Homomorphic Encryptions and Private Set Operations

Jung Hee Cheon

ISaC & Dept. of Math. Sciences
Seoul National University

July 6, 2012 (NIMS)

Contents

- Privacy Homomorphism
- Asymetric Homomorphic Encryption
- Set Operations on Encrypted Data
- Practical Applications

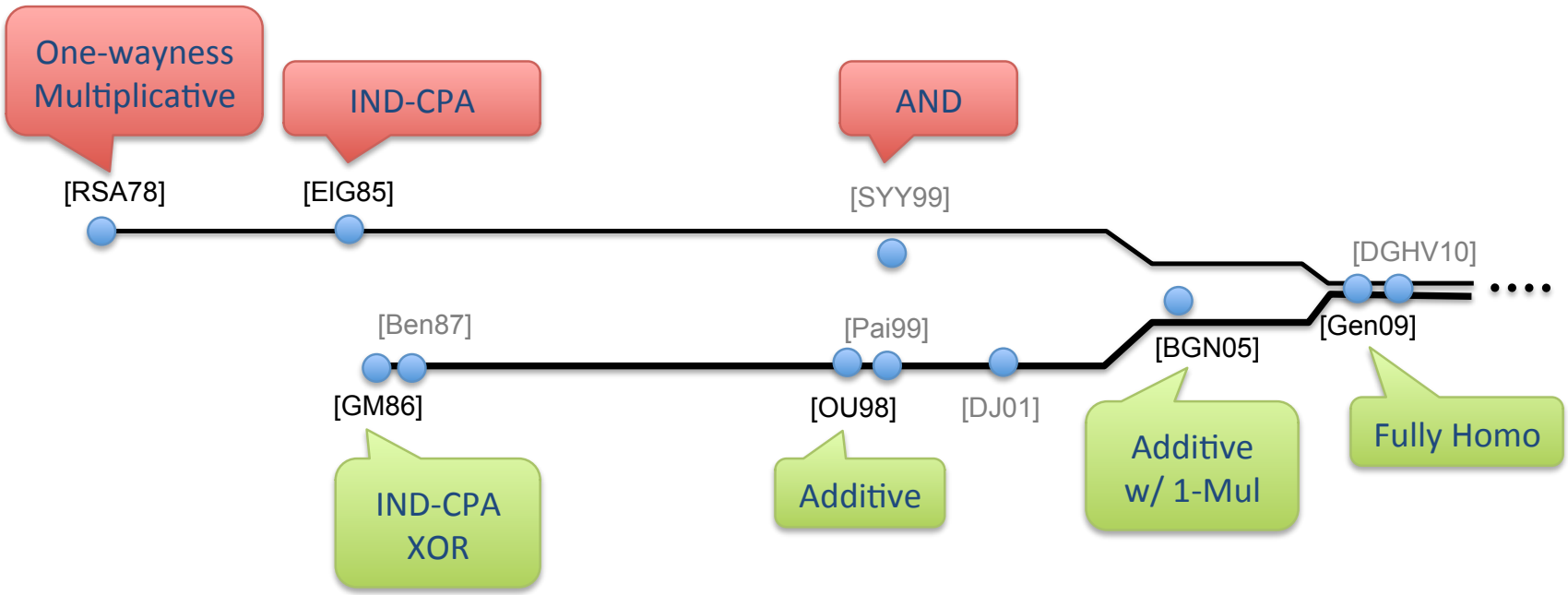
What We Need

- Computing on Encrypted Data
- Homomorphic Encryptions ...

Fully Homomorphic Encryption

- Overkill or not?
- Is there either jump or overlook?

Brief History



Homomorphic Schemes

- Privacy Homomorphism: [RAD78]
- XOR: [GM86]
- AND: [SYY99]
- Addition: [Ben87], [OU98], [NS98],[Pai99], [DJ01] and so on...
- Multiplication: [RSA], [EIG85]
- Addition + d-Multiplication: [BGN05],[MGH10]
- Full Operations: [Gen09], [DGHV10], and so on...

Privacy Homomorphism

- "encryption functions which permit encrypted data to be operated on without preliminary decryption of the operands, for many sets of interesting operations"
- Data can be processed by an external computing facility (Database as a service)
- Suppose the loan company stores their data in a outsourced database [RAD78]
  - What is the size of the average loan outstanding?
  - How much income from loan payments is expected next month?
  - How many loans over $\$5,000$ have been granted?

PH by Rivest et al.

- Private key: two large primes p and q
- Public key: $n = pq$
- Encryption: $E_n(a) = (a \bmod p, a \bmod q)$ for $a \in \mathbb{Z}_n$
- Decryption:
  $D_{p,q}(d_1, d_2) = d_1 q(q^{-1} \bmod p) + d_2 p(p^{-1} \bmod q) \bmod n$
- Support modular addition, subtraction, and multiplication
- Very efficient!

Security of RAD-PH

- Secure under no message attack
- What if we have a pair of pt and ct? (Brickell and Yacobi)
- Introducing random factors...
  - Take $N = \prod p_i$ and $E : Z_N \rightarrow Z_N$.
  - $E(m) = c$, where $c \equiv (m + e_i) \bmod p_i$ for small e_i.
- A homomorphic encryption is insecure under the chosen ciphertext attack

Research Direction

- Construct an efficient additive homomorphic *symmetric* encryption
- Construct a secure ring homomorphic *symmetric* encryption

XOR Homomorphic Encryption

Goldwasser-Micali Scheme [GM86]

- Basic Idea: Map 0 to a random QR and 1 to a random QNR
- on the QR assumption & IND-CPA security
- Work flow
  - KeyGen(1^λ): $pk = (n = pq, \alpha)$ and $sk = (p, q)$
    – $\alpha \notin \mathrm{QR}_n$ s.t $\left(\frac{\alpha}{p}\right) = -1$ and $\left(\frac{\alpha}{q}\right) = -1$.
  - Enc(pk, m): To encrypt a message $m = \{0, 1\}$
    – $c = \alpha^m r^2 \pmod n$ for a random $r \xleftarrow{\$} \mathbb{Z}_n^\times$
  - Dec(sk, c):
    – $m = 0$ if $c \in \mathrm{QR}$; otherwise $m = 1$
- Homomorphism
  – $c_1 c_2 = \alpha^{m_1} r_1^2 \cdot \alpha^{m_2} r_2^2 = \alpha^{(m_1 + m_2 \mod 2)} (r_1 r_2)^2$

AND Homomorphic Encryptions

The Sander-Young-Yung Encryption [SYY99]

- Basic Idea: Special Encoding/Decoding + [GM86]
  - Encode $\{0,1\} \to \{0,1\}^\ell$, $0 \mapsto r \in_R \{0,1\}^\ell$ and $1 \mapsto 0^\ell$
  - Encrypt each bit in the ℓ-bit vector using HE $\mathcal{E}$ of [GM86]
  - Decrypt and decode (msg is 1 iff output is a zero vector)
- Homomorphism
  - $\mathrm{Enc}(m_1) = (\mathcal{E}(a_1), \ldots \mathcal{E}(a_\ell))$ and $\mathrm{Enc}(m_2) = (\mathcal{E}(b_1), \ldots \mathcal{E}(b_\ell))$ where $a_i, b_i \in \{0,1\}$
  - $\mathrm{Enc}(m_1) \wedge \mathrm{Enc}(m_2) = (\mathcal{E}(a_1) \cdot \mathcal{E}(b_1), \ldots, \mathcal{E}(a_\ell) \cdot \mathcal{E}(b_\ell))$
  - Note random $\oplus$ random $=$ random, random $\oplus\, 0^\ell =$ random, $0^\ell \oplus 0^\ell = 0^\ell$
- Message expansion: ℓn

Research Direction

- Construct an homomorphic encryption w.r.t an operation which is functionally complete
- NOR, NAND, ...
- Asymmetric or symmetric

Homomorphic in mod M multiplication

- RSA encryption
  - on the RSA assumption
  - One-wayness security
  - M is a secret composite number (i.e. $\phi(N)$)
- ElGamal encryption
  - on the DDH assumption
  - IND-CPA security
  - M is a public prime
  - Variants based on different hardness assumptions

Additively Homomorphic in $\mathbb{Z}_N$

The Benaloh Encryption [Ben87]

- on the ℓ-th Residuosity Problem
- Basic idea: Remove random part by unknown order and find exhaustively a correct message in $\mathbb{Z}_\ell$
- Message expansion $= \frac{n}{\ell}$ (vs [GM86]: n)

The Naccache-Stern Encryption [NS98]

- on the Factoring n Problem
- Basic idea: Messages are represented by small primes and reconstructed using CRT based on [Ben87]
- Message expansion ≥ 4

Example of NS98

- Outline
  - Message space $\mathcal{M} = \mathbb{Z}_M$ where $M = 2^a 3^b$
  - Decryption utilizes Pohlig-Hellman for solving DL
  - In [OU98] M is unknown prime and in [Pai99] $M = n$ is a hard-to-factor composite
- Scheme
  - KeyGen: $pk = (n = pq, g, h, a, b)$ and $sk = (p, q)$ s.t $p - 1 = 2^a p', q - 1 = 3^b q'$ for some primes p', q', and g is a random generator of order $\lambda(n) = Mp'q'$ and h a random element of order $p'q'$
  - Enc(pk, m): To encrypt $m \in \mathbb{Z}_M$
    – $c = g^m h^r \mod n$ for some random $r \in [1, p'q']$
  - Dec(sk, c):
    – Compute $c^{p'q'} = (g^{p'q'})^m$ and solve DL using Pohlig-Hellman

The Okamoto-Uchiyama Encryption [OU98]

- on the Factoring $n = p^2 q$ Problem/Sylow p-subgroup Problem
- Basic idea:
  - H: the unique subgroup of order p of $\mathbb{Z}_{p^2}^{\times} \subset \mathbb{Z}_n^{\times}$
  - Solve the DLP on $H \subset \mathbb{Z}_n^{\times}$ easily.
- Logarithm
  - $(1+p)^x \equiv 1 + xp \bmod p^2$ and $(1+p)^p \equiv 1 \bmod p^2$
  - $\zeta := 1 + p$ is an order-p element of $\mathbb{Z}_{p^2}^*$.
  - Given $h \in H$, exist $x \in \mathbb{Z}_p$ s.t. $h = \zeta^x$. Then $x := (h-1)/p \bmod p$.
  - Define an isom $L = \log_\zeta : H \to \mathbb{Z}_p$, $h \mapsto (h-1)/p$
  - Given $g, h \in H$, $\log_g h = (\log_\zeta h)/(\log_\zeta g) = L(h)/L(g) = (h-1)/(g-1)$

The Okamoto-Uchiyama Encryption [OU98]

- Work flow
  - KeyGen: $pk = (n = p^2q, g, h, \ell)$ and $sk = (p, q)$ where p, q: ℓ-bit primes, $g \in_R \mathbb{Z}_n^\times$ s.t $g^{\varphi(p^2)} = 1 \mod p^2$ and $g^{p-1} \neq \mod p^2$, and $h = g^n \mod n$
  - Enc(pk, m): to encrypt $m \in \mathbb{Z}_p$
    – $c = g^m h^r \mod n$ for some random $r \in \mathbb{Z}_n$
  - Dec(sk, c)
    – $m = \frac{L(c^{p-1} \mod p^2)}{L(g^{p-1} \mod p^2)} \mod p$
- Message expansion $= 3$

The Paillier Encryption [Pai99]

- Drawbacks of [OU98]: Decryption oracle enables factoring N
- Consider the modulus N^2 for $N = pq$ instead of p^2q. Then the message is defined over Z_N
- Logarithm
  - $(1+N)^x \equiv 1 + xN \bmod N^2$ and $(1+N)^N \equiv 1 \bmod N^2$
  - $\zeta := 1 + N$ is an order-N element of $\mathbb{Z}^*_{N^2}$.
  - Given $h \in H$, we have $h = \zeta^x$ for $x := (h-1)/N \bmod N$.
  - Define an isom $L : H \to \mathbb{Z}_N$, $h \mapsto (h-1)/N$
  - Given $g, h \in H$, $\log_g h = (\log_\zeta h)/(\log_\zeta g) = (h-1)/(g-1)$

The Paillier Encryption [Pai99]

- $\mathcal{E}(m) = g^m h^r \bmod N^2$ for $g = g_0^{\Phi(N)}$ and $h = g_0^N$.
- $\mathcal{E}(m)^{\phi(N)} \equiv g^m \bmod N^2$. Then $L(g^m)/L(g) \equiv m \bmod N$
- Paillier: $\mathcal{E}(m) = g^m r^N \bmod N^2$ for $g = 1 + N$ and random $r \in \mathbb{Z}_{N^2}^*$
- Use $\lambda(N)$ instead of $\phi(N)$
- Variants: EC version [Gal02], Generalized version [DJ01]

Additively Homomorphic Encryption with 2-Mul.

The Boneh-Goh-Nissim Encryption [BGN05]

- on the Subgroup Decision Problem
- Basic idea
  - $|\mathbb{G}| = pg$, $|g| = p$ and $|h| = q$. $\mathcal{E}(m) = g^m h^r$ is add homo on $\mathbb{Z}_p$
  - Use a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$
- Allow one mul: $e(g^{m_1} h^{r_1}, g^{m_2} h^{r_2}) = e(g^{m_1} g^{\alpha r_1}, g^{m_2} g^{\alpha r_2})$ $= e(g, g)^{(m_1+\alpha r_1)(m_2+\alpha r_2)} = g_1^{m_1 m_2}$. ♣
- How to decrypt ..
- Scalar Mul is not possible is p is large

Fully Homomorphic Encryption

The Gentry Encryption [Gen09]

- Hard Problems on Ideal Lattice
- Basic idea
  - Build somewhat homomorphic encryption whose ciphertexts becomes impossible to decrypt after bounded homomorphic operations
  - Homomorphic evaluation of its own decryption circuit $\Rightarrow$ Flush accumulated noise
    – Imagine a locked box containing another locked box, but in the outer box has a key to open the inner box $\Rightarrow$ in the outer box, we can safely disclose the inner box
- Variants: [DGHV10] and so on

Fully Homomorphic Encryption [DGHV10]

- Work flow (Symmetric version of [DGHV10])
  - KeyGen(1^λ): the key is λ^2-bit odd integer p
  - Enc(p, m): To encrypt $m \in \{0, 1\}$
    – $c = m' + pq$, where m' is a λ-bit number s.t $m' = m \mod 2$ and q a λ^5-bit random number
  - Dec(p, c)
    – $m = (c \mod p) \mod 2$
- How homomorphic is it?
- Efficiency (in [DGHV10])
  - Ciphertext size: $\lambda^5 \cdot (\log \lambda)^k$ bits for some k
  - Decryption function evaluation: $\lambda^{10} \cdot (\log \lambda)^k$ computation
- Security: Approximate GCD problem

Additively Homomorphic Encryption with d-Mul.

The Melchor-Gaborit-Herranz Encryption [MGH10]

- on the Unique Shortest Vector Problem
- Basic idea
  - Framework to convert an additive homomorphic encryption based on lattice problem into an encryption supporting arbitrary addition with 2-multiplication

Research Direction

- BGN with large plaintext
- Additive HE allowing d multiplications
- Efficient Somewhat Homomorphic Enc using Lattice
- Optimited construction to specific applications
- ...

Ciphertext Operations

1. Numeric Data: Additive Homomorphic Encryption, Multiplicative Homomorphic Encryption
   - Basic Integer Operations: Add, Subtract, Mul, Div
   - Advanced Operations: Euclidean Alg, GCD, Modular Operations
   - Fast Operations: Gaussian Elimination, Newton Method, FFT
2. Non-Numeric Data: what is the operation of data?
   - Search
   - Intersection, Union and Difference

Encrypted Set Operations

1. We wish to design an encryption scheme satisfying ...
   - Intersection: $\mathcal{E}(A \cup B)$ from $\mathcal{E}(A), E(B)$
   - Union: $\mathcal{E}(A \cap B)$ from $\mathcal{E}(A), \mathcal{E}(B)$
   - Difference: $\mathcal{E}(A \setminus B)$ from $\mathcal{E}(A), \mathcal{E}(B)$

   - Find $\mathcal{E}(A \cup B)$ or $\mathcal{E}(A \cap B)$ from A and $\mathcal{E}(B)$
     - Reduced to the above for PKE $\mathcal{E}$
   - Find $A \cup B$ or $A \cap B$ from A and $\mathcal{E}(B)$
     - Keyword Privacy: Use encrypted keywords
2. Easy Solution: Use deterministic encryption for individual data :-)

Polynomial Representation of Sets

1. S_i: a subset of $\mathbb{Z}_N$
2. Poly Rep of S_i: $f_i(x) = (x - \alpha_1) \cdots (x - \alpha_k)$ for $a_j \in S_i$
   - $V(f)$: the set of all roots of $f(x)$ (in $\mathbb{Z}_N$)
   - $V(f_i) = S_i$
3. Multiset Union: $V(f_1(x) \cdot f_2(x)) = S_1 \cup S_2$
4. (Multi)Set Intersection: $V(af_1(x) + bf_2(x)) \supset S_1 \cap S_2$
   - If a, b are polynomials of deg f_1=deg f_2, $af_1 + bf_2 = c \gcd(f_1, f_2)$ for a poly c uniformly distributed over a fixed deg poly.

Set Encryption

1. E: an Additive Homo Enc on $\mathbb{Z}_N$
   - $E(a) + E(b) = E(a + b)$ for $a, b \in \mathbb{Z}_N$
   - $aE(b) = E(ab)$ for $a, b \in \mathbb{Z}_N$
2. Define an encryption of poly: $E(f)$
   - $E(a_0 + a_1x + \cdots + a_kx^k) := E(a_0) + E(a_1)x + \cdots + E(a_k)x^k$
3. Can compute $E(fg)$ given $E(f)$ and $E(g)$?
   - Yes if E is ring homomorphic.
4. Can compute $gE(f)$ given $E(f)$ and g?
   - $\left(\sum_i E(a_i)x^i\right)\left(\sum_j b_jx^j\right) = \sum_k\left(\sum_{i+j=k} b_jE(a_i)\right)x^k$
   - where $b_jE(a_i) = E(a_ib_j)$

Private Set Intersection

1. Let f_i be a polynomial corr. to a set A_i with size d
2. Encrypt each f_i into $E(r_i f_i)$ for randomly chosen r_i of degree d
3. $\sum_{i \in I} E(r_i f_i) = E(\sum_{i \in I} r_i f_i) = E(rf)$ for$f = \gcd_i(f_i)$ and a random poly r
4. Decrypt it to get rf and factorize it
5. Random polynomial r has one root at average. How to remove it?
6. What do you learn from rf if we know all r_j, f_j except $j = 1$

Private Set Intersection

1. There are n players P_i with set S_i
2. They want to compute the intersection of S_i without revealing other information
3. Application
   - Several companies collaboratively find their common customers without revealing other information (privacy)
   - In cloud, ...
4. With TTP, it is easy. W/o TTP, use secure multiparty computation, which runs in poly time, but not so practical

Private Set Intersection: Protocol

1. Each party P_i publishes $E(f_i)$
2. All parties collaborate to generate $g_i E(f_i)$ for each i
   - Each party P_j generates a random k-deg poly g_{ij}
   - Publish $g_{ij} E(f_i)$ and compute $g_i E(f_i)$ where $g_i = \sum_{j=1}^{n} g_{ij}$
   - Any $(n-1)$ members can have no information on g_i
3. Group Decryption: $\sum_i g_i E(f_i) = E(\sum_i g_i f_i)$ to $\sum_i g_i f_i$
   - $V(\sum_i g_i f_i) \supset \cap_i S_i$
   - Let $\sum_i g_i f_i = fg$ with $V(f) = \cap_i S_i$
   - g can have a root and it is random. Give a redundancy.
   - e.g. Each set element is encoded to end with 11111

Private Multiset Union

1. Let E be a multiplicative homomorphic encryption on $\mathbb{Z}_n$, i.e. $E(x)E(y) = E(xy)$
2. For a subset $A \subset \wp$, define $M_A = (\prod_{p_i \in A} p_i) \bmod n)$ where p_i's are prime corr. to elements of A
3. $E(M_A)E(M_B) = E(M_A M_B)$
4. Its decryption is a product of the elements in $A \cup B$
5. How to remove the garbage: Use the redundancy function (e.g. each element of $\wp$ ends with 11111.)

Private Set Union [SCK02]

1. How to remove the multiplicities?
2. Represent a set S_i with $1/f_i$ for a polynomial f_i with roots S_i
3. $\sum_i r_i/f_i = r/\mathbf{lcm} f_i$ for a random r
4. How to represent $1/f$ with finite coefficients
5. Rational Reconstruction: Given $f, g \in \mathbb{F}_q[x]/(p(x)$ of degree $\deg(p)/2$, one can efficiently recover f, g from f/g mod $p(x)$

Research Directions

- Non-interactive Set Intersection and Union
- How to merge: Support Intersection and Union with one encryption
  - Set Intersection with AHE
  - Set Union with MHE
- Towards complete set operations ...
- Application in voting, cloud computing, and image processing

For Further Reading

 J. Benaloh, "Verifiable secret-ballot elections," PhD Thesis, Yale Univ., 1987

 D. Boneh, E. Goh, and K. Nissm, "Evaluating 2-DNF formulas on ciphertexts," TCC 2005

 I. Damgård and M. Jurik, "A generalization, a simplification and some applications of Paillier's probabilistic public-key system," PKC 2001

 T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," IEEE Tran. Info. Theory, 1985

 S. Galbraith, "Elliptic curve Paillier schemes," J. Cryptology, 2002.

 C. Gentry, "Fully homomorphic encryption using ideal lattices," STOC 2009.

 S. Goldwasser and S. Micali, "Probabilistic encryption," JCSS 1984.

For Further Reading


A. Kawachi, K. Tanaka and K. Xagawa, "Multi-bit cryptosystems based on lattice problems," PKC 2007


C. Melchor, P. Gaborit, and J. Herrandz, "Additively homomorphic encryption with d-Operand multiplication", Crypto 2010


D. Naccache and J. Stern, "A new public key cryptosystem based on higher residues," ACM CCS 1998


T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," Eurocrypt 1998.


P. Paillier, "Public-key cryptosystems based on composite residuosity classes," Eurocrypt 1999


R. Rivest, L. Addleman, and M. Dertouzos, "On data banks and privacy homomorphism," Foundations of Sec. Comp., 1978.

For Further Reading

 R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Comm. ACM, 1978.

 T. Sander, A. Young and M. Yung, "Non-interactive CryptoComputing for NC^1," FOCS 1999

 M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," Eurocrypt 2010.

⁎ All pictures are from Google Image