PBIBD and its applications in Cryptology

Bimal Roy

Indian Statistical Institute
www.isical.ac.in/~bimal
In this talk ...

We will first describe the combinatorial framework of PBIBD

And then proceed to show its applications in Cryptology

1. **Key Predistribution** in Wireless Sensor Networks
2. **Traitor Tracing** in schemes with restricted access
3. **Secret Sharing** schemes using Visual Cryptography
Partially Balanced Incomplete Block Design (PBIBD)
Combinatorial Designs

A set system or design is a pair \((X, A)\), where
- \(X\) is the main set of elements
- \(A\) is a set of subsets of \(X\), called blocks

**Balanced Incomplete Block Design**

\(BIBD(v, b, r, k; \lambda)\) is a design which satisfy
- \(|X| = v\) and \(|A| = b\)
- Each block in \(A\) contains exactly \(k\) elements
- Each element in \(X\) occurs in \(r\) blocks
- Each pair of elements in \(X\) occurs in exactly \(\lambda\) blocks

Example: \(BIBD(7, 7, 3, 3; 1)\) on set \(X = \{0, 1, 2, 3, 4, 5, 6\}\)
\(A = \{ (1, 2, 4), (2, 3, 5), (3, 4, 6), (4, 5, 0), (5, 6, 1), (6, 0, 2), (0, 1, 3) \}\)
PBIBD: Partially Balanced Incomplete Block Design

\( PB[k; \lambda_1, \lambda_2, \ldots, \lambda_m; v] \) is a design such that

- There are \( b \) blocks, each of size \( k \), on a \( v \)-set \( X \)
- It is an association scheme with \( m \) associate classes
- Each element of \( X \) has exactly \( n_i \) number of \( i \)-th associates
- Two \( i \)-th associate elements occur together in \( \lambda_i \) blocks

Example: \( PB[3; 2, 2, 1; 6] \)

\[ X = \{1, 2, 3, 4, 5, 6\} \]
\[ v = 6, \ b = 8, \ r = 4, \ k = 3 \]
\[ A = \{(1, 2, 4), (1, 3, 4), (1, 2, 5), (1, 3, 6), (2, 3, 5), (2, 3, 6), (4, 5, 6), (4, 5, 6)\} \]
PBIBD: Another example

2-associate class PBIBD

1-st associates : Same row or column
2-nd associates: Rest of the elements

1-st associate of 6 : 1, 5, 7, 3, 8, 10
2-nd associate of 6: 2, 4, 9

Block 1: (2, 3, 4, 5, 6, 7) Block 2: (1, 3, 4, 5, 8, 9)
Block 3: (1, 2, 4, 6, 8, 10) Block 4: (1, 2, 3, 7, 9, 10)
Block 5: (1, 2, 6, 7, 8, 9) Block 6: (1, 3, 5, 7, 8, 10)
Block 7: (1, 4, 5, 6, 9, 10) Block 8: (2, 3, 5, 6, 9, 10)
Block 9: (2, 4, 5, 7, 8, 10) Block 10: (3, 4, 6, 7, 8, 9)
Application of PBIBD in Key Predistribution
Key Predistribution

- Security of the WSN depends on efficient key distribution
- PKC and ECC are too computation intensive for WSNs
- Thus we need distribution of keys in nodes prior to deployment

Problem: Distribute node keys from key-pool \{0, 1, 2, 3, 4, 5, 6\}. 

![Diagram showing key distribution among nodes with connections labeled 2 and 3.]
Metrics to evaluate Key Predistribution schemes

General metrics:
- Scalability: Allow post-deployment increase in network size
- Efficiency: Time taken for communication between nodes
- Storage: Amount of memory required to store the keys
- Computation: No. of cycles needed for key agreement
- Communication: No. of messages sent for key agreement

Security metrics:
- Key Connectivity: The probability that two nodes share one/more keys should be high
- Resiliency: Even if a number of nodes are compromised and the keys contained are revealed, the whole network should not fail, i.e., only a part of the network should get affected
Resiliency - an example

\[ V(s) = \text{Fraction of nodes disconnected for } s \text{ nodes compromised} \]
\[ E(s) = \text{Fraction of links broken for } s \text{ nodes compromised} \]

\[ V(2) = \frac{1}{13} = 0.0769 \text{ and } E(2) = \frac{14 + 13 + 12}{105} = 0.371 \]
Mapping PBIBD to Key Predistribution

2-associate class PBIBD

1: (2, 3, 4, 5, 6, 7) 2: (1, 3, 4, 5, 8, 9)
3: (1, 2, 4, 6, 8, 10) 4: (1, 2, 3, 7, 9, 10)
5: (1, 2, 6, 7, 8, 9) 6: (1, 3, 5, 7, 8, 10)
7: (1, 4, 5, 6, 9, 10) 8: (2, 3, 5, 6, 9, 10)
9: (2, 4, 5, 7, 8, 10) 10: (3, 4, 6, 7, 8, 9)

In this situation, we have \( n = 5 \), and

- Number of sensor nodes = \( n(n - 1)/2 = 10 \)
- Number of keys in key-pool = \( n(n - 1)/2 = 10 \)
- Number of keys in each node = \( 2(n - 2) = 6 \)
- Number of keys common to any two nodes = 4 or \( (n - 2) = 3 \)
Advantages of the Design

1. Number of keys per node is $2(n - 2)$, i.e., just $O(\sqrt{N})$, when the size of the network is $N = n(n - 1)/2$.

2. Any two nodes can communicate directly as they have at least one key shared among them.

3. Resiliency is increased in general, as follows.
   3.1 When two nodes in a row (or column) are compromised, then exactly one node will be disconnected ($n > 5$).
   3.2 Any two nodes compromised in different rows (or columns) will not disconnect any other node.
   3.3 If more than $\lceil n/2 \rceil + 1$ nodes are compromised in total, then at least one node will be disconnected.
   3.4 Maximum number of nodes disconnected when $s$ nodes are compromised is $s(s - 1)/2$ (when they are in a row/column).
## Experimental Results

<table>
<thead>
<tr>
<th>$n$</th>
<th>Network size $N$</th>
<th>Number of keys $k$</th>
<th>Captured nodes $s$</th>
<th>Affected nodes $V(s)$</th>
<th>Affected links $E(s)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>435</td>
<td>56</td>
<td>10</td>
<td>0.0753</td>
<td>0.3500</td>
</tr>
<tr>
<td>40</td>
<td>780</td>
<td>76</td>
<td>10</td>
<td>0.0351</td>
<td>0.2510</td>
</tr>
<tr>
<td>50</td>
<td>1225</td>
<td>96</td>
<td>10</td>
<td>0.0156</td>
<td>0.1800</td>
</tr>
<tr>
<td>60</td>
<td>1770</td>
<td>116</td>
<td>10</td>
<td>0.0085</td>
<td>0.1314</td>
</tr>
<tr>
<td>70</td>
<td>2415</td>
<td>136</td>
<td>10</td>
<td>0.0058</td>
<td>0.0724</td>
</tr>
</tbody>
</table>

The values of $V(s)$ and $E(s)$ in the table are experimental data.

### Scope:

- Is it possible to reduce the number of keys, but still improve the resiliency of the network?
- How can we repeatedly apply the PBIBD schemes and increase the scalability of the network?
Application of PBIBD in Traitor Tracing
**Traitor Tracing**

**Situation:**
- Supplier distributes products for only authorized users to use.
- Malicious authorized users (traitors) create pirated copies and distribute them to unauthorized users.

**Goal of Traitor Tracing:**
- Prevent authorized users to produce unauthorized copies.
- Trace the source of piracy if unauthorized copies are created.
- Trace traitors without harming the innocent users.
Traitor Tracing - Setup

Setup: The distributor supplies each user $U_i$ the following:

- A set of $k$ personal keys denoted by $P(U_i)$.
- Enabling block to create session key $s$ using personal keys.
- The plaintext message encrypted using the session key $s$.

Example: Number of users = 4, and Key pool = \{000, 001, 010, 011, 100, 101\}.

\[
P(U_1) = \{000, 010, 100\} \quad P(U_2) = \{000, 011, 101\}
\]
\[
P(U_3) = \{001, 011, 100\} \quad P(U_4) = \{001, 010, 101\}
\]

Session key = 110. (obtained by binary addition of the keys modulo 2)

No other combination of keys can generate the same session key upon binary addition.

\[
\begin{align*}
\{000, 001, 010\} &\rightarrow 011, \quad \{000, 001, 011\} \rightarrow 010, \quad \{000, 001, 100\} \rightarrow 101, \\
\{000, 001, 101\} &\rightarrow 100, \quad \{000, 010, 011\} \rightarrow 001, \quad \{000, 010, 101\} \rightarrow 111, \\
\{000, 011, 100\} &\rightarrow 111, \quad \{001, 010, 100\} \rightarrow 111, \quad \{000, 100, 101\} \rightarrow 001, \\
\{001, 010, 011\} &\rightarrow 000, \quad \{001, 011, 101\} \rightarrow 111, \quad \{001, 100, 101\} \rightarrow 000, \\
\{010, 011, 100\} &\rightarrow 111, \quad \{010, 011, 101\} \rightarrow 100, \quad \{010, 100, 101\} \rightarrow 011, \\
\{011, 100, 101\} &\rightarrow 010.
\end{align*}
\]
**Traitror Tracing - Action**

**Piracy:** Some users pool in their keys to make another valid key.

Users $U_1, U_2, \ldots, U_c$ can collude and create a *pirate decoder* $F$.

$$F \subseteq \bigcup_{i=1}^{c} P(U_i) \text{ and } |F| = k.$$ 

**Tracing:**
- If less than a certain number of authorized users collude, the distributor can trace them using the key distribution scheme.
- If more than this number of traitors collude, the distributor can not trace them without the risk of harming innocent users.

**Problem:** Design such a key distribution scheme for $P(U_i)$. 
c-Traceability Scheme

Suppose there are \( b \) users \( U_i \), each having a share of \( k \) personal keys \( P(U_i) \). Let the size of the whole key pool be \( v \).

c-TS(\( v, b, k \)) is a c-traceability scheme if \textit{at least} one traitor can be identified when a coalition of \( c \) or less traitors collude.

c-FRTS(\( v, b, k \)) is a \textit{fully resilient c-traceability scheme} if \textit{all} the traitors can be identified when a coalition of \( c \) or less traitors collude.

**Problem:** Design c-TS(\( v, b, k \)) or c-FRTS(\( v, b, k \)) using PBIBD, such that it supports a large number of users \( b \), small number of personal keys \( k \), and large margin \( c \) for tracing traitors.
Example: 2-Traceability

There are 25 users, and each is assigned 6 keys.
The pirated set of keys is $F = \{0, 1, 2, 3, 6, 8\}$.

$P(B_1) = \{0, 1, 6, 18, 22, 29\}$, $P(B_2) = \{0, 2, 3, 8, 20, 24\}$,
$P(B_3) = \{1, 3, 4, 9, 21, 25\}$, $P(B_4) = \{2, 4, 5, 10, 22, 26\}$,
$P(B_5) = \{3, 5, 6, 11, 23, 27\}$, $P(B_6) = \{4, 6, 7, 12, 24, 28\}$,
$P(B_7) = \{5, 7, 8, 13, 25, 29\}$, $P(B_8) = \{0, 7, 9, 10, 15, 27\}$,
$P(B_9) = \{1, 8, 10, 11, 16, 28\}$, $P(B_{10}) = \{2, 9, 11, 12, 17, 29\}$,
$P(B_{11}) = \{0, 4, 11, 13, 14, 19\}$, $P(B_{12}) = \{1, 5, 12, 14, 15, 20\}$,
$P(B_{13}) = \{2, 6, 13, 15, 16, 21\}$, $P(B_{14}) = \{3, 7, 14, 16, 17, 22\}$,
$P(B_{15}) = \{4, 8, 15, 17, 18, 23\}$, $P(B_{16}) = \{5, 9, 16, 18, 19, 24\}$,
$P(B_{17}) = \{6, 10, 17, 19, 20, 25\}$, $P(B_{18}) = \{7, 11, 18, 20, 21, 26\}$,
$P(B_{19}) = \{8, 12, 19, 21, 22, 27\}$, $P(B_{20}) = \{9, 13, 20, 22, 23, 28\}$,
$P(B_{21}) = \{10, 14, 21, 23, 24, 29\}$, $P(B_{22}) = \{0, 12, 16, 23, 25, 26\}$,
$P(B_{23}) = \{1, 13, 17, 24, 26, 27\}$, $P(B_{24}) = \{2, 14, 18, 25, 27, 28\}$,
$P(B_{25}) = \{3, 15, 19, 26, 28, 29\}$.

The 2 traitors $B_1$ and $B_2$ are uniquely traced.
For 3 traitors: Confusion between $\{B_1, B_2, B_3\}$ and $\{B_1, B_2, B_{13}\}$.
Mapping PBIBD to Traitor Tracing

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>*</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>7</td>
<td>*</td>
<td>10</td>
</tr>
</tbody>
</table>
| 4 | 7 | 9 | 10| *

2-associate class PBIBD

1: (2, 3, 4, 5, 6, 7)  
2: (1, 3, 4, 5, 8, 9)  
3: (1, 2, 4, 6, 8, 10)  
4: (1, 2, 3, 7, 9, 10)  
5: (1, 2, 6, 7, 8, 9)  
6: (1, 3, 5, 7, 8, 10)  
7: (1, 4, 5, 6, 9, 10)  
8: (2, 3, 5, 6, 9, 10)  
9: (2, 4, 5, 7, 8, 10)  
10: (3, 4, 6, 7, 8, 9)

In this situation, we have $n = 5$, and

- Number of total users: $b = n(n - 1)/2 = 10$
- Number of keys for each user: $k = 2(n - 2) = 6$
- Number of keys in key-pool: $v = n(n - 1)(n - 2)/2 = 30$

Identifiable collusion limit in this scheme is $c = \sqrt{2(n - 2)} \approx 2.$
Our Result

A $\sqrt{2(n-2)} - FRTS(n(n-1)(n-2)/2, n(n-1)/2, 2(n-2))$ can be constructed from a $[2; 0, 1; n(n-1)/2]$-PBIBD, when $n \geq 5$.

Previous example was for a $2 - FRTS(30, 10, 6)$ scheme ($n = 5$).

Merit of the scheme:

- For a system with $N$ users, each user having a set of $O(\sqrt{N})$ keys, a collusion of at most $O(4\sqrt{N})$ traitors can be traced.

- That is, for a set of 10,000 users, each user having a set of 100 keys, a collusion of at most 10 traitors can be traced.

Scope: Improve bound of $c$ compared to $N$ (better than $O(4\sqrt{N})$).
Application of PBIBD in Secret Sharing
Secret Sharing in Visual Cryptography

Visual Cryptography: Naor and Shamir, 1994

- Secret sharing scheme with $n$ participants, 1 secret image
- Secret image to be split into $n$ shadow images called shares
- Certain qualified subsets of participants can recover the secret
- Other forbidden sets of participants have no information
Example: (2, 2) Visual Cryptography Scheme

Number of shares is $n = 2$, and 2 shares can recover the secret.

Shares for Black pixel

Construction of shares

\[
S^1 = \begin{bmatrix}
0 & 1 \\
1 & 0
\end{bmatrix}
\quad \text{and} \quad
S^0 = \begin{bmatrix}
0 & 1 \\
0 & 1
\end{bmatrix}
\]
Problem Statement

Construct a \((m, n)\) Visual Cryptography Scheme (VCS) such that

- There are \(n\) participants and 1 secret image
- Secret image to be split into \(n\) shadow images called shares
- Any \(m\)-subset of participants can recover the secret
- No \(t\)-subset of participants can recover the image if \(t < m\)

In particular, we will construct a \((2, n)\)-VCS in this talk.

Metric: Relative Contrast

If \((2, n)\)-VCS has basis matrices \(S^0\), \(S^1\) and pixel expansion \(m\), then relative contrast for participants in subset \(X\) is given by
\[
\alpha_X(m) = \frac{1}{m}(w(S_X^1) - w(S_X^0)).
\]
Mapping PBIBD to VCS

Suppose there exists an \((v, b, r, k, \lambda_1, \lambda_2)\)-PBIBD. It maps to a \((2, n)\)-VCS with \(n = v\), and pixel expansion \(m = b\).

Relative contrast in a subset \(X = \{\beta, \gamma\}\) of participants:
- If \(\beta, \gamma\) are 1-st associates, \(\alpha_X(m) = \frac{1}{m}(r - \lambda_1)\)
- If \(\beta, \gamma\) are 2-nd associates, \(\alpha_X(m) = \frac{1}{m}(r - \lambda_2)\)

Mapping:
1. Suppose \(N\) is the incidence matrix of the PBIBD.
2. Take share \(S^1 = N\), which has \(r\) number of 1’s in each row.
3. Construct share \(S^0\) with all identical rows, with \(r\) 1’s in each.
4. These shares \(S^0, S^1\) will make a \((2, n)\)-VCS with \(n = v\).
Example: PBIBD to VCS

Let us have a \((v = 6, b = 4, r = 2, k = 3, \lambda_1 = 0, \lambda_2 = 1)\)-PBIBD

- \(X = \{1, 2, 3, 4, 5, 6\}\) and
- \(A = \{(1, 2, 3), (1, 4, 5), (2, 4, 6), (3, 5, 6)\}\)

Construction of a \((2, 6)\)-VCS

\[
S^1 = N = \begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 \\
\end{bmatrix}
\]

and

\[
S^0 = \begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 \\
\end{bmatrix}
\]

Pixel expansion is clearly \(m = 4\), from the rows of the shares. Relative contrast is either \(\frac{1}{2}\) or \(\frac{1}{4}\).
Example: PBIBD to VCS

Visual outcome of (6, 4, 2, 3, 0, 1)-PBIBD to (2, 6)-VCS

Secret image:

One Share

Share 1:

Share 2:

Share 6:

Two Shares

Shares 1 & 6:

Shares 1 & 2:

Relative contrast is $\frac{1}{2}$ for 1 & 6 and $\frac{1}{4}$ for 1 & 2
Thank You