boldsymbol{E \odot B}$ and $\boldsymbol{D \odot B}$ are public and $\boldsymbol{E}$ and $\boldsymbol{D}$ are a private.
- Ban (Ali) calculates the private shared key $\boldsymbol{E \odot D \odot B}$ $\boldsymbol{(D \odot E \odot B)}$.

There is no fast method to calculate $\boldsymbol{E{\odot}D{\odot}B}$ if only $\boldsymbol{B}$, $\boldsymbol{E{\odot}B}$, and $\boldsymbol{D{\odot}B}$ are known. Following these designs, Ban and Ali have the same matrix (only Ban and Ali know it).

**Example 3.2**:

Suppose that Ban and Ali agree upon an $\mathbf{E}: \boldsymbol{y^2 = x^3 + 553}$ over $\boldsymbol{p} = 3023$ and a base matrix $\boldsymbol{B} = \begin{bmatrix}(1977,876) & (2025,384)\\(2028,59) & (1977,2142)\end{bmatrix}$, How can they exchange the key?

**Solution:**

- Ban selects a matrix $\boldsymbol{E} = \begin{bmatrix}1 & 4\\3 & 1\end{bmatrix}$, and calculates $\boldsymbol{E \odot B}$ and sends it to Ali $\boldsymbol{E \odot B} =$

$$\begin{bmatrix}1 & 4\\3 & 1\end{bmatrix} \odot \begin{bmatrix}(1977,876) & (2025,384)\\(2028,59) & (1977,2142)\end{bmatrix}$$

$$= \begin{bmatrix}1(1977,876) & 4(2025,384)\\3(2028,59) & 1(1977,2142)\end{bmatrix} = \begin{bmatrix}(1977,876) & (1769,2636)\\(154,254) & (1977,2142)\end{bmatrix}.$$

- Ali selects a matrix $\boldsymbol{D} = \begin{bmatrix}2 & 5\\3 & 4\end{bmatrix}$, and calculates $\boldsymbol{D \odot B}$ and sends it to Ban

$\boldsymbol{D \odot B}$

$$= \begin{bmatrix}2 & 5\\3 & 4\end{bmatrix} \odot \begin{bmatrix}(1977,876) & (2025,384)\\(2028,59) & (1977,2142)\end{bmatrix}$$

$$= \begin{bmatrix}2(1977,876) & 5(2025,384)\\3(2028,59) & 4(1977,2142)\end{bmatrix} = \begin{bmatrix}(1159,507) & (1474,2846)\\(154,254) & (1368,1640)\end{bmatrix}.$$

- Ban (Ali) calculates the private shared key $\boldsymbol{E \odot D \odot B}$ $(\boldsymbol{D \odot E \odot B})$

$\boldsymbol{E \odot D \odot B}$

$$= \begin{bmatrix}1 & 4\\3 & 1\end{bmatrix} \odot \begin{bmatrix}(1159,507) & (1474,2846)\\(154,254) & (1368,1640)\end{bmatrix}$$

$$\begin{bmatrix}1(1159,507) & 4(1474,2846)\\3(154,254) & 1(1368,1640)\end{bmatrix}$$

$$= \begin{bmatrix}(1159,507) & (134,2857)\\(1942,1305) & (1368,1640)\end{bmatrix}$$

$\boldsymbol{D \odot E \odot B}$

$$= \begin{bmatrix}2 & 5\\3 & 4\end{bmatrix} \odot \begin{bmatrix}(1977,876) & (1769,2636)\\(154,254) & (1977,2142)\end{bmatrix}$$

$$= \begin{bmatrix}2(1977,876) & 5(1769,2636)\\3(154,254) & 4(1977,2142)\end{bmatrix}$$

$$= \begin{bmatrix}(1159,507) & (134,2857)\\(1942,1305) & (1368,1640)\end{bmatrix}$$

Figure 1 depicts the input file size vs encryption execution time, in total in our experiment we have used three different encryption techniques. Our proposed approach ECDH takes less execution time in comparison to other two technique RSA and DSA.

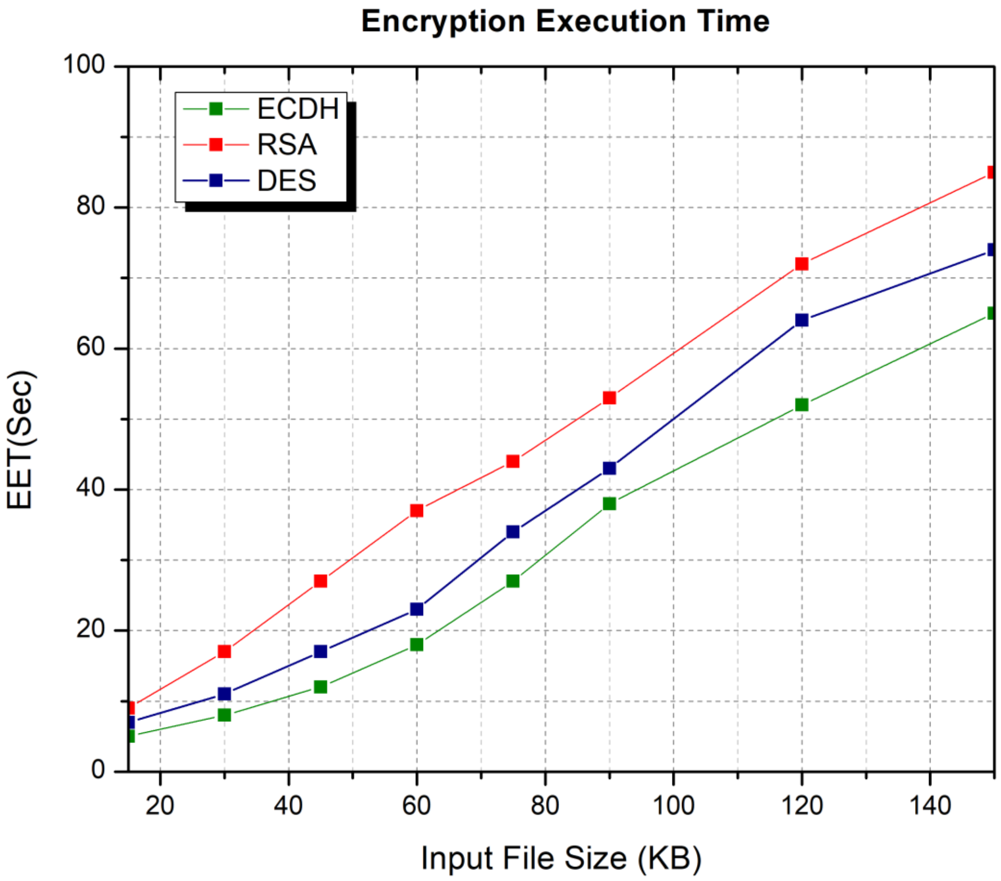

**Figure 1:** Input file size VS encryption execution time

## 4. Security of Proposed Protocol

ECDLP is the foundation for the security of the proposed ECDH method. Attackers must answer either the algorithmic Diffie-Hellman problem or the decisional Diffie-Hellman problem to break the protocol.

Finding $\boldsymbol{E \odot D \odot B}$ or $\boldsymbol{edB}$ from $\boldsymbol{E \odot B}$ or $\boldsymbol{eB}$ and $\boldsymbol{D \odot B}$ or $\boldsymbol{dB}$ in the context of the computational Diffie-Hellman issue. The most potent technique for attacking discrete logarithms in finite fields is the Pollard's-rho approach, which is dependent on the size of a subgroup produced by the generator $\boldsymbol{P}$. The complexity of Pollard's-rho technique equals $\boldsymbol{O(\sqrt{r})}$ if $\boldsymbol{ord\ (P) = r}$, where $\boldsymbol{r}$ is the order of $\boldsymbol{P}$.

Thus, selecting the size of base matrix $\boldsymbol{B}$ is equal to $m \times n$ will make the attackers need to solve the ECDLP $m \times n$ times instead of once, and hence the security of the new approach for ECDH key exchange will increase based on the size of the base matrix. The complexity of Pollard's-rho technique will also increase based on the base matrix.

## 5. Conclusion

The more complex systems are those that can only be attacked by exhaustive search, so the square root attack is the only viable strike against them. The DLP lacks this toughness, but the ECDLP does. However, with the DLP, it is simple to increase the key sizes and thus enhance the hardness and acquire the desired system security. The ECDLP requires a significant quantity of time and computational resources to increase the key sizes, raising the time required to crack them.

The approach we suggest has a level of difficulty comparable to the ECDLP. However, with the method revealed, we can expand the key values as desired by adjusting the size of the base matrix whose components are points on an elliptic curve, thereby increasing system security to the desired level. This new approach to security is built on exponentiating the block matrices of the discrete logarithm on an elliptic curve.

This new approach to security is built on exponentiating the discrete logarithm over block matrices on an elliptic curve. We applied the proposed approach to the Diffie-Hellman key exchange protocol, and in the future, we will attempt to study its use on some cryptosystems.

## 6. References

[1] N. Ferguson, B. Schneier and T. Konho. “Cryptography Engineering: Design Principles and Practical Applications”, John Wiley & Sons, 2011.

[2] W. Stallings. “Cryptography and Network Security: Principles and Practice”, 7th ed.; Prentice Hall, Upper Saddle River, New Jersey, USA, 2017.

[3] W. Diffie and M. E. Hellman. “New directions in cryptography”, IEEE Transactions on Information Theory, Vol. 22, No. 6, pp. 644-654, 1976.

[4] D. Wong, “Real-world cryptography”, Simon and Schuster, 2021.

[5] N. Koblitz. “Elliptic Curve Cryptosystems”. Mathematics of Computation, Vol. 48, No. 77, pp. 203-209, 1987.

[6] V. Miller. “Uses of Elliptic Curves in cryptography”, In Advances in Cryptology-CRYPTO 85; Springer: Berlin/Heidelberg, Germany, pp. 417-426, 1986.

[7] D. Johnson, A. Menezes, “The elliptic curve digital signature algorithm (ECDSA)”, International journal of information security, Vol. 1, No. 1, pp. 36-63, 2001.

[8] Y.-J. Huang, C. Petit, N. Shinohara, and T. Takagi, “Improvement of FPPR method to solve ECDLP”, Pacific Journal of Mathematics for Industry, Vol. 7, No. 1, pp. 1-9, 2015.

[9] J. J. Climent, F. Ferr´andez, J. F. Vicent, and A. Zamora. “A nonlinear elliptic curve cryptosystem based on matrices”, Applied Mathematics and Computation, Vol. 174, No. 1, pp. 150-164, 2006.

[10] F. Amounas and E. H. El Kinani. “Fast mapping method based on matrix approach for elliptic curve cryptography”, International Journal of Information & Network Security (IJINS), Vol. 1, No. 2, pp. 54-59, 2012.

[11] A. Chillali, C. Zakariae, and A. Mouhib. “Prof The" Elliptic" matrices and a new kind of cryptography”, Boletim da Sociedade Paranaense de Matemática Vol. 41, pp. 1-12, 2023.

[12] A. A. Neamah, “New Collisions to Improve Pollard's Rho Method of Solving the Discrete Logarithm Problem on Elliptic Curves”, Journal of Computer Science, Vol. 11, No. 9, pp. 971-975, 2015.

[13] Y. Jiang, Y. Shen, and Q. Zhu. “A lightweight key agreement protocol based on Chinese remainder theorem and ECDH for smart homes”, Sensors, Vol. 20, No. 5, pp. 1-13, 2020.

[14] C. A. Lara-Nino, A. Diaz-Perez, and M. Morales-Sandoval, “Elliptic curve lightweight cryptography: A survey,” IEEE Access, Vol. 6, pp. 72514-72550, 2018.

[15] R. A. Horn and C. R. Johnson, “Topics in Matrix Analysis”, Cambridge University Press, 1994.

[16] Pentyala, S., Liu, M., & Dreyer, M. (2019). Multi-task networks with universe, group, and task feature learning. arXiv preprint arXiv:1907.01791.

[17] Srivastava, Swapnita, and P. K. Singh. "Proof of Optimality based on Greedy Algorithm for Offline Cache Replacement Algorithm." International Journal of Next-Generation Computing 13.3 (2022).

[18] Smiti, Puja, Swapnita Srivastava, and Nitin Rakesh. "Video and audio streaming issues in multimedia application." 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE, 2018.

[19] Srivastava, Swapnita, and P. K. Singh. "HCIP: Hybrid Short Long History Table-based Cache Instruction Prefetcher." International Journal of Next-Generation Computing 13.3 (2022).

[20] Srivastava, Swapnita, and Shilpi Sharma. "Analysis of cyber related issues by implementing data mining Algorithm." 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE, 2019.

[21] P Mall and P. Singh, “Credence-Net: a semi-supervised deep learning approach for medical images,” Int. J. Nanotechnol., vol. 20, 2022.

[22] Narayan, Vipul, et al. "Deep Learning Approaches for Human Gait Recognition: A Review." 2023 International Conference on Artificial Intelligence and Smart Communication (AISC). IEEE, 2023.

[23] Narayan, Vipul, et al. "FuzzyNet: Medical Image Classification based on GLCM Texture Feature." 2023 International Conference on Artificial Intelligence and Smart Communication (AISC). IEEE, 2023.

[24] "Keyboard invariant biometric authentication." 2018 4th International Conference on Computational Intelligence & Communication Technology (CICT). IEEE, 2018.

[25] Mall, Pawan Kumar, et al. "Early Warning Signs Of Parkinson’s Disease Prediction Using Machine Learning Technique." Journal of Pharmaceutical Negative Results (2023): 2607-2615.

[26] Choudhary, Shubham, et al. "Fuzzy approach-based stable energy-efficient AODV routing protocol in mobile ad hoc networks." Software Defined Networking for Ad Hoc Networks. Cham: Springer International Publishing, 2022. 125-139.