Sequential Secret Sharing as a New Hierarchical Access Structure

Mehrdad Nojoumian *
Department of Computer and Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, Florida, USA
mnojoumian@fau.edu

Douglas R. Stinson †
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
dstinson@uwaterloo.ca

Abstract
Due to the rapid growth of the next generation networking and system technologies, computer networks require new design and management. In this context, security, and more specifically, access structures have been one of the major concerns. As such, in this article, sequential secret sharing (SQS), as an application of dynamic threshold schemes, is introduced. In this new cryptographic primitive, different (but related) secrets with increasing thresholds are shared among a set of players who have different levels of authority. Subsequently, each subset of the players can only recover the secret in their own level. Finally, the master secret will be revealed if all the secrets in the higher levels are first recovered. We briefly review the existing threshold modification techniques. We then present our construction and compare it with other hierarchical secret sharing schemes such as disjunctive and conjunctive multilevel secret sharing protocols.

Keywords: Secret Sharing, Access Structure, Dynamic Scheme, Threshold Changeability.

1 Introduction
In a \((t,n)\)-threshold secret sharing [10, 2], a dealer first divides a secret into \(n\) shares to be distributed among \(n\) players. Subsequently, at least \(t\) players can collaborate to recover the secret. For instance, in Shamir secret sharing [10], the dealer initially selects a random polynomial \(f(x) \in \mathbb{Z}_q[x]\) of degree \(t - 1\) such that \(f(0)\) is the secret. He then distributes shares \(f(i)\) among players \(P_i\) for \(1 \leq i \leq n\). As a result, any set of \(t\) or more players can recover the secret using Lagrange interpolation whereas any set of size less than \(t\) cannot gain any information about the secret. Mainly, two adversarial settings are considered in secret sharing schemes. Passive adversary model where the players follow protocols correctly but they may attempt to learn the secret, also known as honest-but-curious adversary. Active adversary model where the players may deviate from protocols while at the same time trying to learn the secret. For further technical discussions, a formal definition of an access structure is first provided.

**Definition.1:** An access structure \(\Gamma\) is a set of authorized subsets of players that satisfies two conditions: (a) if \(A \in \Gamma\) and \(A \subseteq B \subseteq \mathcal{P}\) where \(\mathcal{P}\) is the finite set of the players, then \(B \in \Gamma\), and (b) if \(A \in \Gamma\) then \(|A| > 0\). In a threshold access structure, authorized subsets are all sets of players \(A\) such that \(|A| \geq t\) where \(t\) is the threshold of the scheme.

*Corresponding author: College of Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431.
†Stinson’s research is supported by NSERC discovery grant 203114-11.
In a dynamic secret sharing scheme, the threshold and/or the access structure are changed frequently. The main motivation for construction of such schemes is the fact that the “sensitivity of the secret” and also the “number of players” may fluctuate due to different reasons. For instance, mutual trust may vary or the structure of the players’ organization might be changed (i.e., some parties may leave or new players may join the organization). To the best of our knowledge, the existing dynamic constructions only update the threshold/access structure without changing the secret.

Indeed, the objective of threshold changeability is to transform a \((t, n)\)-secret sharing scheme into a \((t', n)\)-secret sharing scheme whether \(t < t'\) or \(t' < t\). Note that when the threshold is increased while the secret remains unchanged, at least \(n - t + 1\) players must erase their old shares, otherwise, the secret can be recovered by a set of old shares. This issue has been previously stated as an inevitable assumption of proactive secret sharing \([9, 4]\) as well as threshold changeable schemes with constant secrets \([5]\).

To change the threshold and the secret in our multi-level access structure, secure addition and multiplication operations are used. Let \(\alpha\) and \(\beta\) be two secrets shared by \(f(x)\) and \(g(x)\) of degree \(t\). If each participant locally multiplies both shares together, the resulting value is a new share on \(h(x) = f(x) \times g(x)\), where \(h(0) = \alpha \beta\) is the new secret. There exist two issues with this secure multiplication operation. First of all, the degree of \(h(x)\) is \(2t\) instead of \(t\). Second, \(h(x)\) is reducible as a product of two polynomials.

To overcome these problems, \([1]\) applies a degree reduction method in which \(h(x)\) is truncated in the middle to have a degree of \(t\), subsequently, it uses a simple procedure to randomize the coefficients of \(h(x)\) except its constant term; this protocol is later simplified in \([3]\). The addition operation is also done locally, however, it does not require any degree reduction or randomization.

### 1.1 Motivation and Contribution

We introduce a new hierarchical secret sharing protocol as a new application of dynamic threshold schemes, named sequential secret sharing (SQS). In this cryptographic primitive, players with various levels of authority progressively construct a sequence of secret sharing schemes with different (but related) secrets and thresholds in the absence of the dealer. In the subsequent reconstruction phase, each subset of the players can only recover the secret in their own level. As a result, the master secret will be revealed if all the secrets in the higher levels are first recovered.

In the existing hierarchical secret sharing schemes \([11, 12]\), a single secret is shared among the players who are in different authority levels (players in the initial levels have more authority for secret recovery compared to the other parties). Moreover, the secret can be reconstructed without the contribution of players from all levels, i.e., players from certain levels can recover the secret and the contribution of all players may not be required.

However, in our sequential secret sharing, multiple secrets are first generated using a master secret. These secrets are then shared among the players who are in various authority levels. Furthermore, in our scheme, although the players in the initial level have the required authority to recover the master secret, they cannot do that without the sequential cooperation of the players from all levels, i.e., to recover the master secret, all the secrets must be recovered sequentially. For a realization of our access structure, assume the president and vice president, ministers and senators are in three different authority levels. The president and vice president can recover the master secret (to trigger a secret action) only if they have the confirmations of ministers and senators. On the other hand, even by having those confirmations, the final decision is made by the president and vice president. This access structure cannot be modeled by the existing hierarchical schemes.

Our proposed sequential secret sharing is unconditionally secure so that it does not rely on any computational assumptions such as discrete logarithm. Furthermore, in this scheme, players do not require to store extra shares beforehand to generate the subsequent secrets or to change the threshold to different values. Note that each secret is produced based on the linear combination of previous secrets.
2 Threshold Modification Techniques

Before presenting our sequential secret sharing scheme, the existing threshold modification techniques are briefly reviewed. We only demonstrate these protocols in the passive adversary model. For the active adversary setting, see [7, 6].

The first protocol, as shown in Figure 1, illustrates how re-sharing method (also known as 2-level sharing) can be used to decrease/increase the threshold to any arbitrary values. However, this method only works in the “passive” adversary model and it fails to increase the threshold in an active adversary setting. Note that the re-sharing technique can be implemented either by Lagrange method or by a Vandermonde matrix. Suppose the dealer randomly generates \( f(x) \in \mathbb{Z}_q[x] \) of degree at most \( t-1 \) where \( f(0) = \alpha \) is the secret. He then sends share \( f(i) \) to \( P_i \) for \( 1 \leq i \leq n \). For threshold modification, each player re-shares his share using a new random polynomial of degree at most \( t' - 1 \), as shown in Figure 1.

Threshold Modification: from \( t \) to \( t' \) where \( t' > t \) or \( t' < t \)

1. Each player \( P_i \) selects a random polynomial \( g_i(x) \) of degree at most \( t' - 1 \) such that \( g_i(0) = f(i) \). He then gives \( g_i(j) \) to \( P_j \) for \( 1 \leq j \leq n \), i.e., re-sharing the original shares by auxiliary shares. The share-exchange matrix \( E_{n \times n} \), where each player generates a row and receives a column, is as follows:

\[
E_{n \times n} = \begin{pmatrix}
g_1(1) & g_1(2) & \cdots & g_1(n) 
g_2(1) & g_2(2) & \cdots & g_2(n) 
\vdots & \vdots & \ddots & \vdots 
g_n(1) & g_n(2) & \cdots & g_n(n)
\end{pmatrix}
\]

where \( g_i(0) = f(i) \).

2. At this step, a set \( \Delta \) is determined such that it consists of the identifiers of at least \( t \) elected players. Then, the following public constants are computed:

\[
\gamma_i^\Delta = \prod_{j \in \Delta, j \neq i} \frac{j}{j-i} \quad \text{where } 1 \leq i, j \leq n \text{ represent players' ids.}
\]

3. Each player \( P_j \) erases his old shares, and then combines the auxiliary shares he has received from other players to compute his new share as follows:

\[
\varphi_j = \sum_{i \in \Delta} \left( \gamma_i^\Delta \times g_i(j) \right).
\]

Figure 1: Threshold Modification by Lagrange Method in the Passive Adversary Model

Second protocol shows how public evaluation can be used for threshold reduction in either “passive” or “active” adversary model. In this scheme, players collaborate to reveal an extra share on the secret sharing polynomial using the enrollment protocol of [8]. They then combine this share with their existing shares so that the threshold is decreased but the secret remains unchanged. Let \( f(x) \in \mathbb{Z}_q[x] \) be the original polynomial, see Figure 2.

Third protocol, as shown in Figure 3 demonstrates how the threshold can be increased by zero addition in either “passive” or “active” adversary model. In this scheme, players first generate a random polynomial of higher-degree with zero constant term. They then add shares of this polynomial to their original shares. As a result, the threshold is increased but the secret stays the same.
Threshold Decrease: from $t$ to $t-1$

1. Players select an id $j$ such that $j \notin \mathcal{P}$. Subsequently, $t$ players $P_i$ are selected (1 $\leq i \leq t$). They compute Lagrange constants $\gamma = \prod_{1 \leq k \leq t, i \neq k} \frac{j-k}{i-k}$, where $i, j$ and $k$ are players’ ids.

2. Each $P_i$ multiplies his share $f(i)$ by his Lagrange constant. He then randomly splits the result into $t$ portions, i.e., $f(i) \times \gamma = \partial_1 + \partial_2 + \cdots + \partial_t$ for $1 \leq i \leq t$.

3. Players exchange $\partial_k$’s through pairwise channels. As a result, each player $P_k$ holds $t$ values. He adds them together and reveals $\sigma_k = \sum_{i=1}^{t} \partial_k$ to everyone.

4. All the players add these values $\sigma_k$ for $1 \leq k \leq t$ together to compute the public share $f(j) = \sum_{k=1}^{t} \sigma_k$.

5. Each $P_i$ combines his private share $f(i)$ with the public share $f(j)$ as follows:

$$\hat{f}(i) = f(j) - j \left( \frac{f(i) - f(j)}{i-j} \right).$$

6. Shares $\hat{f}(i)$ are on a new polynomial $\hat{f}(x) \in \mathbb{Z}_q[x]$ of degree at most $t-2$ where $\hat{f}(0) = f(0)$. Therefore, $t-1$ players are now sufficient to recover the secret.

Figure 2: Threshold Decrease by Public Evaluation in the Passive Adversary Model

Threshold Increase: from $t$ to $t'$ where $t' > t$

1. Players use polynomial production to generate shares of an unknown secret $\delta$ on a polynomial $g(x)$ of degree $t' - 2$.

2. Each player $P_i$ multiplies his share $g(i)$ by $i$. Now, each $P_i$ has a share of 0 on the polynomial $\hat{g}(x) = xg(x)$ of degree $t' - 1$.

3. Each player adds his share $f(i)$ of secret $\alpha$ to his share $ig(i)$ of 0. As a result, each player has a share of $\alpha$, where the new threshold is $t' > t$.

Polynomial Production

1. First, $t$ players $P_i$ are selected at random in order to act as independent dealers.

2. Each of the $t$ chosen players $P_i$ shares a secret, say $\delta_i$, among all the players using a Shamir scheme, where the degree of the secret sharing polynomial is $t-1$. Then, all players have shares of every secret $\delta_i$.

3. Every player adds his shares of the $\delta_i$-s together. As a result, each player has a share on a polynomial $g(x)$ of degree $t-1$ with a constant term $\delta = \sum \delta_i$.

Figure 3: Threshold Increase by Zero Addition in the Passive Adversary Model

3 Sequential Secret Sharing (SQS)

We now propose a new hierarchical scheme, named sequential secret sharing, where the threshold and the secret are changed based on the linear combination of the previous unknown secrets. In this protocol, players progressively construct a sequence of secret sharing schemes with different thresholds and secrets in the absence of the dealer, that is, they will modify the threshold while generating multiple secrets. For the sake of simplicity, we just use the addition operation in order to change the secret, however, the multiplication operation can also be used. All computations are performed in finite field $\mathbb{Z}_q$. Let’s first start with an example to make this protocol clear.
Example 1: Suppose the goal is to create a three-level sequential secret sharing scheme among a set of thirteen players. Consider the following subsets of players:

\[\mathcal{P} = \{P_1, \ldots, P_{13}\}, \quad \mathcal{P}_1 = \{P_1, P_2, P_3\}, \]
\[\mathcal{P}' = \{P_4, \ldots, P_{13}\}, \quad \mathcal{P}_2 = \{P_4, P_5, P_6, P_7\}, \]
\[\mathcal{P}_3 = \{P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}\}.\]

**Sharing Phase**

1. The dealer first shares a master secret \(\alpha_1\) with the players in \(\mathcal{P}\) using a \((2, 13)\)-threshold scheme. We denote this sharing by the following notation:

\[\alpha_1 : \mathcal{P} = \{P_1, \ldots, P_{13}\}^{t_0 = 2}.\]

2. (a) The players \(P_i \in \mathcal{P}\) use polynomial production to create shares of an unknown secret \(\beta_1\) having a threshold \(t_1 = 3\).

(b) They add their shares locally to obtain the shares of \(\alpha_2 = \alpha_1 + \beta_1\) which has a threshold of \(t_1 = 3\). All the players erase the shares of \(\alpha_1\).

(c) Players \(\{P_1, \ldots, P_3\}\) only keep the shares of \(\beta_1\), and players \(\{P_4, \ldots, P_{13}\}\) only keep the shares of \(\alpha_2\). Using the notation defined above, the result is denoted by:

\[\beta_1 : \mathcal{P}_1 = \{P_1, P_2, P_3\}^{t_1 = 3} \quad \text{and} \quad \alpha_2 : \mathcal{P}' = \{P_4, \ldots, P_{13}\}^{t_1 = 3}.\]

3. (a) The players \(P_i \in \mathcal{P}'\) use polynomial production to create shares of an unknown secret \(\beta_2\) having a threshold \(t_2 = 4\).

(b) They add their shares locally to obtain the shares of \(\alpha_3 = \alpha_2 + \beta_2\) which has a threshold of \(t_2 = 4\). The players \(P_i \in \mathcal{P}'\) erase the shares of \(\alpha_2\).

(c) Players \(\{P_4, \ldots, P_7\}\) only keep the shares of \(\beta_2\). Also, \(\{P_8, \ldots, P_{13}\}\) increase the threshold from \(t_2 = 4\) to \(t_3 = 6\) and keep the shares of \(\alpha_3\). We denote this by:

\[\beta_2 : \mathcal{P}_2 = \{P_4, \ldots, P_7\}^{t_2 = 4} \quad \text{and} \quad \alpha_3 : \mathcal{P}_3 = \{P_8, \ldots, P_{13}\}^{t_3 = 6}.\]

**Recovery Phase**

1. In the first step, six players \(\mathcal{P}_3 = \{P_8, \ldots, P_{13}\}\) recover the secret \(\alpha_3\). These players are in the highest level.

2. Subsequently, \(\mathcal{P}_2 = \{P_4, \ldots, P_7\}\) recover the secret \(\beta_2\). As a result, \(\alpha_2\) is uniquely revealed since \(\alpha_3 = \alpha_2 + \beta_2\).

3. Finally, \(\mathcal{P}_1 = \{P_1, \ldots, P_3\}\) recover the secret \(\beta_1\). As a result, the master secret \(\alpha_1\) is revealed since \(\alpha_2 = \alpha_1 + \beta_1\).

Note that the above example has \(\ell = 3\) levels and thresholds \(t_0 = 2, t_1 = 3, t_2 = 4\) and \(t_3 = 6\). Again, we emphasize that the above protocol can be also implemented by the multiplication operation if it is required to do so, i.e., using \(\alpha_{i+1} = \alpha_i \beta_i\) rather than \(\alpha_{i+1} = \alpha_i + \beta_i\). In this case, a threshold reduction mechanism must be used after each multiplication, as shown in Figure 2 or its alternative version that is secure under the active adversary model [7, 6]. We now provide the definition of sequential secret sharing and then we demonstrate our protocol in Figure 4.
Definition 2: Sequential secret sharing is a hierarchical secret sharing scheme where a master secret $\alpha_1$ along with $\ell - 1$ secrets $\alpha_2, \ldots, \alpha_{\ell}$ are shared among the players with monotonically increasing thresholds $t_0 < t_1 < \cdots < t_{\ell}$. Let $\mathcal{P}$ be a set of $n$ players and assume $\mathcal{P}$ is composed of $\ell$ disjoint levels

$$\mathcal{P} = \bigcup_{i=1}^{\ell} \mathcal{P}_i,$$

where $\mathcal{P}_i \cap \mathcal{P}_j = \emptyset$ for all $1 \leq i < j \leq \ell$ and $|\mathcal{P}_i| \geq t_i$ for all $i$.

Secret $\alpha_i$ (at level $k$) can be then recovered only if players in $\mathcal{R}_k = \bigcup_{i=k}^{\ell} \mathcal{P}_i$ cooperate and recover their secrets sequentially, i.e., from the highest level $\ell$ down to level $k$, meaning that the master secret $\alpha_1$ can be only recovered by players $\mathcal{P}_1$ only if the players in all levels sequentially reconstruct their secrets.

<table>
<thead>
<tr>
<th>Sharing Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. A dealer uses a Shamir scheme to distribute shares of an initial secret $\alpha_1$ with threshold $t_0$ among players $\mathcal{P} = {P_1, \ldots, P_n}$ and then he leaves the scheme.</td>
</tr>
<tr>
<td>2. Subsequently, players repeat the following steps for $1 \leq i \leq \ell - 1$ to construct an $\ell$-level sequential secret sharing scheme:</td>
</tr>
<tr>
<td>(a) The players in $\mathcal{P}$ use polynomial production protocol, presented in Figure 3, to generate shares of a random secret $\beta_i$ with threshold $t_i$, where $t_{i-1} &lt; t_i$.</td>
</tr>
<tr>
<td>(b) They compute shares of $\alpha_{i+1} = \alpha_i + \beta_i \mod q$; the threshold of $\alpha_{i+1}$ is $t_i$. Then they erase their shares of $\alpha_i$.</td>
</tr>
<tr>
<td>(c) A subset of players, say $\mathcal{P}_i \subset \mathcal{P}$ where $</td>
</tr>
<tr>
<td>(d) If $i = \ell - 1$ (i.e., the last step of the protocol is being executed), they increase the threshold from $t_{\ell-1}$ to $t_\ell$. Otherwise (if $i &lt; \ell - 1$), they set $\mathcal{P} \leftarrow \mathcal{P} \setminus \mathcal{P}_i$.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recovery Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Appropriate subsets of the players first collaborate to recover $\alpha_i$ as well as $\beta_{i-1}, \ldots, \beta_1$. Note that the players may only recover these secrets down to a specific level $i$ if it is intended to do so.</td>
</tr>
<tr>
<td>2. They then solve the following system of linear congruences: $\alpha_{i+1} \equiv \alpha_i + \beta_i \mod q$ for $i = \ell - 1$ down to $i = 1$. (It is clear that each congruence has a unique solution for $\alpha_i$ given $\alpha_{i+1}$ and $\beta_i$.) Therefore, $\alpha_i, \ldots, \alpha_1$ are recovered.</td>
</tr>
</tbody>
</table>

Figure 4: Sequential Secret Sharing Protocol

The security proof of our proposed sequential secret sharing is pretty much similar to Shamir’s secret sharing scheme [10]. In Step-1, the dealer uses this scheme to share the master secret $\alpha_1$ among all the players. In Step-2.a, players use the polynomial production protocol, shown in Figure 3, to generate shares of random numbers $\beta_i$. In this protocol, players simply act as independent dealers and use the Shamir’s secret sharing scheme to generate these random numbers. In Step-2.b, players locally add their shares together to compute shares of the secret $\alpha_i + \beta_i$. They also erase shares of the secret $\alpha_i$ (in the first round, $\alpha_1$ is the master secret). That way shares of the secrets $\alpha_i$ for $1 \leq i \leq \ell - 1$ are erased in the scheme and they cannot be recovered directly, i.e., they can only be reconstructed using $\alpha_i$ and $\beta_{i+1}, \ldots, \beta_1$.

In Step-2.c, players are divided into two disjoint subsets where one set only keeps shares of $\beta_i$ and the other subset only keeps shares of $\alpha_{i+1}$. This means the players in each subset erase the shares associated to the other subset’s secret. Note that in the next iteration, shares of the previous secret $\alpha_{i+1}$ is also erased. Finally, in Step-2.d, players increase the threshold from $t_{\ell-1}$ to $t_\ell$ using a Shamir-based threshold increase protocol, see Figure 4.
It is worth mentioning that the master secret $\alpha_1$ can be recovered correctly because secrets $\alpha_\ell$ and $\beta_{\ell-1}, \ldots, \beta_1$ can be sequentially reconstructed by the Lagrange interpolation method. Furthermore, as we stated earlier, each congruence equation $\alpha_{i+1} \equiv \alpha_i + \beta_i \mod q$ for $i = \ell - 1$ down to $i = 1$ has a unique solution for $\alpha_i$ given $\alpha_{i+1}$ and $\beta_i$.

4 Comparison with the Existing Hierarchical Schemes

The first hierarchical secret sharing scheme is proposed by Simmons [11], named disjunctive multilevel secret sharing. Subsequently, this construction is changed into conjunctive multilevel secret sharing by Tassa [12]. In both constructions, only a single secret is shared among the players who are in various authority levels whereas we generate different (but related) secrets with increasing thresholds in our access structure. We briefly illustrate these two constructions and provide an example for further clarification.

**Definition.3:** In a hierarchical secret sharing scheme, a secret $\alpha$ is shared among the players with monotonically increasing thresholds $t_1 < t_2 < \cdots < t_\ell$. Let $\mathcal{P}$ be a set of $n$ players and assume $\mathcal{P}$ is composed of $\ell$ disjoint levels:

$$\mathcal{P} = \bigcup_{i=1}^{\ell} \mathcal{P}_i, \text{ where } \mathcal{P}_i \cap \mathcal{P}_j = \emptyset \text{ for all } 1 \leq i < j \leq \ell \text{ and } |\mathcal{P}_i| \geq t_i \text{ for all } i.$$

In disjunctive model, secret $\alpha$ can be recovered by a set of players $A$, i.e., an authorized subset, only if

$$|A \cap \bigcup_{i=1}^{\ell} \mathcal{P}_i| \geq t_j \text{ for at least one } j \text{ where } 1 \leq j \leq \ell,$$

i.e., at least one threshold must be satisfied at level 1 to $j$. In conjunctive model, secret $\alpha$ can be recovered by a set of players $A$ only if

$$|A \cap \bigcup_{i=1}^{\ell} \mathcal{P}_i| \geq t_j \text{ for all } j \text{ where } 1 \leq j \leq \ell.$$

**Example.2:** Suppose there exist four levels with $t_1 = 2$, $t_2 = 3$, $t_3 = 4$ and $t_4 = 6$ thresholds. An authorized subset $A$ is shown in Table 1. It’s clear that, in both schemes, the players in the initial levels have more authority compared to the other players. For instance, two players from the first level are enough to recover the secret in the disjunctive model. Also, six players from the first level are enough to reconstruct the secret in the conjunctive model; note that six players from level 2, 3 or 4 won’t be able recover the secret in this model.

<table>
<thead>
<tr>
<th>Authorized Set</th>
<th>Disjunctive</th>
<th>Conjunctive</th>
</tr>
</thead>
<tbody>
<tr>
<td>at least 2 players from level 1</td>
<td>at least 2 players from level 1</td>
<td>and</td>
</tr>
<tr>
<td>or</td>
<td>or</td>
<td>and</td>
</tr>
<tr>
<td>at least 3 players from levels 1 or 2</td>
<td>at least 3 players from levels 1 or 2</td>
<td>and</td>
</tr>
<tr>
<td>or</td>
<td>or</td>
<td>and</td>
</tr>
<tr>
<td>at least 4 players from levels 1, 2 or 3</td>
<td>at least 4 players from levels 1, 2 or 3</td>
<td>and</td>
</tr>
<tr>
<td>or</td>
<td>or</td>
<td>and</td>
</tr>
<tr>
<td>at least 6 players from levels 1, 2, 3 or 4</td>
<td>at least 6 players from levels 1, 2, 3 or 4</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Example of Disjunctive and Conjunctive Threshold Secret Sharing
As shown, in our sequential secret sharing, a master secret along with \( \ell - 1 \) related secrets are shared among the players whereas, in disjunctive/conjunctive secret sharing, only one secret is shared. Furthermore, in our scheme, although the players in the initial level have the required authority to recover the master secret, they cannot do that without the sequential cooperation of the players from all levels. On the other hand, in disjunctive/conjunctive secret sharing, cooperation of the players from all levels may not be required. As a realization of our hierarchical access structure, we could imagine the president and vice president as the set \( P_1 \), ministers as the set \( P_2 \), and senators as the set \( P_3 \) accordingly. The president and vice president can recover the master secret (to trigger a secret action) only if they have the confirmations of ministers and senators. On the other hand, even by having those confirmations, the final decision (recovering the master secret \( \alpha_1 \) to trigger the intended action) is made by the president and vice president, i.e., distribution of the authority all over the hierarchical access structure. As we stated earlier, this access structure cannot be modeled by the existing hierarchical secret sharing schemes \([11, 12]\).

5 Concluding Remarks

In this article, we proposed a new hierarchical secret sharing scheme in which multiple secrets are shared among subsets of players with different levels of authority. In this protocol, reconstruction of the master key by the highest ranked players is subject to the cooperation of the players in the lower levels. On the other hand, even by having the secrets of the lower levels, the master key can only be recovered by the highest ranked players. We believe that SQS can be utilized in various cryptographic constructions.

References