Auditing Information Leakage for Distance Metrics

Yikan Chen
David Evans

www.securecomputation.org
Motivating Scenario

How to determine whether it is safe to release the response?
From the Attacker’s View

\[ X, Q \in \{0, 1\}^{10} \]

\[ f(X, Q_i) \]

Hamming distance between X and Q

Smallest-Entropy Strategy

[Neuwirth, 1982]
Contribution

Target: class of distance functions

An efficient query auditor to estimate maximum information leakage

Technique for single bit information auditing
Strategy

Determine **# of secrets** consistent with all queries
  - For a special class of functions:
    Reduce to *0-1 integer programming*
  Too expensive to compute exhaustively
    - Use heuristics [find a meaningful bound]

**Assumption**
  - clients are authenticated by the server
  - clients cannot share information with each other
**Additively Decomposable Functions**

**Definition**

A function, \( f(A, B) \), where \( A, B \in \Sigma^N \), is *additively decomposable* if:

\[
\forall i \in [0, N], f(A, B) = f(A_{0:i}, B_{0:i}) + f(A_{i:N}, B_{i:N})
\]

- **Hamming distance**
  \[ |i \in N : A_i \neq B_i| \]

- **Squared Euclidean distance**
  \[ \sum (A_i - B_i)^2 \]

- **Manhattan distance**
  \[ \sum |A_i - B_i| \]

- **Edit distance**
  Ins, del and sub

- **Chebyshev distance**
  \[ \max(|A_i - B_i|) \]

- **Lee distance**
  \[ \min(|A_i - B_i|, k - |A_i - B_i|) \]
Influence Function

**Influence function** for any sequence $X'$: $g(\Delta i)$

$\Delta i$: changes made on $i$th bit of $X$

For *additively decomposable* functions, $g(\Delta i)$ for each bit is *independent* and for any consistent sequence over a set of queries:

$$\sum g(\Delta i) = 0$$

**Hamming distance:**

$$g(\Delta i) = \begin{cases} 
1, & X_i = Q_i \\
-1, & X_i \neq Q_i 
\end{cases}$$
Reduction for Hamming Distance

Convert a Hamming distance sequence leakage problem to:

\[ AK = 0 \]

where \( A_{ij} = \begin{cases} 1, & X_j = Q_{ij} \\ -1, & X_j \neq Q_{ij} \end{cases} \)

\[ K \in \{0, 1\}^N \]

Example:

\( X=1111 \quad Q_1=1100 \quad Q_2=1110 \)

\[ k_1 + k_2 - k_3 - k_4 = 0 \]
\[ k_1 + k_2 + k_3 - k_4 = 0 \]

find the number of possible 0-1 solutions for \( K \)
Computing Number of Solutions

Exact methods: exponential growth
Exhaustive Search
Gröbner Basis ([Bertsimas, 2000])

Lower bound is good enough!
Divide-and-merge algorithm
Divide-and-merge Algorithm

Find # of 0-1 solutions for the equation set: $AK=0$

Divide and exhaustive search
- Divide matrix A into small blocks
- Analyze each output exhaustively

Sort and select
- Sort output
- Select $r$ best combinations

Merge
- Merge adjacent blocks

$X, \{Q_i\}$
\[
X = 11111111\quad Q_1 = 01001000\quad Q_2 = 00111100
\]

\[
A = \begin{bmatrix}
-1 & 1 & -1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 1 & 1
\end{bmatrix}
\]

\[
k_1, k_2 \in \{0, 1\}
\]

Complexity

\[
A(K) = 0
\]

linear to \(N\)

linear to \(r^2\)
Evaluation

Implementation experiment settings
Matlab R2010b, 4 GB RAM 2.13 GHz CPU

Experiment settings
Averaging from 5 experiments
Scalability
Tightness
Real Application Performance (Iris Recognition)
Evaluation

See the paper for detailed results and performance on iris application.
Related Work

Estimating Information Leakage

– No auditor ([Goodrich, Oakland 2009])
– Weak auditor ([Wang, Wang et al., CCS 2009])

Differential Privacy

– Add noise to protect secret information ([Dwork, ICALP 2006])

Auditing Aggregate Queries

– Auditing SUM, MEAN or MAX queries over a set of private entries in a database ([Elshiekh, 2008], [Nabar et al., VLDB 2006], [Wang, Li et al., CCS 2009])
Summary

• Query auditor for the server
  – Fast
  – Performance adjustable
  – Single-bit leakage (see paper)

• Challenges
  – Non-additive decomposable functions
  – Secure two-party computation protocol
Thanks!

www.securecomputation.org
Single Bit Information Leakage

**Motivation**

One bit leakage can also be crucial:

SNP (single nucleotide polymorphism)

**Assumptions**

- The client knows nothing
- Any single bit information is leaked only when this bit is 100% determined
Single Bit Information Leakage

Straightforward idea

$$\text{AK} = 0$$

Bit fully determined = corresponding $$k_i$$ can only be 1

Check if there is a non-zero solution when $$k_i = 0$$

Make it quicker......

Build pre-computed libraries containing all possible combinations

Libraries can be pre-computed since they are not related to sequence length (N), X and Q

Without libraries: $$C_{1000-1}^{6-1} = 8.2 \times 10^{12}$$ checks

With libraries: 9632 checks