Evaluating 2-DNF Formulas on Ciphertexts

Dan Boneh,
Eu-Jin Goh,
and Kobbi Nissim

Theory of Cryptography Conference 2005

Homomorphic Encryption

Enc. scheme is homomorphic to function f if

- from E[A], E[B], can compute E[f(A,B)]

e.g. f can be +, $\times$, $\oplus$, …

Ideally, want f = NAND, or f = {+,$\times$}

- Called doubly homomorphic encryption

Can do universal computation on ciphertext!

Why is doubly homomorphic encryption useful?

Gives efficient solutions for many problems. e.g.

1. 2 party Secure Function Evaluation
2. Computing on encrypted databases

App: Database Computation

Outsourced server with database containing encrypted data

- User wants to compute function g on encrypted data
  - e.g. data mining, data aggregation

With doubly homomorphic encryption,

- Database encrypted with doubly hom. enc.
- User sends g to server
- Server computes g on encrypted database
- Encrypted result returned to user

These applications are pretty cool,

so where can I get a fully homomorphic encryption scheme?

Sorry, it doesn’t exist (yet).

- Long standing open problem [RAD78]
- Existing schemes hom. to 1 function
  - E.g. ElGamal ($\times$), Paillier ($+$), GM ($\oplus$)

But some progress …

Main Result

Homomorphic encryption scheme that supports one $\times$ and arbitrary $+$.

- Based on finite bilinear groups with composite order
- Semantic security based on natural decision problem

Related Work

Sander et al. [SYY99]

- Enc. scheme — NC^1 circuit eval. on CTs

  $\Rightarrow$ Can evaluate 2-DNFs on CTs

But CT len. exponential in circuit depth

- CT size doubles for every + op
  - Poly. len. 2-DNF gives poly. size CT
- Our scheme — constant size CT

  — crucial for our apps

Enc. Scheme

Keygen(τ):

- G: bilinear group order $n = q_1 q_2$ on ell. curve over F_p.
- Pick rand $g, u \in G$. Set $h = u^{q_2}$.
- PK = (n, G, G_1, e, g, h) SK = q_1

Encrypt(PK, m): $m \in \{1,\ldots,T\}$

- Pick random r from Z_n.
- Output $C = g^m h^r \in G$.

Decrypt(SK, C):

- Let $C^{q_1} = (g^m h^r)^{q_1} = (g^{q_1})^m$; $v = g^{q_1}$
- Output m = Dlog of C^{q_1} base v.

Note: decrypt time is $O(\sqrt{T})$.

Homomorphisms

Given $A = g^a h^r$ and $B = g^b h^s$:

To get encryption of a + b

- pick random $t \in Z_n$
- compute $C = AB \cdot h^t = g^{a+b} h^{r+s+t} \in G$

To get encryption of a × b

- let $h = g^{\alpha q_2}$, $g_1 = e(g,g)$, $h_1 = e(g,h)$
- pick random $t \in Z_n$
- compute

$$C = e(A,B) \cdot h_1^{t} = g_1^{ab} h_1^{r'} \in G_1$$

Complexity Assumption

Subgroup assumption:

Gen. rand. bilinear group G of order $n = q_1 q_2$,

then following two distributions indistinguishable:

- x is uniform in G
- x is uniform in q_1—subgroup of G.

Thm: system is semantically secure, unless the subgroup assumption is false.

Why not use Pallier directly?

- Paillier CT: $C = g^m r^n \pmod{n^2}$
  - Can we directly apply bilinear map to C?

Short ans: No.

- Miller's alg. for pairing needs order of curve.
- Fact: Knowing order of curve mod n
  allows factoring of n.

Applications

what can you do with 1 $\times$ and arbitrary + ?

1. Evaluate multi-variate polynomials of total degree 2
   - Caveat: result in small set e.g. $\{0,1\}$

2. Evaluate 2-DNF formulas $\vee\ (b_{i,1} \wedge b_{i,2})$
   - By arithmetizing 2-DNF formulas to multi-variate poly. with deg 2

1) Evaluating Quadratic Poly.

polynomials of total deg 2

- $x_1 x_2 + x_3 x_4 + \ldots$

- $+, \times$ hom. allow eval. of such poly. on CT
- but to decrypt, result must be in known poly. size interval.
- evaluate dot products

2) 2 Party SFE for 2-DNF

Bob

$A = (a_1,\ldots,a_n)$

$\in \{0,1\}^n$

Alice

$\phi(x_1,\ldots,x_n) = \vee^k_{i=1}(y_{i,1} \wedge y_{i,2})$ s.t. $y_{i,*} \in \{x_1, \neg x_1, \ldots, x_n, \neg x_n\}$.

Get Arithmetization Φ:

- replace $\vee$ by $+$, $\wedge$ by $\times$, $\neg x_i$ by $(1 - x_i)$.
- Φ is poly. with total deg 2!

2-DNF Protocol (Semi-Honest)

Bob	Alice
$A = (a_1,\ldots,a_n)$	$\phi(x_1,\ldots,x_n) = \vee^k_{i=1}(y_{i,1} \wedge y_{i,2})$
	Φ = arith. of ϕ

Invoke Keygen(τ)

Encrypt A

$\xrightarrow{\text{PK}, E[a_1],\ldots,E[a_n]}$

Eval. $E[r \cdot \Phi(A)]$ for random r

$\xleftarrow{E[r \cdot \Phi(A)]}$

If decrypt = 0, emit 0. Else, 1.

Bob's Security: Alice cannot distinguish bet. Bob's possible inputs — from semantic security of E.

Alice's Security: Bob only knows if A satisfies $\phi()$ — by design, Bob output distrib. depends only on this.

SFE for 2-DNF

Communication Complexity = $O(n \cdot \tau)$

- garbled circuit comm. comp. = $\Theta(n^2)$

Secure against unbounded Bob

- garbled circuit (Alice garbles ϕ) secure against unbounded Alice

Prove security against malicious Bob
(details in paper)

Concrete applications

1. Improve basic step in Kushilevitz-Ostrovsky PIR protocol from $\sqrt{n}$ to $\sqrt[3]{n}$

2. Gadget: “check” if CT contains 1 of 2 values.
   - Most voter efficient E-voting scheme
   - Universally verifiable computation

PIR/SPIR

Bob: wants D(R,S)

Set assignment A:

$x_R = y_S = 1$,

$x_i = x_j = 0$

for $i \neq R$, $j \neq S$

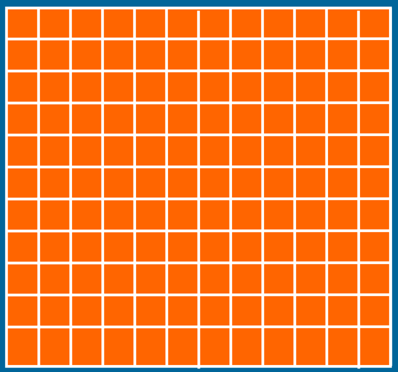
Do 2-DNF SFE

with A and ϕ

Get $\phi(A) = D(R,S)$

Database D

$\sqrt{n}$ $|D| = n$



$\sqrt{n}$

D uses 2-DNF

$\phi(x_1,\ldots,x_{\sqrt{n}}, y_1,\ldots,y_{\sqrt{n}})$

$= \vee_{D(i,j)=1} (x_i \wedge y_j)$

Comm. Complexity = $O(\tau \cdot \sqrt{n})$ [$O(\tau \cdot \sqrt[3]{n})$ balanced]

Alternative scheme — each db entry $O(\log n)$ bits

Gadget

Suppose CT: $C = E[v]$.

Given 2 messages v_0, v_1 and random r, anyone can compute

$$E\,[\, r \cdot (v - v_0)\,(v - v_1)\,]$$

- If $v \neq v_0, v_1$, result is E[random]
- Otherwise, result is E[0]

- can ensure/verify that CT is enc. of v_0 or v_1

Applications:

1. 2-DNF SFE secure against malicious Bob
2. E-voting: voter ballots need no ZK proofs
3. Universally Verifiable Computation
   - Anyone can check comp. public function on private inputs done correctly without learning anything else

Conclusions

Adding even limited additional homomorphism has many uses.

Open Problems:

- Extend encryption scheme to
  1. efficiently handle arbitrary messages
  2. arbitrary # of multiplications
- Find n-linear maps
  - allow eval. of polynomials with total deg n

Questions?