The Stanford InfoLab

Hector Garcia-Molina
Bob Mungamuru

Stanford University
InfoLab Research

- Information:
  - how to obtain information
  - how to manage it
  - how to exploit it
Hector’s Current Interests

- Information Privacy & Security
- Managing Bio-Diversity Information
- Entity Resolution
- Web Information
- Peer to Peer Systems
Information Privacy & Security

• How to build a SECURE database system?
  – good performance
  – usable
Information Privacy & Security

- How to build a SECURE database system?
  - good performance
  - usable

![Diagram showing database system architecture]

Client \( \rightarrow \text{Encrypt} \rightarrow \text{dbms} \)

Query \( Q \) \( \rightarrow \) Client-side Processor \( \rightarrow \) Answer

"All Data"
Information Privacy & Security

- Preservation, Performance, Functionality

![Diagram showing the relationship between preservation and privacy with easy and goal points marked.]
Key Configuration Management

- Safeguarding sensitive data
  - against data loss
  - against unauthorized access
Splitting

- **XOR**
  
  \[
  \begin{array}{c}
  010 \\
  110 \\
  101 \\
  001 \\
  \end{array}
  \]

- **Encryption**

  \[
  \begin{array}{c}
  \text{data} \\
  \text{key} \\
  \text{key} \\
  \text{ciphertext} \\
  \end{array}
  \]

- **DB + update logs**

  \[
  \begin{array}{c}
  \text{DB} \\
  \text{update log} \\
  \text{old snapshot} \\
  \end{array}
  \]
Example Configuration

data
Alice
Bob
Carol
Sharing Keys

![Diagram showing the sharing of keys between nodes labeled C, S, data1, Bob, and data2. The diagram illustrates the relationships and flows between these entities.]
“Problematic” Configuration
Configurations

Implementable

Proper

Simple

Read-Once
Checking a Configuration

\[ a = b = c \]
\[ (a = -1) \oplus (b = -1) \]
\[ (c = -2) \oplus (d = -2) \]

no satisfying assignment!

therefore, unimplementable
Checking Another Configuration

(a = -1) ⊕ (d = -1) ⊕ (e = -1) ⊕ (c = -1)
(a = -2) ⊕ (b = -2)
(b = -3) ⊕ (c = -3)

Satisfying Assignment:  a = -1, b = -2, c = -3
Checking Another Configuration

\[(a = -1) \oplus (d = -1) \oplus (e = -1) \oplus (c = -1)\]
\[(a = -2) \oplus (b = -2)\]
\[(b = -3) \oplus (c = -3)\]

Satisfying Assignment: \(a = -1, b = -2, c = -3\)
Proper Configurations

\[(e = -1) \oplus (f = -1) \oplus (d = -1) \]
\[a = e = b = f = c\]
Proper Configurations

Improper configuration, since $e$ and $f$ are forced to be equal.

\[ (e = -1) \oplus (f = -1) \oplus (d = -1) \]

\[ a = e = b = f = c \]

added constraint:

\[ e \neq f \neq d \]
Simple Configurations

- Simple configurations have special structure:
  - S-vertices have at least one unshared child
  - C-vertices have no shared children

- Theorem:

  All simple configurations are proper.
Simple Configurations

The configuration on the right is simple.
Read-Once Configurations
Comments

Logical transformations **do not** necessarily preserve properties

- \( d \ (b + (ac(a+c)) + ae) \) ... unimplementable

- \( d \ (a + b) \ (b + c + e) \) ... implementable

- \( d \ (b + ac + ae) \) ... simple
Comments