Fully Homomorphic Encryption

Damien Stehlé
Scribe: Antoine Pouille

November 24th 2014

The lecture is based on the following reference:

- Gentry, Sahai, Waters, CRYPTO '13 [5]

1 Introduction

Definition Let P be the set of plaintexts (here $P = \{0, 1\}$), and C the set of ciphertexts. A homomorphic encryption (HE) over (P, C) consists of four probabilistic polynomial time (ppt) algorithms:

- KeyGen: $1^\lambda \to sk, pk, evk$ (evaluation key)
- Enc: $pk, m \to c \in C$ with $m \in P$
- Dec: $sk, c \in C \to m' \in P$
- Eval: $evk, f, (c_1, \ldots, c_p) \in C^\ell \to c_f \in C$
  f "function" with ℓ inputs, described by a binary circuit, $\{0, 1\}^\ell \to \{0, 1\}$

Functionality The homomorphic encryption scheme is said F-homomorphic for a family of circuits F if:
If $c_1 = Enc_{pk}(m_1), \ldots, c_p = Enc_{pk}(m_\ell)$
Then $Dec_{sk}(Eval_{evk}(f, (c_1, \ldots, c_\ell) = f(m_1, \ldots, m_\ell))\ \forall m_1, \ldots m_\ell \in \{0, 1\}\ \forall f \in F$, with overwhelming probability over the random coins of KeyGen, Enc, Eval and Dec.

Security Indistinguishability under chosen-plaintext attack (IND-CPA) security of Enc: The distributions $(pk, evk, Enc_{pk}(0))$ and $(pk, evk, Enc_{pk}(1))$ are computationnally indistinguishable.

Remark Indistinguishability under chosen ciphertext attack (IND-CCA) security (with decryption oracle) is impossible.
IND-CCA1 security (with decryption queries before the challenge phase) might be possible.

Compactness Homomorphic encryption (HE) is said compact if $\exists c > 0$ such that
$\forall f \in F, \forall m_1, \ldots, m_l, bitsize(Eval(evk, f, Enc_{pk}(m_1) \ldots Enc_{pk}(m_l))) \leq \lambda^c$ (to avoid trivial solutions)

A Homomorphic Encryption scheme (HE) is said fully-homomorphic if F is the set of all circuits and if HE is compact.

1.1 Applications

- Confidential outsourced computations
- Secure multi-party computations

2 History

It was suggested in 1978 by Rivest, Adleman, Dertouzos, which invented the concept [8].

Some partially homomorphic algorithms:

- El Gamal ($\times$)
  $(g^{r_1}, h^{r_1}.M)$ & $(g^{r_2}, h^{r_2}.M)$
  $\Rightarrow (g^{r_1+r_2}, h^{r_1+r_2} m_1 m_2)$
- Goldwasser-Micali (+) [6]
- Paillier (+) [7]
- Boneh, Goh, Nissim ($+^* \times +^*$) [1]
- First fully homomorphic encryption scheme: Gentry '09 [4]
- Brakerski-Vaikuntanathan '11 [2]

  Fully Homomorphic Encryption (FHE) based on Learning With Errors (LWE).

3 The GSW basic encryption scheme (Gentry Sahai Waters)

KeyGen $B \leftarrow U(\mathbb{Z}_q^{m\cdot(n-1)}), b = Bt + e$ with $t \leftarrow U(\mathbb{Z}_q^{(n+1)})$ and $e \leftarrow D_{\mathbb{Z}^m,\alpha q}$
The hardness of Learning With Errors (LWE) makes it computationally indistinguable from uniform distribution (over $\mathbb{Z}_q^{m\cdot n}$). (Here, $n, m = \tilde{O}(\lambda), q = \lambda^{O(\log\lambda)}, \alpha \simeq \frac{\sqrt{n}}{q}$.)
$PK := A = (b| - B) \in \mathbb{Z}_q^{m.n}$
$sk := s = \begin{pmatrix} 1 \\ t \end{pmatrix} \in \mathbb{Z}_q^n \quad (A \cdot s = b - Bt = e)$

Enc $A' = R.A$ with $R \leftarrow D_{\mathbb{Z}^{m\times m},\sigma}$.
If A is uniform, then A' will be almost uniform by leftover hash lemma (we can choose $\sigma = \sqrt{n}$).
Remark We can even take $R \leftarrow U(\{0,1\}^{n.m})$.
We define $C = A' + M.Id_n$ (which looks uniform, independently of M) ensures indistinguishability under chosen-plaintext attack (IND-CPA) security.

Dec We have:

$$\begin{aligned} C.s &= A's + Ms \\ &= R\underline{As} + Ms \\ &= Re + Ms \end{aligned}$$

We have $\|R.e\| \leq poly(m).\alpha q$
If $Cs - s$ is small, then reply 1.
If Cs is small, then reply 0.

Is it homomorphic?
Let $C_1.s = M_1 s + e_1$ and $c_2.s = M_2 s + e_2$.
Then $(C1 + C2)$ is a valid ciphertext for $M_1 + M_2$.
We have:

$$\begin{aligned} (c_1 + c_2)s &= M_1 s + e_1 + M_2 s + e2 \\ &= (M_1 + M_2)s + \underbrace{(e_1 + e_2)}_{\text{new noise } e_+} \end{aligned}$$

The new noise satisfies $\|e_+\| \leq \|e_1\| + \|e_2\|$.

If α is small enough, then $\|e_+\| << q$ and Dec works correctly.

$$\begin{aligned}(C_2.C_1).s &= C_2(C_1 s)\\ &= C_2(M_1 s + e_1)\\ &= M_1(C_2 s) + C_2 e_1 \quad (M_1 \in \{0,1\} : \text{scalar})\\ &= M_1(M_2 s + e_2) + C_2 e_1\\ &= (M_1 M_2)s + M_1 e_2 + C_2 e_1\end{aligned}$$

It fails because $C_2 e_1$ is not small mod q! It's not multiplicatively homomorphic.

4 The GSW homomorphic encryption scheme

4.1 Description

It relies on three functions:
BD $Z_q \to \{0,1\}^{l=\lfloor \log_2(q) \rfloor + 1}$
$x \to (x_0, \ldots, x_{l-1})$ such that $x = \sum_{i=0}^{l-1} x_i 2^i$
BD $^{-1}$: $Z^l \to Z_q$
$(x_0, \ldots, x_{l-1}) \to x = \sum_{i=0}^{l-1} x_i 2^i [q]$ (Note that we have $BD^{-1} \circ BD = id, BD \circ BD^{-1} \neq id$)
P2 $Z_q \to Z_q^l$
$x \to (x, 2x, 4x, \ldots, 2^{l-1}x)$
Extended to vectors (acting entry by entry)
Extended to matrices (row after row).

Properties We have the following properties.

$$\begin{aligned}BD^{-1} \circ BD &= id\\ < a, s > &= < BD(a), P2(s) >\\ < a, P2(s) > &= < BD^{-1}(a), s >\end{aligned}$$

Enc ($M \in \{0,1\}$) does not output elements in $Z_q^{n.n}$, but in $\{0,1\}^{N.N}$ instead, with $N = n.l \simeq n \log(q)$.
$BD(R.A) = K$
And, $c = BD(RA) + MId_N$
Remark The public key and secret key are the same as before.

$$\begin{aligned}C &= BD(BD^{-1}(BD(RA) + M.Id)) = BD(RA + M.BD^{-1}(Id_{N.N}))\\ &= BD(RA + M.BD^{-1}(Id))\end{aligned}$$

RA looks uniform mod q, so $RA + \underbrace{M.BD^{-1}(Id_{N.N})}_{\text{fixed}}$ looks uniform mod q, independently of M.

And $BD(RA + M.BD^{-1}(Id_{N.N}))$ too, as if its distribution did not depend on M: we have indistinguishability under chosen-plaintext attack (IND-CPA) security.

Dec ($C \in \{0,1\}^{N.N}, s \in Z_q^n$)

$$\begin{aligned}C.P2(s) &= BD(BD^{-1}(BD(RA) + MId)).P2(s)\\ &= BD^{-1}(BD(RA) + MId).s\\ &= (BD(RA) + MId).P2(s)\\ &= BD(RA).P2(s) + M.P2(s)\\ &= RAs + M.P2(s)\\ &= \underbrace{Re}_{\text{small}} + \underbrace{M.P2(s)}_{\text{big}}\end{aligned}$$

If $c.P2(s) - P2(s)$ small then return 1.
If $c.P2(s)$ small then return 0.

4.2 Homomorphism

Let's assume that $C_1 P2(s) = e_1 + M_1 P2(s)$ and $C_2 P2(s) = e_2 + M_2 P2(s)$.
Now, the c_i are $\{0,1\}^{N.N}$ and we replaced s by $P_2(s)$.

Then $(C_1 + C_2)P2(s) = (e_1 + e_2) + (M_1 + m_2)P2(s)$
$(C_2.C_1)P2(s) = (M_2 M_1)P2(s) + (M_1.e_2 + C_2.e_1)$

We have the following relations:
$\|e_+\| \leq \|e_1\| + \|e_2\|$
$\|e_\times\| \leq |M_1|.\|e_2\| + poly(N).\|e_1\| \leq poly(m.\log(q)).(\|e_1\| + \|e_2\|)$

There are two difficulties to get Homomorphic Encryption for binary circuits:
1 - Add is mod q, instead of mod 2.
2 - $C_2.C_1$ is not binary... and we end up with the same problem as before.

1 - NAND $\{0,1\}^2 \to \{0,1\}$ is universal. Hence, it is sufficient to play with NAND circuits.
$Eval(NAND, c_1, c_2) := (Id - C_2 C_1) \pmod 2$
$$(Id - C_2 C_1)P_2(s) = \underbrace{(1 - M_2 M_1)}_{\in \{0,1\} \text{ if } M_1, M_2 \in \{0,1\}} P2(s) + (M_1 e_2 + C_2 C_1)$$

2 - Replace $C_2 C_1$ by $BD(BD^{-1}(Id - C_2 C_1))$. Indeed, it is binary, and we have:
$BD(BD^{-1}(Id - C_2 C_1))P2(s) : (1 - M_2 M_1)P2(s) + e_x$

$$\begin{aligned} \|e_x\| &\leq |M1|\|e_2\| + poly(N)\|e_1\| \\ &\leq poly(n\log(q)).(\|e_1\| + \|e_2\|) \end{aligned}$$

5 Noise growth and Fully Homomorphic Encryption

Fresh noises (at the input of the circuit) are smaller than $B = \alpha q.poly(n\log(q))$
At the end of the NAND circuit, the noise is smaller than $B.poly(n\log(q))^D$ with D the circuit depth.
And finally, smaller than $\alpha q.poly(n\log(q))^{D+1}$

We want te be able to decrypt the output ciphertext.
For this, it suffices to have $\alpha.q.poly(n\log(q))^{D+1} \leq \frac{q}{16}$.
This can be obtained by setting $\alpha \approx \frac{1}{poly(n\log(q))^{D+1}}$

For a fixed α, we are limited to depth D circuits, for some D. So, it is not fully homomorphic.

Gentry's bootstrapping from Homomorphic Encryption to Fully Homomorphic Encryption.
Let c be a ciphertext. Then define:
$c' := Eval_{evk}(DecryptionCircuit, Enc_{pk}(c), Enc_{pk}(sk))$

$$\begin{aligned} Dec_{sk}(c') &= DecryptionCircuit(Dec_{sk}(Enc_{pk}(c)), Dec_{sk}(Enc_{pk}(sk))) \\ &= DecryptionCircuit(c, sk) \\ &= \text{plaintext underlying } c \end{aligned}$$

Remarks

- The decryption algorithm can be converted into a NAND circuit.
- The ciphertext c and the secret key sk are decomposed in bits $c_1, \ldots, sk_1, \ldots$, and each one of these is re-encrypted.
- The decryption circuit must be already among the circuits we can homomorphically evaluate.

Exercise Implement decryption with a $O(\log(n \log\log(q)))$ depth circuit.

We can set $D \geq O(\log(n \log\log(q)))$ in GSW and get a Fully Homomorphic Encryption scheme via Gentry's bootstrapping. We obtain $\alpha \simeq \frac{1}{(n \log\log(q))^{\log(n \log\log(q))}}$, which is only slightly smaller than $\frac{1}{poly(n)}$.

Remark In [3], Brakerski and Vaikuntanathan get $\alpha \simeq \frac{1}{poly(n)}$ $\|e_\times\| \leq |M_1|.\|e_2\| + poly(N).\|e_1\|$

The other issue with Gentry's bootstrapping is that we need to publicly give $enc_{pk}(sk)$ (evaluation key). (More precisely, we are given encryptions of the bits of sk.)
We do not know how to make this provably secure. We assume it is, and call it a circular security assumption.

References

[1] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. In *Theory of cryptography*, pages 325–341. Springer, 2005.

[2] Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-lwe and security for key dependent messages. In *Advances in Cryptology–CRYPTO 2011*, pages 505–524. Springer, 2011.

[3] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) lwe. *SIAM Journal on Computing*, 43(2):831–871, 2014.

[4] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, volume 9, pages 169–178, 2009.

[5] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Advances in Cryptology–CRYPTO 2013*, pages 75–92. Springer, 2013.

[6] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of computer and system sciences*, 28(2):270–299, 1984.

[7] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in cryptology—EUROCRYPT'99*, pages 223–238. Springer, 1999.

[8] Ronald L Rivest, Len Adleman, and Michael L Dertouzos. On data banks and privacy homomorphisms. *Foundations of secure computation*, 4(11):169–180, 1978.