Chapter 8

Generalized Multi-secret Sharing based on Elliptic Curve and Pairing

8.1 Introduction

Use of elliptic curve and pairing in secret sharing is gaining more importance. The use of elliptic curve helps to improve the security and also the computational complexity is reduced. In this chapter we propose a multi secret sharing scheme with monotone generalized access structure. The scheme makes use of Shamir’s scheme and Elliptic Curve pairing for the implementation. The shares are chosen by the participant itself, so the consistency of the shares are ensured in this scheme. The participant shares remain secret during the reconstruction phase and this provides

Some results of this chapter are included in the following paper.

multi use facility where same share can be used for the reconstruction of multiple secrets. The shared secret, access structure or the participant set can be modified without updating the secret shadow of each participant. This provides dynamism and adds more flexibility to the scheme. The combiner can also verify the shares of the other participants during the reconstruction phase in order to identify the cheaters. The cheating detection and cheater identification is done by using bilinear pairing. This scheme is simple and easy to implement compared with other generalized multi secret sharing scheme with extended capabilities using pairing. The important properties of this proposed scheme are

- Generalized access structure.
- Multi secret sharing.
- Multi use, where each participant has to keep only a single share and can be reused.
- Dynamic, participant or access structure can be modified.
- No secure channel is required.
- Cheater Identification facility.
- Consistency of the shares can be verified.

The use of elliptic curve and pairing have found applications in secret sharing schemes very recently. Several schemes based on threshold and generalized secret sharing is proposed and they have found useful applications. Pairing can be used to introduce verifiability and cheating detection in secret sharing scheme with more security. Chen Wei et al [218] in 2007 proposed a dynamic threshold secret sharing scheme based on bilinear maps. A threshold multi secret sharing scheme based on elliptic curve discrete logarithm is proposed by Runhua Shi et al [194] in
8.2. Pairing and Secret Sharing

2007. Sharing multiple secrets which are represented as points on elliptic curve using self pairing [133] is proposed by Liu et al [137] in 2008. Wang et al [216] proposed a verifiable threshold multi secret sharing scheme in 2009. In Wang’s et al scheme, the number of secrets must be less than or equal to the threshold and also more public values must be changed when the secret need to be updated. Eslami et al [69] in 2010 proposed a modified scheme which avoids these problems. Several publicly verifiable secret sharing schemes are proposed based on pairing. But most of them are single secret sharing schemes [212] [220] [223]. An efficient One Stage Multi Secret Sharing(OSMSS) is proposed recently in 2014 by Fatemi et al [70]. Generalized secret sharing with monotone access structure is also proposed using Elliptic Curve and Bilinear Pairing with capability to detect cheating [100] [226].

8.2 Pairing and Secret Sharing

Pairing on elliptic curves have a number of important cryptographic applications. While first used for cryptanalysis, pairings have since been used to construct many cryptographic systems for which no other efficient implementation is known, such as identity based encryption, attribute based encryption [67] etc. The mapping allows development of new cryptographic schemes based on the reduction of one problem in one group to a different, usually easier problem in the other group. The first group is usually called GAP Diffie-Hellman Group, where the Decisional Diffie Hellman problem (DDHP) [29] is easy. But the Computational Diffie Hellman (CDHP) problem remains hard.

Let $G$ be a cyclic additive group generated by $P$ whose order is prime $q$. For all $a, b, c \in \mathbb{Z}_q^*$. The CDHP is, given $P, aP, bP$, compute $abP$. DDHP is defined as, given $P, aP, bP, cP$, decide whether $c = ab$ in $\mathbb{Z}_q^*$. 

191
The important pairing based construct is the bilinear pairing. A mapping from \((G_1 = \langle P \rangle, +)\) to \((G_2, \cdot)\), two groups of the same prime order \(q\) and the DLP is hard in both the groups is called Bilinear Maps, if the following conditions are satisfied.

1. **Bilinearity**: \(\forall P, Q \in G_1, \forall a, b \in \mathbb{Z}_q^*\)
   \[
e(aP, bQ) = e(P, Q)^{ab}\]

2. **Non-Degeneracy**: If everything maps to the identity, that’s obviously not interesting
   \[
   \forall P \in G_1, P \neq 0 \Rightarrow \langle e(P, P) \rangle = G_2 \quad (e(P, P) \text{ generates } G_2)
   \]
   In other words:
   \[
P \neq 0 \Rightarrow e(P, P) \neq 1
   \]

3. **Computability**: \(e\) is efficiently computable. i.e., there is a polynomial time algorithm to compute \(e(P, Q) \in G_2\), for all \(P, Q \in G_1\).

We can find \(G_1\) and \(G_2\) where these properties hold. The Weil and Tate pairings prove the existence of such constructions [7] [30]. These pairing have found numerous cryptographic applications [115]. Typically, \(G_1\) is an elliptic curve group and \(G_2\) is a finite field.

### 8.3 Proposed Secret Sharing Scheme

The proposed scheme makes use of Shamir’s scheme and also Elliptic Curve Pairing for the implementation. The scheme can be used to share multiple secrets with out changing the participant share. The process of sharing a single secret is mentioned below. The scheme can be extended
to share multiple secrets $K_1, K_2, \ldots, K_m$. Let $P_1, P_2, \ldots, P_m$ be the set of participants involved. A monotone access structure with minimal qualified set $A^0 = \{A_1, A_2, \ldots, A_t\}$ is considered. The scheme also uses a public notice board where every user have the access, but only the Dealer can write or modify the data. The participant select their shares and are kept secret. The problem with Dealer sending inconsistent shares can thus be avoided. The scheme is also multi use i.e., the same share can be used for sharing several secret. The dynamic nature of the scheme allows the participants set or the access structure to be changed without changing the existing participant’s share. This adds more flexibility to the scheme. The use of Elliptic Curve makes the scheme more robust and secure.

The Secret Sharing Scheme consist of four important phases.

1. Initialization.
2. Share generation.
4. Verification and Secret Reconstruction.

### 8.3.1 Initialization

The initialization phase need to be executed only once for a particular secret sharing scheme. It is assumed that the Dealer is a trusted authority and there are $n$ authenticated participants $P_1, P_2, \ldots, P_n$. In the initialization phase some public parameters are posted on the public bulletin called notice board which can be accessed by every participant.

1. The Dealer (D) chooses an elliptic curve $E$ over $GF(q)$, where $q = p^r$ and $p$ is a large prime such that DLP and ECDLP in $GF(q)$ is hard. Let $G_1$ and $G_2$ be two cyclic group of order $q$ for some large prime $p$. $G_1$ is an additive group of points of an elliptic curve over $F_p$ and $G_2$
Generalized Multi-secret Sharing based on Elliptic Curve and Pairing

is multiplicative sub group of an extension of finite field \( \mathbb{F}_{p^2}^* \). Elliptic curve pairing can be used to map elements from group \( G_1 \) to \( G_2 \).

Modified Weil pairing \( \hat{e} \) is used for the implementation.

2. D chooses a generator \( G \) of \( G_1 \) and also defines a hash function \( H \) which maps \( H : G_1 \rightarrow \{0,1\}^l \), where \( l \) is the bit length of the field.

3. D then publish \( \{E, G_1, G_2, q, \hat{e}, G, H\} \) in the notice board.

8.3.2 Share Generation

In this phase shares of secrets are generated. The shares are not generated by the Dealer instead they are selected by the participants and send to the Dealer. Dealer will verify the shares and assign the shares corresponds to each participants and publish them in the public notice board.

1. Each participant \( P_i \) select a random number \( X_i \in \mathbb{Z}_q^* \) and compute \( Y_i = X_i G \). The participant will keep \( X_i \) secret and send \( Y_i \) to the Dealer.

The Dealer needs to ensure that these \( Y_i \)'s are distinct to make sure that each participant is using different shares. If \( Y_i = Y_j \) for some \( P_i \neq P_j \), then the Dealer will ask for new share. It is noted that only the pseudo shares are send to the Dealer. An intruder or the Dealer cannot obtain any information about the secret share because ECDLP in the field is hard to solve.

2. The Dealer then publish the pseudo shares \( (p_i, Y_i) \) corresponds to each participants, where \( p_i \in \mathbb{Z}_q^* \) is the public identity corresponds to each participant \( P_1, P_2, \ldots, P_n \) chosen randomly.
8.3. Proposed Secret Sharing Scheme

8.3.3 Secret Distribution

In this phase the Dealer will share the secret by publishing informations on a public notice board corresponds to the secret to be shared. Each participant can make use of these public values. These public information together with the secret share of each participant can be used for the retrieval of shared secret. Sharing of a single secret value is mentioned here. The same steps can be followed to share multiple secrets.

1. Let $K$ be the secret to be shared. Dealer will set up a polynomial $f(x)$ of degree 1.
   i.e., $f(x) = K + bx$, where $b \in \mathbb{Z}_q^*$.  

2. For each minimal qualified subset in $A^0$, an integer $a_1, a_2, \ldots, a_t \in \mathbb{Z}_q^*$ is chosen to represent the $t$ qualified subsets.

3. Choose a random number $X_0 \in \mathbb{Z}_q^*$ and compute $Y_0 = X_0G$ also $Y_i' = X_0Y_i$, for $i = 1, 2, \ldots, n$.

4. Compute $f(1)$ and for each qualified subset $A_j = \{P_{1j}, P_{2j}, \ldots, P_{dj}\}$ in $A^0$, compute
   \[ A_j = f(a_j) \oplus H(Y_{1j}') \oplus H(Y_{2j}') \oplus \cdots \oplus H(Y_{dj}') , \]  
   $1 \leq j \leq t$.

5. Publish $Y_0, f(1), (a_1, A_1), (a_2, A_2), \ldots, (a_t, A_t)$ on the public bulletin.

8.3.4 Verification and Secret Reconstruction

The participants in each qualified subset $A_j = \{P_{1j}, P_{2j}, \ldots, P_{dj}\}$, $1 \leq j \leq t$ can reconstruct the secret using the secret share and also the public values in the bulletin board. Each user contribute his pseudo share for the reconstruction of secret. The pseudo share is computed from his secret share and the public informations. The designated combiner can also identify the cheaters during the reconstruction phase using pairing.
Generalized Multi-secret Sharing based on Elliptic Curve and Pairing

1. Each participant $P_{ij}$ in the qualified subset $A_j$ get the public information $Y_0$ from the bulletin board and computes $Y_{ij}' = Y_0X_{ij}$, using the secret share $X_{ij}$. The participant then delivers $Y_{ij}'$ to the designated combiner.

2. Combiner checks $\hat{e}(G, Y_{ij}') = \hat{e}(Y_0, Y_{ij})$. If it is not true then the share send by the participant $P_{ij}$ is invalid and corrective measures have to be taken in this case.

3. Once all the valid shares are received, the combiner can retrieve $f(a_j) = A_j \oplus H(Y_{1j}') \oplus H(Y_{2j}') \oplus \cdots \oplus H(Y_{dj}')$.

4. Using $f(1)$ and $f(a_j)$, the polynomial can be reconstructed using Lagrange Interpolation [120],

$$f(x) = f(1).\frac{x - a_j}{1 - a_j} + f(a_j).\frac{x - 1}{a_j - 1} \quad (8.1)$$

5. The shared secret $K$ is $f(0)$.

8.4 Security Analysis

One of the major requirement of a secret sharing scheme is the secure distribution of the shares to the participants by the Dealer. An untrustable Dealer may send inconsistent shares to the participant. The verifiable secret sharing ensures that the shares are consistent i.e., the authorized set of shares when combined will generate the same secret. Here the secret shares are chosen by the participant itself and send to the Dealer during the share generation. The pseudo shares are also used during the reconstruction. So the Dealer or any other participant does not have any idea about the secret share chosen by the participant. Combiner can also verify the shares send by the participants using this pseudo shares. Finding $X_i$ from $Y_i$ or finding $X_0$ from $Y_0$ is hard as solving the ECDLP. Hence the security of the secret
8.4. Security Analysis

share depends on the hardness of solving ECDLP. It is noted that there is no efficient algorithm exist for solving the ECDLP. The birthday paradox method can be used to solve ECDLP which is having a running time of $O(n)$, where $n$ is the order of the group. This attack can be broken by choosing a field of size at least 160 bits. In order to overcome the attack of the sub exponential algorithm used for solving discrete logarithm problem, we need a field of at least 1024 bit size. This shows that elliptic curve group provides more security with less number of bits. Thus the use of elliptic curve field instead of finite field results in savings of both time and space.

Cheating detection and Cheater identification is a major security requirement. It is done very efficiently using elliptic curve pairing. An efficient algorithm is proposed by Victor Miller [148] for computing the Weil pairing, which is having polynomial time complexity.

**Theorem 8.4.1.** The probability that the participant distribute invalid shares during the reconstruction is negligible.

**proof.** The designated combiner can verify the shares send by the participants by checking $\hat{e}(G,Y_{ij}') = \hat{e}(Y_0,Y_{ij})$. If there is a mismatch, the participant is a cheater and we are able to detect and identify the cheater.

From the properties of bilinear pairing

\[
\begin{align*}
\hat{e}(G,Y_{ij}') &= \hat{e}(Y_0,Y_{ij}) \\
\hat{e}(G,Y_0X_{ij}) &= \hat{e}(GX_0,GX_{ij}) \\
\hat{e}(G,Y_0X_{ij}) &= \hat{e}(GX_0,GX_{ij}) \\
\hat{e}(G,GX_0X_{ij}) &= \hat{e}(GX_0,GX_{ij}) \\
\hat{e}(G,G)^{X_0X_{ij}} &= \hat{e}(G,G)^{X_0X_{ij}}
\end{align*}
\]

If this equation doesn‘t hold then the participant is a cheater.
Generalized Multi-secret Sharing based on Elliptic Curve and Pairing

During the reconstruction phase each participant submit $Y'_{ij} = X_{ij}Y_0$. The participant does not have to disclose his secret share $X_{ij}$. An attacker has to solve the ECDLP to find the secret share from the pseudo share which is computationally hard. The secret share can be reused with out compromising the security of the scheme. This provides more flexibility unlike other secret sharing schemes.

The list of participants in the authorized access structure can only reconstruct the secret. This is achieved with Shamir’s secret sharing scheme. It is noted that Shamir’s scheme provides information theoretic security. The security does not depends on the assumptions about any hard mathematical problem. The polynomial used in the scheme is having only degree one which makes the scheme computationally efficient. In order to reconstruct the degree one polynomial, two points are necessary. $f(1)$ is published in the bulletin board. The other value $f(a_j)$ can only be computed by the list of authorized participant in each authorized subset ($A_j$) mentioned in the minimal qualified set ($A^0$). The participants not mentioned in the authorized subset cannot obtain $f(a_j)$ and hence cannot reconstruct the polynomial. No information about the secret $f(0)$ is thus revealed. Lagrange Interpolation can be done efficiently in $O(n \log^2 n)$ time. However the reconstruction of the degree 1 polynomial from two points $(1, f(1))$ and $(a_j, f(a_j))$ can be done with four multiplication and an inverse computation. The addition and subtraction does not cost much.

The scheme can be easily extended to share multiple secrets. The Dealer has to construct polynomials of degree 1 corresponds to each secret $K_1, K_2, \ldots, K_m$ to be shared. The Dealer then publish $f(1)_i, Y_{0i}$ and $A_{ji}$, $1 \leq i \leq m$ corresponds to each secret along with other public parameters. It is noted that only public parameters need to be added to share more secrets. However participant shares remain same for the multi secret sharing.
8.4. Security Analysis

The proposed scheme is dynamic in nature. When the Dealer wants to share a new secret, he just modify the public parameters. The participant does not need to alter their secret shadows. When a participant wants to change his secret shadow in case of leakage, he can do so. He will send the pseudo share to the Dealer after choosing a new secret share. The Dealer then modify the public parameters accordingly. This process does not affect the secret share of other participants. When a new participant joins the system and the access structure changes, then only the public parameters need to be changed. Other participants need not renew their secret shadows unlike other secret sharing schemes.

The verifiability is implicit in the scheme. Since the secret shadows are chosen by the participant itself, an adversary or the Dealer have no idea about the secret shadows. This also provides multi use capability where same shadows can be used for the sharing and reconstruction of several secrets.

It is noted that there is no secret communication exist between the Dealer and the participants. So the scheme avoids the need of a secure channel. The scheme is also efficient because of the low computational cost. Let us define the following terms to represent the time taken for each operation.

\[ T_{ECM} \] – The time taken for computing \( nX \), where \( n \) is a scalar and \( X \) is an elliptic curve point.

\[ T_P \] – The time taken for Pairing.

\[ T_H \] – The time taken for executing the Hash function \( H \).

\[ T_L \] – Time for polynomial reconstruction.

\( n \) – Total number of participants.

\( d \) – Number of participants in each qualified access set.
In the system initialization phase, each entity including the participants and the Dealer has to compute a public share from his secret share. This needs \((n + 1)\) point multiplication. The cost is \((n + 1)T_{ECM}\). Also the hash function has to be computed for each \(d\) participants set in the access structure. This need a computational cost of \(dT_H\). The cost of XOR and the polynomial evaluation does not cost much. So the total computational cost in the initialization and the secret distribution phase is \(O((n + 1)T_{ECM} + dT_H)\). During the verification and secret reconstruction phase, each participant has to do a point multiplication. The combiner has to do two pairing operation for verification. The hash operations has to be done for each participant share involved in the secret reconstruction. The final secret is obtained by Lagrange Interpolation. It is noted that the polynomial used is of degree one. So the interpolation doesn’t take too much computational cost. The computational cost involved in XOR operations is also negligible. Thus the total computational cost involved in the secret reconstruction and verification phase is \(dT_{ECM} + 2TP + dT_H + T_L\). It is noted that the overall computational cost depends mainly on the point multiplication by a constant. If \(X\) is an elliptic curve point then \(nX\) can be done efficiently by double-and-add method. Suppose \(n = 2^k - 1\) then in the worst case it needs \(2k\) point operations i.e., \(k\) additions and \(k\) multiplications. If we use ternary expansion then computing \(nX\) never requires more than \(\frac{3}{2}k + 1\) point operations i.e., \(k + 1\) doubling and \(\frac{1}{2}k\) additions.

### 8.5 Concluding Remarks

We have proposed a novel generalized multi secret sharing scheme based on elliptic curve and bilinear pairing in this chapter. The scheme is computationally efficient and provides more security. The scheme is multi use and dynamic in nature. Participants can be added, the access
8.5. Concluding Remarks

structure can be changed or more secrets can be shared without changing participants' shadow. The shares are chosen by the participant itself. It avoids the verifiability problem and also the need for a secure channel. The pairing helps to identify the cheating and also to detect the cheaters. Unlike other multi secret sharing schemes, it is simple and easy to implement. The number of public parameters is also less. When a participant leaves the system, the access structure changes. We have to remove the participant entry from all the sets where the participant belongs and then modify the public parameters according to the new access structure. We have also done a detailed analysis of the proposed algorithm and mentioned the complexities involved in terms of the complexity of the elliptic curve point operations, pairing, time taken for hash function and Lagrange interpolation. One disadvantage with this scheme is that it cannot reconstruct the shared multiple secrets simultaneously.