Mathematics Interdisciplinary Research **5** (2020) 193 − 206

# An ECDLP-Based Verifiable Multi-Secret Sharing Scheme

*Khadijeh Eslami and Mojtaba Bahramian* *

**Abstract**

Secret sharing is an important issue in cryptography which has many applications. In a secret sharing scheme, a secret is shared by a dealer among several participants in such a way that any authorized subset of participants can recover the secret by pooling their shares. Recently, several schemes based on elliptic curves and bilinear maps have been presented. Some of these schemes need a secure channel, there are restrictions on the number of secrets, or the participants or the dealer are unable to verify the validity of the shares. In this paper, we present a new verifiable $(t, n)$-threshold multi-secret sharing scheme based on elliptic curves and pairings that does not have any of the above restrictions. The hardness of a discrete logarithm problem on elliptic curves guarantees the security of the proposed scheme.

Keywords: secret sharing scheme, elliptic curves, pairings, discrete logarithm problem.

2010 Mathematics Subject Classification: 94A62, 14H52.

**How to cite this article**
Kh. Eslami and M. Bahramian, An ECDLP-Based Verifiable Multi-Secret Sharing Scheme, *Math. Interdisc. Res.* **5** (2020) 193 − 206.

## 1. Introduction

Secret sharing scheme is an important initiative in cryptography, which is a method of sharing a secret among a set of participants by a dealer in such a way that only predefined subsets of participants can reconstruct the secret by collecting their shares. A scheme in which more than one secret is shared is called a multi-secret

*Corresponding author (E-mail: bahramianh@kashanu.ac.ir)
Academic Editor: Ali Reza Ashrafi
Received 1 January 2020, Accepted 5 July 2020
DOI: 10.22052/mir.2020.217418.1193

sharing scheme. The person responsible for the recovery of the secret, who can be one of the participants, is called a combiner.

The scheme in which the dealer distributes a secret between $n$ participants such that any group of at least $t$ participants can retrieve the secret, but no group of less than $t$ participants can get anything about it, is called a $(t, n)$-threshold secret sharing scheme.

The first secret sharing schemes were independently introduced by Shamir [11] and Blakley [2] in 1979. Shamir's scheme is based on the Lagrange interpolating polynomial, and Blakley's scheme is based on linear projective geometry.

In Shamir's scheme, in order to distribute a secret $s$ between participants $U_1, \cdots, U_n$, the dealer chooses a large prime number $p$ and a polynomial $f(x) \pmod p$ of degree $t-1$ such that $f(0) = s$. The dealer sends the value $s_i = f(i)$ to the participant $U_i$ in a secure channel. If any $t$ participants gather, they can compute the polynomial $f(x)$ and $s = f(0)$, using Lagrange's interpolation technique, while any number of less than $t$ participants cannot recover the secret at all.

Blakley's scheme is also a $(t, n)$-threshold secret sharing scheme in which the dealer chooses $n$ nonparallel $(t-1)$-dimensional hyperplanes $H_1, \ldots, H_n$ that they intersect at exactly one point. The secret $s$ can be any single coordinate of the point of intersection. In order to distribute the secret $s$, the dealer sends the hyperplane $H_i$ to the participant $U_i$ in a secure channel. Clearly, any group of at least $t$ participants can obtain the intersection of the hyperplanes by pooling their shares and they can reconstruct the secret, whereas any group of less than $t$ participants gain no information about it.

In secret sharing scheme, it is assumed that the dealer and the participants are honest, however, a dishonest dealer may send a fake shadow to the participants or a malicious participant may send a fake share during the reconstruction phase. Therefore, it is advantageous that each participant can verify the validity of shares submitted by the dealer during the share allocation phase and shares submitted by other participants during the reconstruction phase. For this purpose, verifiable secret sharing (VSS) scheme was suggested by Chore et. al. [4] in 1985 and verifiable multi-secret sharing (VMSS) scheme was proposed by Harn [5] in 1995. Also in 2004 an information theoretic secure VSS was proposed by Tang et. al. [13].

In recent years, several schemes based on elliptic curves and pairing maps have been presented. However, there are some drawbacks as follows:

- Some of these schemes need a secure channel.
- There are restrictions on the number of secrets in some of these schemes.
- The participants or the dealer are unable to verify the validity of the shares submitted in the share distribution phase or during the reconstruction phase.

In 2007, a verifiable multi-secret sharing scheme based on elliptic curves was proposed by Shi et. al. [12]. The need for a secure channel and restrictions on the number of shared secrets are some drawbacks of this scheme. Chen et. al. [3] presented a secret sharing scheme based on bilinear pairings in 2008. In this scheme only one secret can be distributed among participants. Wang improved Chen's scheme and introduced a $(t, n)$-threshold multi secret sharing scheme [14]. Although a few secrets are shared in this scheme, only $t$ secrets can be shared during a secret sharing process. In 2008, a multi-point sharing scheme using self-pairing on elliptic curves was proposed by Liu et. al. [8]. Liu's scheme needs a secure channel, there is restriction on the number of secrets and the correctness of secret shadows cannot be verified. In 2016, Binu presented a verifiable multi-secret sharing scheme to address some of these problems, while in both schemes [8, 1] communication between the dealer and participants is done over secure channel.

In this paper we propose a new $(t, n)$-threshold multi secret sharing scheme based on elliptic curves and bilinear maps. Our scheme is verifiable, doesn't require a secure channel and there is no restriction on the number of secrets.

The rest of the paper is organized in the following manner: Section 2, presents a definition of elliptic curves, discrete logarithm problem and bilinear pairings. In Section 3, we explain our scheme which is based on the elliptic curves and bilinear pairings. The security analysis and conclusions of proposed scheme is presented in Sections 4 and 5. Finally we give an example for our scheme in Section 6.

# 2. Preliminaries

In this section, we will briefly introduce elliptic curves, discrete logarithm problem and bilinear pairings.

## 2.1 Elliptic Curves

Elliptic curves are used in many mathematical fields such as cryptography and solving Diophantine equations. The elliptic curve cryptography (ECC) was suggested by Neal Koblitz [6] and Victor S. Miller [9] in 1985. The security of the schemes that use the group of points on elliptic curves, is based on the hardness of discrete logarithm problem.

Let $K$ be a field. An elliptic curve $E$ defined over $K$ is a smooth plane cubic curve given by a long Weierstrass equation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

where $a_1, a_2, a_3, a_4, a_6 \in K$. The homogenization of the curve $E$ is given by

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3.$$

The only point at infinity on this curve is $[0 : 1 : 0]$, we denote this point by $\mathcal{O}$ from now on. This point is the neutral element in the group structure on $E$.

If $char(K) \neq 2, 3$ then, by a suitable change of variables, we have the short Weierstrass equation

$$y^2 = x^3 + Ax + B, \tag{1}$$

where $A, B \in K$. It is well-known that if $E$ is a curve given by the Weierstrass equation (1), then $E$ is an elliptic curve if and only if its discriminant $\Delta = 4A^3 + 27B^2$ is nonzero.

We will now proceed to define the group structure on $E$. Let $E$ be an elliptic curve over $K$, defined by (1). Let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ be two points on $E$. The addition $R = P + Q = (x_3, y_3)$ is defined as the following statements:

- If $x_1 = x_2$ and $y_1 \neq y_2$, then $R = \mathcal{O}$.
- If $x_1 \neq x_2$, then $x_3 = \lambda^2 - x_1 - x_2$ and $y_3 = \lambda(x_1 - x_3) - y_1$, where $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$.
- If $P = Q$ and $y_1 = 0$, then $R = \mathcal{O}$.
- If $P = Q$, and $y_1 \neq 0$, then $x_3 = \lambda^2 - 2x_1$ and $y_3 = \lambda(x_1 - x_3) - y_1$, where $\lambda = \frac{3x_1^2 + A}{2y_1}$.

The points on $E$ form an additive abelian group with $\mathcal{O}$ as the identity element.

## 2.2 Discrete Logarithm Problem

Let $G$ be a group, $g \in G$ and $h \in \langle g \rangle$. The discrete logarithm problem (DLP) on $G$ is the problem of finding the integer $k$ such that $h = g^k$. The integer $k$ is called the discrete logarithm of $h$ to the base $g$. In the case that $E$ is the group of points on an elliptic curve, the discrete logarithm problem on $E$ is called the elliptic curve discrete logarithm problem (ECDLP). The hardness of discrete logarithm problem on some groups is the base of many cryptosystems such as Diffie-Hellman key exchange protocol, ElGamal public-key encryption scheme and the digital signature algorithm (DSA).

## 2.3 Bilinear Pairings

Let $\Gamma$ be an additive finite group in which the DLP is hard. A (self) pairing on $\Gamma$ is a map $e : \Gamma \times \Gamma \to \Gamma$ with the following properties:

1. $e$ is bilinear: $e(aP, bQ) = ab.e(P, Q)$ for all $a, b \in \mathbb{Z}$, and $P, Q \in \Gamma$.
2. $e$ is non-degenerate in each variable: If $e(P, Q) = 0$ for all $Q \in \Gamma$, then $P = 0$. Similarly, if $e(P, Q) = 0$ for all $P \in \Gamma$, then $Q = 0$.
3. $e$ is computable: There exists an efficiently computable (a polynomial time) algorithm for computing $e(P, Q) \in \Gamma$, for all $P, Q \in \Gamma$.

Here, we give a description of self-pairing which has been proposed in [7]. Let $q$ be some power of a prime number, $E$ be an elliptic curve defined over $\mathbb{F}_q$ and let $l$ be an integer coprime to $q$. Therefore, $E[l]$, the set of $l$-torsion points of $E(\overline{\mathbb{F}_q})$, is isomorphic to $\mathbb{Z}_l \oplus \mathbb{Z}_l$. Let $\{P, Q\}$ be a set of generators of $E[l]$. For any two points $G$ and $H$ in $E[l]$, there are integers $a_1, a_2, b_1, b_2 \in [0, l-1]$ such that $G = a_1P + b_1Q$ and $H = a_2P + b_2Q$. By considering the fixed integers $\alpha, \beta \in [0, l-1]$, we can define the following self-pairing map:

$$e_{\alpha,\beta} : E[l] \times E[l] \to E[l]$$
$$e_{\alpha,\beta}(G, H) = (a_1b_2 - a_2b_1)(\alpha P + \beta Q).$$

**Theorem 2.1.** [7, Prop.3.1] *The self-pairing $e_{\alpha,\beta}$ has the following properties:*

1. *Identity: For all $G \in E[l]$, $e_{\alpha,\beta}(G, G) = \mathcal{O}$.*
2. *Bilinearity: For all $G, H, R \in E[l]$, $e_{\alpha,\beta}(G+H, R) = e_{\alpha,\beta}(G, R)+e_{\alpha,\beta}(H, R)$ and $e_{\alpha,\beta}(G, H + R) = e_{\alpha,\beta}(G, H) + e_{\alpha,\beta}(G, R)$.*
3. *Anti-symmetry: For all $G, H \in E[l]$, $e_{\alpha,\beta}(G, H) = -e_{\alpha,\beta}(H, G)$.*
4. *Non-degeneracy: For all $G \in E[l]$, $e_{\alpha,\beta}(G, \mathcal{O}) = \mathcal{O}$. Moreover, if $e_{\alpha,\beta}(G, H) = \mathcal{O}$ for all $H \in E[l]$, then $G = \mathcal{O}$.*

# 3. Proposed Scheme

In this section, we describe a new verifiable $(t, n)-$threshold secret sharing scheme using elliptic curves and bilinear maps. The proposed scheme is divided into three stages: Initialization phase, Points sharing phase, Reconstruction, and the Verification phase. During the discussion, we use the following notations:

- $D$: the dealer, who wants to share the secret among the participants,
- $\{U_1, U_2, \ldots, U_n\}$: the set of all the participants,
- $\{K_1, K_2, \ldots, K_m\}$: the set of all secrets to be shared,
- $e$: a bilinear self-pairing.

## 3.1. Initialization Phase

In the initialization phase, dealer $D$ publishes some public information which can be accessed by every participant.

1. The Dealer $D$ chooses an elliptic curve $E$ over $\mathbb{F}_q$, $q = p^r$, where $p$ is a large prime such that ECDLP in $E(\mathbb{F}_q)$ is hard. The dealer then chooses a large prime $l$ coprime to $p$, and he/she selects $E[l] \subseteq E(\mathbb{F}_{q^k})$ for some positive integer $k$. Also the dealer chooses a hash function $h : E \to \mathbb{Z}_l^*$.

2. $D$ chooses a point $G \in E[l]$ and he/she selects an integer $d \in \mathbb{Z}_l^*$ as a private key. Also, $D$ chooses a pairing map $e : E[l] \times E[l] \to E[l]$.

3. $D$ publishes $\{E, q, l, G, dG, h, e\}$.

4. Each participant $U_i$ chooses $s_i \in E[l]$, computes $S_i = e(s_i, dG)$, and publishes it.

5. The dealer, using his/her own private key $d$, computes $d^{-1}e(s_i, dG)$, and obtains $e(s_i, G)$.

6. The dealer accepts $s_i$, when he/she ensures that $e(s_i, G) \neq e(s_j, G)$ for all $i \neq j$. (If $e(s_i, G) = e(s_j, G)$ for some $i \neq j$, then D should demand these participants to choose another secret shadow until $e(s_i, G)$'s are different for $i = 1, 2, \ldots, n$.)

## 3.2. Points Sharing Phase

In this phase, the dealer uses the following steps to distribute the shadows among the participants such that any group of at least $t$ participants can easily reconstruct the shared points, while any less than $t$ participants cannot reveal it.

Let $K_1, \ldots, K_m \in \mathbb{Z}_l$ be all the secrets to be shared. Dealer $D$ considers the following cases according to the values of $m$ and $n$.

**(i)** $m \leq n$

In this case, the dealer secretly and randomly chooses $n-m$ values $K_{m+1}, \ldots, K_n$ in $\mathbb{Z}_l$ as sharing secrets. Then $D$ publishes

$$R_i = K_i + \sum_{\substack{j=1 \\ j \neq i}}^{n} h(e(s_j, G)), \quad i = 1, \ldots, n,$$

and he/she performs the following steps:

1. The dealer $D$ considers the matrix

$$A = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & (n-t) & (n-t)^2 & \ldots & (n-t)^{n-1} \end{bmatrix}.$$

2. $D$ computes $e(s_i, G)$ for $i = 1, 2, \ldots, n$, constructs an $n$-column vector

$$(e(s_1, G)), \ldots, h(e(s_n, G))]^T,$$

where $T$ denotes the transpose of a matrix.

3. The dealer computes

$$A \times X = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & 2^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & (n-t) & \dots & (n-t)^{n-1} \end{bmatrix} \begin{bmatrix} h(e(s_1,G)) \\ h(e(s_2,G)) \\ \vdots \\ h(e(s_n,G)) \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_{n-t} \end{bmatrix}, \tag{2}$$

and then $D$ publishes $[I_1, I_2, \dots, I_{n-t}]^T$.

**(ii)** $m > n$

In this case, the dealer $D$ selects random points $s_{n+1}, \dots, s_m$ in $E[l]$, then $D$ publishes

$$R_i = K_i + \sum_{\substack{j=1 \\ j \neq i}}^{m} h(e(s_j, G)), \quad i = 1, \dots, m,$$

then $D$ performs the following steps:

1. The dealer $D$ considers the matrix

$$A = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 2 & 2^2 & \dots & 2^{m-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & (m-t) & (m-t)^2 & \dots & (m-t)^{m-1} \end{bmatrix}.$$

2. The dealer computes $e(s_i, G)$ for $i = 1, \dots, n$, and he/she constructs the $m$-column vector $[h(e(s_1,G)), \dots, h(e(s_m,G))]^T$.

3. $D$ computes

$$A \times X = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & 2^{m-1} \\ \vdots & \vdots & & \vdots \\ 1 & (m-t) & \dots & (m-t)^{m-1} \end{bmatrix} \begin{bmatrix} h(e(s_1,G)) \\ h(e(s_2,G)) \\ \vdots \\ h(e(s_m,G)) \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_{m-t} \end{bmatrix} \tag{3}$$

and then he/she publishes $[I_1, I_2, \dots, I_{m-t}]^T$.

## 3.3. Secrets Reconstruction and Verification Phase

In the case that $m \leq n$, the equation (2) is a system of $(n-t)$ linear equations in $n$ unknowns. Suppose that $t$ distinct participants want to reconstruct all the secrets. Without loss of generality, let participants $U_i$ for $i = 1, 2, \dots, t$, provide their shares. When the combiner receives $s_i$ for $i = 1, 2, \dots, t$, he/she first computes $e(s_i, dG)$. If $e(s_i, dG) = S_i$, then verification of the shares is confirmed. Next, the combiner computes $h(e(s_i, G))$ for $i = 1, 2, \ldots, t$. Therefore, $t$ unknowns of the equation (2) are determined and the other $(n - t)$ variables can be obtained by solving the system of equations in (2). Now, $\sum_{j=1, j\neq i}^{n} h(e(s_i, G))$ will be computed by combiner. Finally, all the secrets can be reconstructed by $R_i$.

In the case that $m > n$, as in the previous state, the shares are confirmed and the secrets are obtained.

## 4. Security Analysis

Here, we present security analysis of the proposed scheme by proving the following theorems.

**Theorem 4.1.** *Any $t-$out$-$of$-n$ participants can reconstruct all secrets and any group of fewer than t participants cannot compute any secrets.*

*Proof.* We consider $m \leq n$. Without loss of generality, suppose that the combiner receives $s_i$ for $i = 1, 2, \ldots, t$. Then, the equation (2) is reduced to a system of $(n - t)$ equations and $(n - t)$ unknowns and it could be represented as

$$A' \times X' = \begin{bmatrix} 1 & \cdots & 1 \\ 2^t & \cdots & 2^{n-1} \\ \vdots & & \vdots \\ (n-t)^t & \cdots & (n-t)^{(n-1)} \end{bmatrix} \begin{bmatrix} h(e(s_{t+1}, G)) \\ \vdots \\ h(e(s_n, G)) \end{bmatrix} = \begin{bmatrix} I'_1 \\ \vdots \\ I'_{n-t} \end{bmatrix}.$$

The square matrix $A'$ is a Vandermonde matrix on distinct elements. Therefore, $det(A') \neq 0$ and $(A')^{-1}$ can be computed to obtain $h(e(s_{t+1}, G)), \ldots, h(e(s_n, G))$. If fewer than $t$ participants pool their secrets, the number of unknowns in (3) is more than the number of equations. Therefore, the secrets cannot be computed. The case $m > n$ is similar. □

**Theorem 4.2.** *In the proposed scheme, we could distinguish cheater from honest participant.*

*Proof.* According the initialization phase, point $S_i$ can be calculated using the private key $s_i$. When the combiner receives $s_i$, he/she computes $e(s_i, dG)$. If $e(s_i, dG) = S_i$, the verification passes and the combiner accepts the $s_i$. Because if a cheater sends a fake $s_i$ to the combiner, then $e(s_i, dG) \neq S_i$, therefore $s_i$ will be rejected and the cheater will be distinguished. □

**Theorem 4.3.** *Adversary cannot obtain the dealer's secret information from public information.*

*Proof.* If an adversary wants to compute $d$ from $dG$, he/she must solve a discrete logarithm problem in $E[l]$, but there is no known efficient algorithm to solve it in polynomial time. Therefore, the dealer's secret information cannot be obtained from public information. □

**Theorem 4.4.** *There is no need to a secure channel for the proposed scheme.*

*Proof.* According to properties of bilinear maps, $e(s_i, dG) = de(s_i, G)$. Therefore, if an attacker wants to compute $e(s_i, G)$ from $e(s_i, dG)$, he/she needs to obtain $d$ from $G$ and $dG$, which is a discrete logarithm problem in $E[l]$ and it is hard according to our assumption. □

**Theorem 4.5.** *The dealer cannot become a cheater.*

*Proof.* Since each participant chooses his/her share, the dealer has no role in selection of participant's secret share. Therefore, the dealer cannot become a cheater. □

## 5. Comparison

In this section, we compare the efficiency of our proposed scheme with schemes [1, 10, 12, 14] based on elliptic curves and bilinear maps in terms of performance and security. In this paper, a $(t, n)$-threshold verifiable multi-secret sharing scheme based on elliptic curve discrete logarithm problem is proposed. In distribution phase we presented two different algorithms for the cases $m \leq n$ and $m > n$. Considering $r = \max\{n, m\}$, we need to solve $r - t$ simultaneous equations to reconstruct $m$ secrets. It is found that in [12, 14], the number of secrets that can be shared are atmost the threshold $t$ and this number in [1] is atmost $n$. So, these schemes are not suitable to share more than $t$ secrets, while in the proposed scheme and [10] there is no restriction on the number of secrets. The number of public parameters in our scheme is $n + 2r - t + 7$, which is fewer than the number of public parameters in [10]. Distributing the shares between the dealer and participants in [1, 12] needs a secure channel, while in our scheme it is provided by a public channel in such a way that each participant generates their own share. Furthermore, in our scheme combiner is able to verify the validity of the shares submitted by participants during the reconstruction phase. In table 1 we summarize the comparison of our, Shi, Wang, Binu and Patel's schemes. This comparison includes:
(1): The number of shared secrets,
(2): The number of public parameters for sharing $m$ secrets among $n$ participants for a $(t, n)$-threshold scheme,
(3): The verifiability of secret shadows,
(4): The need for a secure channel during share distribution.

Now, we analyze the computational cost of the efficiency of the proposed scheme. The performance analysis and comparison with mentained schemes can be viewed from the following notions in table 2.

In the initialization and distribution phases the dealer and each participant choose their secret key and they compute public keys. So, the computational cost for computing the common keys $e(s_i, G)$ and $h(e(s_i, G))$ is $(n+1)T_M + nT_P$

Table 1: Comparison of the efficiency

| Schemes | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| Binu [1] | $\leq n$ | $9+n+m$ | Yes | Yes |
| Patel [10] | unrestricted | $2n+m+2t+7$ | Yes | No |
| Shi [12] | $\leq t$ | $\leq 6+3n+t$ | Yes | Yes |
| Wang [14] | $\leq t$ | $2n+7$ | Yes | No |
| Proposed | unrestricted | $n+2r-t+7$ | Yes | No |

| | |
|---|---|
| $n$: | The total number of participants |
| $m$: | The total number of secrets |
| $t$: | The number of participants in reconstruction phase |
| $T_M$: | The time for scalar multiplication on elliptic curve |
| $T_S$: | The time for addition on elliptic curve |
| $T_P$: | The time for computing pairing operation on elliptic curve |
| $T_H$: | The time for computing one way hash function |
| $T_{DK}$: | The time for executing a double knapsack algorithm |
| $T_L(t)$: | The time for evaluating a Lagrange polynomial interpolation of degree $t-1$ |
| $T_E(t)$: | The time to solve a system of $t$ equations |

and $rT_H$ respectively. Thus, the total computational cost in these two phases is $(n+1)T_M+nT_P+rT_H$. Note that addition and multiplication in finite fields have lower computational cost than other operations, therefore, we do not count them.

In the verification phase combiner to verify the honesty of the participant $U_i$ just needs to compute $e(s_i, dG)$, which the total computational cost is $tT_P$.

In the reconstruction phase, combiner computes $h(e(s_i, G))$ and solves a system of $(n-t)$ linear equations. The total computational cost in this phase is $tT_P + tT_H + T_E(n-t)$.

Table 2: Computational cost analysis

| Schemes | Distribution | Verification | Reconstruction |
|---|---|---|---|
| Binu [1] | $6T_M + (m+3)T_S + (m+n+1)T_P$ | $tT_P$ | $T_L(t) + 2T_M + (m+1)T_S + mT_P$ |
| Patel [10] | $(2n+1)T_M + T_{DK}$ | $2nT_M + nT_{DK}$ | $T_L(m)$ |
| Shi [12] | $2nT_M + nT_S$ | $t(t+2)T_M + tT_S$ | $T_E(t)$ |
| Wang [14] | $2nT_M + nT_H$ | $2tT_P$ | $tT_M + tT_H + T_E(t)$ |
| Proposed | $(n+1)T_M + nT_P + rT_H$ | $tT_P$ | $tT_P + tT_H + T_E(n-t)$ |

# 6. An Example

We describe our proposed scheme by presenting an example. In this example, we would like to say that how a dealer $D$ distributes two secrets among five participants $U_1$, $U_2$, $U_3$, $U_4$, and $U_5$, in such a way that any group with three participants can recover the secrets by pooling their shares. We use GP/PARI software for implementing the scheme.

## 6.1. Initialization Phase

1. Suppose that the dealer chooses $q = 29^5$ and elliptic curve $E : y^2 = x^3+5x+1$ over the Field $\mathbb{F}_q = \mathbb{F}_{29}(z)$, where $z$ is a root of the polynomial $x^5 + x^4 + 25x^3 + 26x^2 + 3x + 1 \in \mathbb{F}_{29}[x]$. The order of $E(\mathbb{F}_q)$ is $2^5 \times 11 \times 71 \times 821$.

2. The dealer selects $l = 821$. We have $E[l] \cong \mathbb{Z}_l \oplus \mathbb{Z}_l$. Let $P$, $Q$ and $G$ be the points that are randomly selected by the dealer, such that $\{P, Q\}$ is a set of generators for $E[l]$ and $G \in E[l]$. In this example, let

   $$\begin{aligned}
   &P = [19z^4 + 17z^3 + 19z^2 + 14z + 26, 27z^4 + 24z^3 + z^2 + z + 3],\\
   &Q = [11z^4 + 12z^3 + 5z^2 + 18z + 19, 4z^4 + 15z^3 + 3z^2 + 26z + 10],\\
   &G = [15z^4 + 16z^3 + 13z^2 + z + 22, 9z^4 + 16z^2 + 24z + 25].
   \end{aligned}$$

   The dealer selects the integer $d = 500$ as a private key, and he/she computes: $dG = [16z^4 + 13z^3 + 3z^2 + 23z + 28, 24z^4 + 17z^3 + 27z^2 + 25z]$.
   The dealer selects $\alpha = 341$ and $\beta = 875$, computes

   $$\begin{aligned}
   W &= \alpha P + \beta Q\\
   &= [8z^4 + 8z^3 + 4z^2 + 7z + 16, 14z^4 + 8z^3 + 14z^2 + 6z + 17]
   \end{aligned}$$

   and he/she considers the pairing map

   $$\begin{gathered}
   e : E[l] \times E[l] \to E[l]\\
   e(H_1, H_2) = (a_1b_2 - a_2b_1)(\alpha P + \beta Q)
   \end{gathered}$$

   for any $H_1 = a_1P + b_1Q$ and $H_2 = a_2P + b_2Q$.

3. The dealer considers a hash function $h : E[l] \to \mathbb{Z}_l^*$.

4. $D$ publishes $\{E, q, l, G, dG, e, h\}$.

5. The participant $U_i$ $(1 \leq i \leq 5)$ chooses $s_i \in E[l]$, computes $S_i = e(s_i, dG)$ and then he/she publishes it. Here,

   $$\begin{aligned}
   &S_1 = [27z^4 + 2z^2 + 7z + 22, 21z^4 + z^3 + 22z^2 + 10z + 25],\\
   &S_2 = [10z^4 + 20z^3 + 28z^2 + 14z + 6, 26z^4 + 2z^3 + 6z^2 + 16z + 19],\\
   &S_3 = [26z^4 + 22z^3 + 11z^2 + 11z + 22, 7z^4 + 19z^3 + 19z^2 + 17z + 8],\\
   &S_4 = [2z^4 + 27z^3 + 19z^2 + 27z + 21, 26z^4 + 6z^3 + 21z^2 + 24z + 22],\\
   &S_5 = [12z^4 + 10z^3 + 19z^2 + 26z + 15, 4z^4 + 7z^3 + 17z^2 + 25z + 27].
   \end{aligned}$$

6. In this step the dealer computes $d^{-1}e(s_i, dG)$ and then he/she obtains:

$$\begin{aligned}
e(s_1, G) &= [15z^4 + 16z^3 + 3z^2 + 24z + 25, 10z^4 + 17z^3 + 6z^2 + 13z + 10],\\
e(s_2, G) &= [25z^4 + 13z^3 + 13z^2 + 22z + 20, 4z^4 + 13z^3 + 18z^2 + 26z + 19],\\
e(s_3, G) &= [9z^4 + z^3 + 28z + 23, 27z^4 + 9z^3 + 19z^2 + 19z + 24],\\
e(s_4, G) &= [3z^4 + 5z^3 + 16z^2 + 8z + 14, 22z^4 + 2z^3 + 2z^2 + 24z],\\
e(s_5, G) &= [12z^4 + 20z^3 + 3z^2 + 17z, 15z^4 + 2z^3 + 16z^2 + 3z + 10].
\end{aligned}$$

Since $e(s_i, G) \neq e(s_j, G)$ for all $i \neq j$, then the dealer accepts $s_i$ for $1 \leq i \leq 5$.

## 6.2. Points Sharing Phase

Suppose that $K_1 = 25$ and $K_2 = 764$ are two secrets that will be shared and let $h(e(s_1, G)) = 212$, $h(e(s_2, G)) = 97$, $h(e(s_3, G)) = 370$, $h(e(s_4, G)) = 200$ and $h(e(s_5, G)) = 54$. The dealer secretly and randomly chooses $K_3 = 20$, $K_4 = 500$ and $K_5 = 320$.

1. The dealer computes and publishes: $R_1 = 746, R_2 = 779, R_3 = 583, R_4 = 412, R_5 = 378$.

2. The dealer computes

$$A \times X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 2^2 & 2^3 & 2^4 \end{bmatrix} \begin{bmatrix} h(e(s_1, G)) \\ h(e(s_2, G)) \\ h(e(s_3, G)) \\ h(e(s_4, G)) \\ h(e(s_5, G)) \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}, \tag{4}$$

and then $D$ publishes $I_1 = 112$ and $I_2 = 245$.

## 6.3. Secrets Reconstruction and Verification Phase

Let participants $U_1$, $U_2$ and $U_3$ want to pool their shares together and reconstruct all the secrets. When the combiner receives $s_1$, $s_2$ and $s_3$, he/she firstly computes $e(s_i, dG)$. If $e(s_i, dG) = S_i$ for $i = 1, 2, 3$, then verification of the shares is confirmed. Next, the combiner computes $h(e(s_i, G))$ for $i = 1, 2, 3$. Therefore, three unknowns in (4) are determined and the other two variables can be obtained by solving the system of two equations and two unknowns. Finally, the secrets can be reconstructed as follows:

$$\begin{aligned}
K_1 &= R_1 - \textstyle\sum_{j=1, j\neq 1}^{5} h(e(s_j, G) = 25,\\
K_2 &= R_2 - \textstyle\sum_{j=1, j\neq 2}^{5} h(e(s_j, G) = 764.
\end{aligned}$$

**Acknowledgment.** This research was in part supported by the University of Kashan under grant number 682460/001.

**Conflicts of Interest.** The authors declare that there are no conflicts of interest regarding the publication of this article.

# References

[1] V. P. Binu and A. Sreekumar, Threshold Multi Secret Sharing Using Elliptic Curve and Pairing, *Int. J. Inform. Process* **9** (4) (2015) 100 – 112.

[2] G. Blakley, Safeguarding cryptographic keys, *Proc AFIPS* 1979 *National Computer Conference*, AFIPS Press, New york, 1979, 313 – 317.

[3] W. Chen, X. Long, Y. B. Bai and X. P. Gao, A new dynamic threshold secret sharing scheme from bilinear maps, *International Conference on Parallel Processing Workshops* (ICPPW 2007), Xian, 2007, p. 19. `DOI: 10.1109/ICPPW.2007.10`.

[4] B. Chor, S. Goldwasser, S. Micali and B. Awerbuch, Verifiable secret sharing and achieving simultaneity in the presence of faults [A], 26 *th Annual Symposium on Foundations of Computer Science* (sfcs 1985), Portland, OR, USA, 1985, pp. 383 – 395, `DOI: 10.1109/SFCS.1985.64.`

[5] L. Harn, Efficient sharing (broadcasting) of multiple secret, *in IEE Proceedings - Computers and Digital Techniques,* **142** (3) (1995) 237 – 240.

[6] N. Koblitz, Elliptic curve cryptosystems, *Math. Comp.* **48** (177) (1987) 203 – 209.

[7] H. S. Lee, A self-pairing map and its applications to cryptography, *Appl. Math. Comput.* **151** (3) (2004) 671 – 678.

[8] D. Liu, D. Huang, P. Luo and Y. Dai, New schemes for sharing points on an elliptic curve, *Comput. Math. Appl.* **56** (6) (2008) 1556 – 1561.

[9] V. Miller, Use of elliptic curves in cryptography, Advances in cryptology-CRYPTO '85 (Santa Barbara, Calif., 1985), 417–426, *Lecture Notes in Comput. Sci.*, 218, Springer, Berlin, 1986.

[10] N. Patel, P. D. Vyavahare and M. Panchal, A Novel Verifiable Multi-Secret Sharing Scheme Based on Elliptic Curve Cryptography, *The Tenth International Conference on Emerging Security Information, Systems and Technologies*, 2016.

[11] A. Shamir, How to share a secret, *Comm. ACM* **22** (11) (1979) 612 – 613.

[12] R. Shi, H. Zhong and L. Huang, A (a(t, n)-threshold verified multi-secret sharing scheme based on ecdlp, *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing* (SNPD 2007), Qingdao, 2007, pp. 9–13, `DOI: 10.1109/SNPD.2007.416.`

[13] C. Tang, D. Pei, Z. Liu and Y. He, Non-interactive and informationtheoretic secure publicly verifiable secret sharing, *Cryptology ePrint Archive, Report* 2004/201, 2004, (available at `http://eprint.iacr.org/`).

[14] S. J. Wang, Y. R. Tsai and C. C. Shen, Verifiable threshold scheme in multi-secret sharing distributions upon extensions of ecc, *Wireless Pers. Commun.* **56** (1) (2011) 173 – 182.

Khadijeh Eslami
Department of Pure Mathematics,
Faculty of Mathematical Sciences,
University of Kashan,
Kashan, I. R. Iran
e-mail: kh.eslami@grad.kashanu.ac.ir

Mojtaba Bahramian
Department of Pure Mathematics,
Faculty of Mathematical Sciences,
University of Kashan,
Kashan, I. R. Iran
e-mail: bahramianh@kashanu.ac.ir