Vector Space Secret Sharing Scheme

Mustafa Atici

Western Kentucky University
Department of Mathematics and Computer Science

52 MIGHTY Conference, Indiana State University, Terre Haute
IN. April 27-28, 2012
1. Security in cryptography is based on the secret key $K$.

2. In private-key cryptography, some time it is not secure to give secret key to an individual (participant).

3. Therefore secret sharing scheme was introduced to share secret key $K$ among authorized group of participants.
Secret sharing scheme works as follows: Let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be set of all participants.

**STEP 1:** Determine authorized group

**STEP 2:** Secure and public information are given to all participants for secret key $K$.

**STEP 3:** When authorized group of participants pool their share, then they will recover the secret key $K$.

**STEP 4:** If one or more participants are missing from the group, then remaining members of the authorized group cannot determine the secret key $K$. 
Example: Time magazine (May 4, 1992)

Russian nuclear ignition key

\[ \mathcal{P} = \{ \text{Boris Yeltsin, Yevgeni Shaposhnikov, Defence Ministry} \} \]

Authorized group \( B \subset \mathcal{P} \) such that \( |B| = 2 \).
Some of the well-known secret sharing schemes are:

1) The Shamir Threshold Scheme (also Blakley)

2) The Monotone Circuit Construction

3) Brickell Vector Space Construction
Let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be set of participants and $\Gamma = \{B_1, B_2, ..., B_k\}$ be an access structure on $\mathcal{P}$.

Let $p$ be large enough prime number and $d \geq 2$ be an integer number.

Suppose there exist a function $\phi: \mathcal{P} \rightarrow (\mathbb{Z}_p)^d$ with the following property:

$$(1, 0, ..., 0) = \langle \phi(P_i) : P_i \in B \rangle \iff B \in \Gamma = \{B_1, ..., B_k\}.$$  \hspace{1cm} (1)
Algorithm I: Vector Space Sharing Scheme (Due to Brickell)

**Input:** access structure $\Gamma$ and $\phi$ function satisfying (1)

**Initial Phase:**
1) for $1 \leq i \leq n$
2) $D$ gives public share $\phi(P_i) \in (\mathbb{Z}_p)^d$ to $P_i$

**Share Computation:**
3) $D$ chooses secret key $K \in \mathbb{Z}_p$
4) $D$ secretly chooses $a_2, a_3, ..., a_d \in \mathbb{Z}_p$ and forms vector
   $$a = (K, a_2, a_3, ..., a_d)$$
5) for $i = 1$ to $n$
6) $D$ computes $y_i = a.\phi(P_i)$
7) $D$ gives secret share $y_i$ to $P_i$
Example: Let $\mathcal{P} = \{P_1, P_2, P_3, P_4\}$ be set of participants and $\Gamma = \{B_1, B_2\} = \\{\{P_1, P_2, P_3\}, \{P_1, P_4\}\}$ be access structure. By trial and error we can find the following $\phi$ function, where $d = 3, p \geq 3$:

$\phi(P_1) = (0, 1, 0)$
$\phi(P_2) = (1, 0, 1)$
$\phi(P_3) = (0, 1, -1)$
$\phi(P_4) = (1, 1, 0)$

$(1, 0, 0) = \phi(P_2) - \phi(P_1) + \phi(P_3), \text{ where } B_1 = \{P_1, P_2, P_3\} \in \Gamma$

$(1, 0, 0) = \phi(P_4) - \phi(P_1), \text{ where } B_2 = \{P_1, P_4\} \in \Gamma$

No other subset of $\mathcal{P}$ which does not contain $B_1$ or $B_2$ cannot create $(1, 0, 0)$
We will represent $\phi$ as a matrix:

$$
\phi = \begin{pmatrix}
0 & 1 & 0 \\
1 & 0 & 1 \\
0 & 1 & -1 \\
1 & 1 & 0
\end{pmatrix}
$$

**Algorithm 1** is very efficient algorithm but requirement of existence of function $\phi$ is the only drawback.

There is no known efficient algorithm to construct such function $\phi$ for any given access structure $\Gamma$.

Stinson indicated in his book that trail and error (brute force search) is the only way to find it.

For large parameters $n, p, d$ exhausted search is time consuming.
Even if construction of such function $\phi$ is not very easy for every access structure

There is very elegant algorithm to construct a $\phi$ function for one particular access structure.

Let $G = (V, E)$ be a complete multipartite graph

Then define participant set $\mathcal{P} = V$ and access structure $\Gamma = E$

Construction of $\phi$ function for the vector space secret sharing is very easy (based on theorem in Stinson)
**Example:** Complete bipartite graph $G = (V, E)$

$V = \{P_1, P_2, P_3, P_4, P_5\}$ and

$E = \{\{P_1, P_3\}, \{P_1, P_4\}, \{P_1, P_5\}, \{P_2, P_3\}, \{P_2, P_4\}, \{P_2, P_5\}\}$

$\mathcal{P} = V$, $\Gamma = E$, and $V(G) = V_1 \cup V_2 = \{P_1, P_2\} \cup \{P_3, P_4, P_5\}$.

Pick two $x_1 = 1$, $x_2 = 2$, of $\mathbb{Z}_p^2$, where $p \geq 2$ and function as follows:

\[
\begin{array}{c|c}
\hline
x_1 & 1 \\
\hline
x_1 & 1 \\
\hline
x_2 & 1 \\
\hline
x_2 & 1 \\
\hline
x_2 & 1 \\
\hline
\end{array}
= 
\begin{array}{c|c}
1 & 1 \\
\hline
1 & 1 \\
\hline
2 & 1 \\
\hline
2 & 1 \\
\hline
2 & 1 \\
\hline
\end{array}
\]
Algorithm II: Construction of $\phi$ for multipartite graph

Input: Complete multipartite graph $G = (\mathcal{P}, \Gamma)$

1) determine disjoint partitions of $V(G) = \bigcup_{i=1}^{k} V_i$
2) choose distinct $x_i \in \mathbb{Z}_p$ for $i = 1, 2, ..., k$, where $p \geq k$
3) for $j = 1$ to $\vert \mathcal{P} \vert$
4) if $P_j \in V_i$, for some $i$
5) define $\phi(P_j) = (x_i, 1)$
6) return $\phi$
Let $G = (V, E)$ a multipartite graph but not complete

$\mathcal{P} = V$ and $\Gamma = E$ such that

$\Gamma = \{B_1, B_2, ..., B_m\}$ has the following properties:

1) $B_i \cap B_j = \emptyset$ for all $i \neq j$

2) $|B_i| = k$ for $i = 1, 2, ..., m$
Example: $G = (V, E)$ with $V = \{1, 4\} \cup \{2, 5\} \cup \{3, 6\}$ and
$E = \{(1, 2), (1, 3), (2, 3), (4, 5), (4, 6), (5, 6)\}$

$\mathcal{P} = V = \{1, 2, 3, 4, 5, 6\}$ and $\Gamma = \{B_1, B_2\} = \{\{1, 2, 3\}, \{4, 5, 6\}\}$
$|B_i| = k = 3$ so $d = 2k - 1 = 6 - 1 = 5$, and let us take $p = 5$

First construct $A_1$ and $A_2$ for $B_1 = \{1, 2, 3\}$ and $B_2 = \{4, 5, 6\}$, respectively

\[
A_1 = \begin{bmatrix}
1 & 1 & 0 & 2 & 0 \\
0 & 1 & 1 & 2 & 2 \\
0 & 0 & 1 & 0 & 2 \\
\end{bmatrix} \quad A_2 = \begin{bmatrix}
1 & 1 & 0 & 3 & 0 \\
0 & 1 & 1 & 3 & 3 \\
0 & 0 & 1 & 0 & 3 \\
\end{bmatrix}
\]

Then $\phi$ is

\[
\phi = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} = \begin{bmatrix}
1 & 1 & 0 & 2 & 0 \\
0 & 1 & 1 & 2 & 2 \\
0 & 0 & 1 & 0 & 2 \\
1 & 1 & 0 & 3 & 0 \\
0 & 1 & 1 & 3 & 3 \\
0 & 0 & 1 & 0 & 3 \\
\end{bmatrix}
\]
Algorithm III: Construction of $\phi$

Input: $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$, $\Gamma = \{B_1, B_2, \ldots, B_m\}$, where $B_i \cap B_j = \emptyset$ for all $i \neq j$ and $|B_i| = k$

1) pick $x_i \in \mathbb{Z}_p$ such that $1 < x_1 < x_2 < \ldots < x_m$
2) for $s = 1$ to $m$
3) construct $A_s = (a_{ij})_{k \times 2^{k-1}}$ with all 0 entries
4) for $i = 1$ to $k$
5) $a_{ii} = 1$
6) for $i = 1$ to $k - 1$
7) $a_{i(i+1)} = 1$
8) for $i = 1$ to $k - 1$
9) $a_{i(k+i)} = x_s$
10) for $i = 2$ to $k$
11) $a_{i(k+i-1)} = x_s$
12) return $\phi = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix}$
Matrix $A_i$ constructed by **Algorithm III** will be like

<table>
<thead>
<tr>
<th>$1$</th>
<th>$2$</th>
<th>$3$</th>
<th>$4$</th>
<th>..</th>
<th>$k-1$</th>
<th>$k$</th>
<th>$k+1$</th>
<th>$k+2$</th>
<th>$k+3$</th>
<th>..</th>
<th>$2k-1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1$</td>
<td>$1$</td>
<td>$0$</td>
<td>$0$</td>
<td>..</td>
<td>$0$</td>
<td>$0$</td>
<td>$x_i$</td>
<td>$0$</td>
<td>$0$</td>
<td>..</td>
<td>$0$</td>
</tr>
<tr>
<td>$0$</td>
<td>$1$</td>
<td>$1$</td>
<td>$0$</td>
<td>..</td>
<td>$0$</td>
<td>$0$</td>
<td>$x_i$</td>
<td>$x_i$</td>
<td>$0$</td>
<td>..</td>
<td>$0$</td>
</tr>
<tr>
<td>$0$</td>
<td>$0$</td>
<td>$1$</td>
<td>$1$</td>
<td>..</td>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>$x_i$</td>
<td>$x_i$</td>
<td>..</td>
<td>$0$</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>..</td>
<td>$1$</td>
<td>$1$</td>
<td>$0$</td>
<td>$0$</td>
<td>$..$</td>
<td>$x_i$</td>
<td>$x_i$</td>
</tr>
<tr>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>$0$</td>
<td>..</td>
<td>$0$</td>
<td>$1$</td>
<td>$0$</td>
<td>$0$</td>
<td>$..$</td>
<td>$0$</td>
<td>$x_i$</td>
</tr>
</tbody>
</table>
Properties of block $A_i$:
1. The first column has unique 1.
2. Columns 2 through $k$ have exactly two 1’s.
3. Columns $k + 1$ through $2k - 1$ have exactly two $x_i$’s.

**Lemma**

Let $B_i = \{P_{i1}, P_{i2}, ..., P_{ik}\}$ be an authorized set. Assume $A_i$ is created by **Algorithm III** for $B_i$. Then $(1, 0, 0, ..., 0)$ can be written as linear combination of shares, i.e. rows of $A_i$, of $B_i$ but if one or more rows of $A_i$ is missing, then $(1, 0, 0, ..., 0)$ cannot be written as linear combination of remaining rows of $A_i$. 
Proof.

Let $a_j$ be $j$th row of $A_i$. Then 

$(1, 0, 0, ..., 0) = (a_1 + a_3 + ...) - (a_2 + a_4 + ...) \text{ by properties of } A_i$

Now let $C = \{P_{i_{j_1}}, P_{i_{j_2}}, ..., P_{i_{j_l}}\} \subset B_i$. Without loose of generality we can assume that $i_{j_1} < i_{j_2} < ... < i_{j_l}$.

If $i_{j_i} \neq 1$, then it is obvious that $(1, 0, 0, ..., 0)$ cannot be linear combination of these rows. Hence $P_{i_{j_1}} = P_1$.

Since $C$ is unauthorized, there is at least one participant $P_{i_{j_s}}$ which is not in $C$. Let $s$ be the smallest index such that $P_{i_{j_s}} \notin C$

Let $a_1, a_2, ..., a_l \in \mathbb{Z}_p$

Suppose:

$(1, 0, 0, ..., 0) = a_1(1, 1, ..., x_i, 0, ..., 0) + \sum_{r=2}^{l} a_r \phi(P_{i_{j_r}}) \iff a_1 = 1, a_1 + a_2 = 0, ..., a_{s-2} + a_{s-1} = 0, a_{s-1} = 0, ... \text{ where } s \geq 2.$

Since $a_1 = 1$, then $a_2 = -1(p - 1 \text{ in } \mathbb{Z}_p)$ so on, hence we get $a_{s-1} = 1$ (or $-1$ based on even or odd $s$ value) contradiction with $a_{s-1} = 0.$
Theorem

Let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be set of participants. Access structure $\Gamma = \{B_1, B_2, ..., B_m\}$ is given where $B_i \cap B_j = \emptyset$ for all $i \neq j$ and $|B_i| = k$ for $i = 1, 2, ..., m$. Then the function $\phi$, which is constructed by Algorithm III, satisfies (1).
Proof.

Let $C = \{P_{j_1}, P_{j_2}, \ldots, P_{j_l}\} \subset \mathcal{P}$. If $C$ is an authorized set, then $B_i \subset C$ for some $i$. Hence by previous lemma we are done.

If $C$ is not authorized set, then we have the following cases:

Case 1: If $|C| = l < k$

Case 2: If $|C| = l = k$

Case 3: If $|C| = l > k$