Computing with private data: when cryptography meets signal processing

Mauro Barni
University of Siena
What: the SSP – SPED paradigm

I do not trust him but I need his help

Interactive SSP protocol

secured data 0110011001 1001001010

result

secured data 0111010001 1100101010

result

private data

private data

MMSP 2011, 18 October 2011, Hangzhou, China

M. Barni, Dept. Information Engineering, Univ. of Siena, VIPP group
Why? Network and web security

• Privacy-Preserving Intrusion Detection
  – Analysis of private log files, traffic monitoring

• Abuse detection in social networks
  – Chat rooms or messaging services ensure user anonymity
  – Users should be traceable if they severely violate the terms of usage.
  – To limit traceability to severe instances, abuse detection could be carried out on encrypted data and anonymity revoked only in case of violation

• Oblivious Web Ratings
  – The popularity of web pages is assessed by a third party analyzing the encrypted log files of a web server
Why ? Profiling / recommendation services

- Targeted Recommendations
  - Personalized recommendations have high business value but open a privacy-problem
  - Problems can be avoided by methods that analyze the relevant user habits in the encrypted domain.

- Data Mining for Marketing
  - Knowledge of preferences of class of users is invaluable information in marketing.
  - Performing classifications in the encrypted domain can prevent privacy concerns
Why ? Access control and biometrics

• Private Access control via encrypted queries
  – Access to a service is granted upon inspection of a biometric template (BT)
  – The BT is encrypted so to avoid revealing the biometry and the identity of the user accessing the service

• Biometric control in public places (airport …)
  – An encrypted BT is used to look for criminals or terrorists in public locations
  – Only if a match is found the identity is revealed thus avoiding tracing honest citizens
Why ? Biomedical data processing

• Storing biomedical data on remote servers
  – Medical sensitive data/signals are stored under encryption
  – Additional services are provided by processing the encrypted data
  – Google-health

• Privacy-preserving remote services
  – a remote diagnosis services analyses encrypted data and provides recommendations without violating the users’ privacy

• Analysis of bio-signals
  – by processing encrypted bio-signals the analysis reveals only the information it is intended for
Why? Consumer electronics - entertainment

- Privacy preserving search for content
  - again a case of searching with encrypted queries
- DRM
  - the identity of the buyer is embedded in the purchased media without disclosing it to the seller
- Transcoding
  - transcoding of (encrypted) multimedia data at non-trusted nodes

Transcoding without decryption key
How? The tools

- Homomorphic encryption
- Blinding / obfuscation
- Oblivious transfer
- Garbled circuits
- Hybrid approach
The *homomorphic* paradigm

An algebraic operation on the plain messages is mapped into a (possibly different) algebraic operation on the encrypted messages

\[ a \cdot b = D_{pr} [E_{pub}(a) \circ E_{pub}(b)] \]

if

\[
\begin{align*}
\circ &= + \\
O &= \times
\end{align*}
\]

\[ \Rightarrow a + b = D_{pr} [E_{pub}(a) \times E_{pub}(b)] \quad \text{additive HE} \]

\[ Ka = D_{pr} [E_{pub}(a) \times E_{pub}(a) \ldots E_{pub}(a)] = D_{pr} [E_{pub}(a)^K] \]

if

\[
\begin{align*}
\circ &= \times \\
O &= \times
\end{align*}
\]

\[ \Rightarrow a \times b = D_{pr} [E_{pub}(a) \times E_{pub}(b)] \quad \text{multiplicative HE} \]
The *homomorphic* paradigm

With additive HE a number of interesting operators can be applied to signals:

Component-wise encryption ⇒ $E[(a_1, a_2 \ldots a_n)] = (E[a_1], E[a_1] \ldots E[a_n])$

Scalar product (known vector) : $\langle a, b \rangle = \sum_{i=1}^{n} a_i b_i \Rightarrow E[\langle a, b \rangle] = \prod_{i=1}^{n} E[a_i]^{b_i}$

FIR filtering : $a_n = \sum_{k=1}^{L} a_{n-k} h_k \Rightarrow E[a_n] = \prod_{k=1}^{L} E[a_{n-k}]^{h_k}$

Linear transforms: $X_k = \sum_{i=1}^{n} a_{k,i} x_i \Rightarrow E[X_k] = \prod_{i=1}^{L} E[x_i]^{a_{k,i}}$
Pailler’s cryptosystem

Composite residuosity problem

Given $c, \gamma$ and $n$ find $m$ such that

$$c = \gamma^m r^n \mod n^2$$

for some $r$

Randomization

Plain message

Public key

Additive Homomorphism follows from properties of exponentials

Security $\to c$ at least 2048 bits
Non-linear functions and full HE

\[
\begin{aligned}
\text{if } \otimes \text{ and } \oplus \text{: } & \left\{ 
\begin{array}{l}
a + b = D[E(a) \oplus E(b)] \\
a \times b = D[E(a) \otimes E(b)]
\end{array}
\right. \\
\text{full HE}
\end{aligned}
\]

Kind of holy Graal in cryptography
recent breakthrough by Gentry

\ldots

still impractical

\ldots

SSP designers can rely on additive HE only
Non-linear functions through blinding

- Assume an additive cryptosystem is available.
- Bob needs to apply a non-linear function $f(\cdot)$ to $x$ available to him in encrypted format.

Alice

\[
g(y) \quad \rightarrow \quad E[g(y)]
\]

Bob

- Generates $a$ and $b$ randomly
- $E[y] = E[ax+b]:$ blinding
- Obtains $E[f(x)]$ from $E[g(y)]$

- Works if $f(x) = \alpha(a,b) g(y) + \beta(a,b) x + \gamma(a,b)$
- … and is difficult (impossible) to recover $x$ from $y$
Example: squaring an encrypted number

Alice

\[ y = D[E[y]] \]
\[ g(y) = y^2 = x^2 + b^2 + 2xb \]
\[ E[g(y)] \]

Bob

\[ E[x] \]
\[ E[y] = E[x + b] = E[x]E[b] \]
\[ E[y] \]
\[ E[g(y)] \]
\[ E[x^2] = E[g(y) - b^2 - 2bx] \]
\[ = E[g(y)]E[-b^2]E[x]^{-2b} \]
An alternative approach: OT + GC

- Private computation of any function expressed as a logic (non recursive) circuit
- Symmetric cryptography
- Inputs at the bit level
- Thought to be impractical until 4-5 years ago
  - now: about 50,000 gates per second
Oblivious transfer (OT)

- 1-out-n, parallel version
- Base for a large number of SSP protocols
A garbled gate (Yao’s approach)

- For each wire
  - Generate 2 random secrets for each input and output bit: $X^0, X^1, Y^0, Y^1, Z^0, Z^1$
  - Encrypt the output secret with the concatenation of corresponding input secrets
  - Put the encrypted output secrets in a table with randomly permuted rows
- The encryption scheme must have
  - Elusive and efficiently verifiable range
- Given the correct input secrets decrypt all possible outputs and obtain the output secret

Garbled table:

- $\text{Enc}_{X_0, Y_0}[Z^{(0,0)} = Z^0]$
- $\text{Enc}_{X_0, Y_1}[Z^{(0,1)} = Z^0]$
- $\text{Enc}_{X_1, Y_0}[Z^{(1,0)} = Z^0]$
- $\text{Enc}_{X_1, Y_1}[Z^{(1,1)} = Z^1]$
A garbled circuit

Client

input bits

Oblivious transfer ()

EvalGC()

Result

Server

Circuit

input secrets

CreateGC()
Hybrid solution

• HE:
  – pros: no interaction for linear operations, no need of bit-wise representation
  – cons: difficulty with non-linear operations, asymmetric encryption, key-length

• GC:
  – pros: universal computing, symmetric crypto
  – cons: bit-wise representation, size of logic circuit may grow more than linearly

• Most recent trend: hybrid solution
  – combine GC and HE
  – transcoding overhead
Role of SP designers

- Optimize algorithms in terms of
  - bit length and number of variables
    - Representation accuracy has a strong impact on
      - Accuracy of results
      - Complexity of the protocol
    - adopted tools in view of available SSP primitives
      - Simple operations in the plain domain may be very complex when applied on encrypted signals
SSP at work: biometric-based authentication

- Criminal tracking with privacy protection for citizens: if you are not a criminal the system will not track you
- Privacy preserving access control: I know you can access a service but don’t know who you are
SSP at work: biometric-based authentication

Client

Feature extraction

$E[t] = E[t_1] \ldots E[t_n]$  

Distance computation

$E[d_1] \ldots E[d_m]$  

Comparison with threshold

$T$

YES / NO

Server

$X_1, X_2, X_3, \ldots, X_m$

YES / NO
SP choices

• Choice of feature set and distance function that ease an SSP implementation
• Classical approaches based on minutiae are difficult to implement
• Our choice:
  – **Fingercode**
    • Energy contained in different areas of the fingerprint image in different frequency bands
    • **Minimize number of features**
    • **Representation accuracy**
  – **Euclidean distance**
**Distance computation: classical approach**

- The Squared Euclidean distance between an encrypted and a known vector is easy to compute by relying on HE.

\[
d(t, x)^2 = \sum_{i=1}^{n} (t_i - x_i)^2 = \sum_{i=1}^{n} t_i^2 + \sum_{i=1}^{n} x_i^2 - 2 \sum_{i=1}^{n} t_i x_i
\]

- Computed by the client
- Computed by the server
- Computed by the server via HE

\[
E[d^2] = E\left[\sum_{i=1}^{n} t_i^2\right] E\left[\sum_{i=1}^{n} x_i^2\right] \prod_{i=1}^{n} E[x_i]^{-2t_i}
\]
Threshold comparison

• Comparison is by far easier through GC’s

• Hybrid solution
  • distances computed via HE are converted into (secret) bits
  • Pass from HE to GC representation
  • Run the GC
Comparison circuit
Performance

• Set-up
  – Java-based implementation
  – PC-platform (clock 2GHz, RAM 2GByte)
  – Pailler + GC
  – 96 features, 4 bits per feature
• Fingercode, Hybrid:
  – time: < 0.1 sec for template
  – bandwidth: 100Kbit per template
• Similar performance with
  – face recognition, iris recognition
Remote classification of ECG signals

Client

Server
- Remote diagnosis service
- Alert service based on medical data repository

ECG

Features

Classifier

Response
The SSP protocol

Client

ECG

Preprocessing

Feature extraction (AR model)

• The client protects
  • ECG data (features)

Server

• Server protects
  • NN weights

neural net classification

Result

NN parameters

Preprocessing

Feature extraction (AR model)

neural net classification

Result

MMSP 2011, 18 October 2011, Hangzhou, China

M. Barni, Dept. Information Engineering, Univ. of Siena, VIPP group
Secure computing with NN

Pailler Homomorphic encryption

GC approach
Representation accuracy

Representing the input and intermediate values with 12 bits is enough to achieve the best classification accuracy.

7-8 bits are enough to represent the output of the NN (before taking the max).
Performance

• Set-up
  – Java-based implementation
  – PC-platform (clock 2GHz, RAM 2GByte)
  – Pailler + GC

• Communication complexity (per heart beat)
  – 80 Kbit (for short term security)
  – 120 Kbit (for long-term security)

• Running time
  – 3-4 seconds per heart beat
  – Almost real-time
A roadmap for future research

• Efficiency, efficiency, efficiency
  – Crypto-level
    • more efficient primitives: fully homomorphic encryption
  – SP level
    • SSP-oriented algorithm design
    • ad-hoc security measures
    • New routes to SSP (compressive sampling ?)

• Security against malicious adversaries
  – recent breakthrough: GC construction against malicious adversary at 7000 gates/s

• System-level solutions, new applications

• Multi-disciplinary training, awareness raising
Thank you for your attention