Paillier homomorphic encryption

and its application to build a share conversion protocol

Federico Mazzone

Università degli Studi di Trento

June 18, 2020

Part I

Multiplicative-to-additive share conversion protocol [1]

Context

Alice's share

$a \in \mathbb{Z}_q$

Secret

$x = a \cdot b \pmod{q}$

Bob's share

$b \in \mathbb{Z}_q$

$\downarrow$

Share conversion protocol

$\downarrow$

Alice's share

$\alpha \in \mathbb{Z}_q$

Secret

$x = \alpha + \beta \pmod{q}$

Bob's share

$\beta \in \mathbb{Z}_q$

Assumptions

- There are two actors: Alice and Bob.
- They both know a prime q.
- Alice knows $a \in \mathbb{Z}_q$ (private).
- Bob knows $b \in \mathbb{Z}_q$ (private).
- They both do not know anything useful about the secret share of the other.
- Alice is associated with a public key E_A, with modulus n, for an additively homomorphic scheme:

$$\mathrm{Dec}(c_1 +_E c_2) = m_1 + m_2 \pmod{n}$$
$$\mathrm{Dec}(a \times_E c) = am \pmod{n}$$

- K is a public integer such that $K > q$ and $n > K^2 q$.

Protocol

A	B

$c_A := E_A(a)$ — A $\xrightarrow{c_A}$ B

A $\longleftrightarrow$ B: A proves in ZK that $a < K$

B: $\beta' \in_R \mathbb{Z}_n$
$\beta := -\beta' \pmod q$
$c_B := (b \times_E c_A) +_E E_A(\beta')$

B $\xrightarrow{c_B}$ A

A $\longleftrightarrow$ B: B proves in ZK that $b < K$

A: $\alpha' := D_A(c_B)$
$\alpha := \alpha' \pmod q$

Numerical example

- Let us consider the prime $q = 101$ and the secret $x = 45$.
  Alice and Bob have respectively the private shares $a = 70$ and $b = 80$.
- Alice is associated to a public key of an homomorphic encryption scheme that works modulo $n = 1115111$.
  She encrypts a into c_A and sends it to Bob.
- Bob picks randomly $\beta' = 954245 \in \mathbb{Z}_n$, builds the quantity c_B as described before and sends it to Alice.
- Alice decrypts the quantity, getting $ab + \beta' = 959845$.
- Now, Bob sets $\beta := -\beta' \pmod{q} = 3$ and Alice sets $\alpha := ab + \beta' \pmod{q} = 42$. Note that $\alpha + \beta = 45 = x$.

Correctness

Assuming both players are honest, Alice receives:

$$\alpha = ab - \beta \pmod{n}$$

But we need this to be true mod $q < n$.

The only way is that the reduction mod n does not apply, namely the protocol is correct when $ab + \beta' < n$.

The protocol is almost surely correct:

$$\mathbb{P}(\beta' \geq n - ab) = \frac{n - (n - ab)}{n} = \frac{ab}{n} < \frac{K^2}{K^2 q} = \frac{1}{q}$$

Security

Both the messages look like random quantity to the other actor:

- Alice's one due to the semantic security of the encryption:

$$c_A := E_A(a).$$

- Bob's one due to the added noise β':

$$c_B := (b \times_E c_A) +_E E_A(\beta') \longrightarrow ba + \beta'.$$

Remarks

- The **ZK proofs** only ensure correctness and not the security: an adversary may be just interested in making the protocol fail, without recovering the other's secret.
- The described protocol is **secure** and **overwhelmingly correct**. We can modify it and choose $\beta' \in_R \mathbb{Z}_{n-K^2}$, so with a distribution statistically close to the one on $\mathbb{Z}_n$. In this way the protocol becomes just **statistically secure** but **always correct**.
- The **homomorphic cryptosystem** let Bob to make computations without getting any information on the Alice's share.
- For the **parameters size**, the two ZK proofs must be considered. There is an efficient range proof [1] that require $K \sim q^3$ and so $n \sim q^8$. Indeed, a typical choice of parameters is q 256 bits, K 768 bits and n 2048 bits.

Part II

Paillier homomorphic encryption scheme [2]

Composite residuosity

Fix $n := pq$ where p, q are RSA primes. Let $\lambda := \mathrm{lcm}(p-1, q-1)$.

N.B.: p, q with the same length, so that

$$\implies \quad p \nmid q-1 \wedge q \nmid p-1 \quad \implies \quad \gcd(n, \phi(n)) = 1 \quad \implies \quad \gcd(n, \lambda) = 1$$

Definition

A number z is a **n-th residue** modulo n^2 iff

$$\exists y \in \mathbb{Z}^*_{n^2} : \qquad y^n = z \pmod{n^2}$$

Remark

The n-th residues form a multiplicative subgroup of $\mathbb{Z}^*_{n^2}$ of order $\phi(n)$.

$\mathrm{CR}[n]$ = the problem of deciding whether an element is a n-th residue or not.

Composite residuosity classes

Let $g \in \mathbb{Z}^*_{n^2}$

$$\begin{aligned}\mathcal{E}_g : \mathbb{Z}_n \times \mathbb{Z}^*_n &\to \mathbb{Z}^*_{n^2} \\ (x, y) &\mapsto g^x y^n \pmod{n^2}\end{aligned}$$

Let $\mathcal{B}$ be the elements of $\mathbb{Z}^*_{n^2}$ with order a nonzero multiple of n. Note that:

$$g \in \mathcal{B} \implies \mathcal{E}_g \text{ bijective}$$

Then, for $g \in \mathcal{B}$ and $w \in \mathbb{Z}^*_{n^2}$, there exists a unique pair $(x, y) \in \mathbb{Z}_n \times \mathbb{Z}^*_n$ such that $\mathcal{E}_g(x, y) = w$.

This unique $x \in \mathbb{Z}_n$ is called the **n-th residuosity class** of w wrt g and is denoted by $[\![w]\!]_g$.

Lemma

- *$[\![w]\!]_g = 0$ iff w is a n-th residue modulo n^2.*
- $[\![w_1 w_2]\!]_g = [\![w_1]\!]_g + [\![w_2]\!]_g \mod n$

Composite residuosity class problem

Class$[n, g]$ = the problem of computing $[\![w]\!]_g$ for given w, g, n.

Lemma

Class$[n, g]$ is random-self-reducible over w.
(hint: $w = \bar{w} g^\alpha \beta^n \Longrightarrow [\![w]\!]_g = [\![\bar{w}]\!]_g + \alpha$)

Lemma

Class$[n, g]$ is random-self-reducible over g.
(hint: $[\![w]\!]_{g_1} = [\![w]\!]_{g_2} [\![g_2]\!]_{g_1} \Longrightarrow [\![g_1]\!]_{g_2}^{-1} = [\![g_2]\!]_{g_1} \Longrightarrow [\![w]\!]_{g_1} = [\![w]\!]_{g_2} [\![g_1]\!]_{g_2}^{-1}$)

So we can just look upon it as a computational problem which only depends on n and we can denote it by Class$[n]$.

Theorem

$$\mathit{Class}[n] \leq_p \mathit{Fact}[n]$$

On $\mathcal{S}_n := \{u \in \mathbb{Z}_{n^2} : u = 1 \pmod{n}\}$ we can define the function

$$L(u) := \frac{u-1}{n}$$

Lemma

$$\forall w \in \mathbb{Z}^*_{n^2}, \quad L(w^\lambda \mod n^2) = \lambda [\![w]\!]_{1+n} \pmod{n}$$

Proof (of the theorem)

We can prove that $[\![g]\!]_{1+n} = [\![1+n]\!]_g^{-1}$ is invertible and, by Lemma, we have that $L(g^\lambda \mod n^2) = \lambda [\![g]\!]_{1+n}$ is invertible. So

$$\frac{L(w^\lambda \mod n^2)}{L(g^\lambda \mod n^2)} = \frac{\lambda [\![w]\!]_{1+n}}{\lambda [\![g]\!]_{1+n}} = \frac{[\![w]\!]_{1+n}}{[\![g]\!]_{1+n}} = [\![w]\!]_g \pmod{n}$$

Solving Fact[n] implies knowing λ.

Conjectures

$CR[n] \equiv$ D-Class$[n]$

$(\leq_p)$ decide whether or not $[\![w]\!]_g = 0$.
$(\geq_p)$ decide whether or not wg^{-x} is a n-th residue.

$$CR[n] \equiv \text{D-Class}[n] \leq_p \text{Class}[n] \leq_p \text{RSA}[n, n] \leq_p \text{Fact}[n]$$

Conjecture

Decisional Composite Residuosity Assumption (DCRA):
There exists no polynomial time algorithm to decide $CR[n]$.

Conjecture

Computational Composite Residuosity Assumption (CCRA):
There exists no polynomial time algorithm to solve Class$[n]$.

Paillier encryption scheme

Key generation

1. Choose p, q RSA primes.
2. Compute $n := pq$ and $\lambda := \operatorname{lcm}(p-1, q-1)$.
3. Define the integer division quotient function $L(u) := (u-1)/n$.
4. Choose $g \in_R \mathbb{Z}^*_{n^2}$ such that the inverse of $L(g^\lambda \mod n^2) \pmod n$ exists.
5. Public key $= (n, g)$. Private key $= (p, q, \lambda)$.

Encryption: plaintext $m \in \mathbb{Z}_n$

1. Pick $r \in_R \mathbb{Z}^*_n$.
2. Compute the ciphertext $c := \mathcal{E}_g(m, r) = g^m r^n \pmod{n^2}$.

Decryption: ciphertext $c \in \mathbb{Z}_{n^2}$

1. Compute the plaintext $m := \dfrac{L(c^\lambda \mod n^2)}{L(g^\lambda \mod n^2)} \pmod n$.

Additive homomorphic properties

$$E(m_1) \cdot E(m_2) = g^{m_1} r_1^n \cdot g^{m_2} r_2^n = g^{m_1+m_2} \cdot (r_1 r_2)^n$$
$$E(m)^a = (g^m \cdot r^n)^a = g^{am} \cdot (r^a)^n$$

So we can define operations on ciphertexts:

$$c_1 +_E c_2 := c_1 \cdot c_2 \pmod{n^2}$$
$$a \times_E c := c^a \pmod{n^2}$$

and get

$$D(c_1 +_E c_2) = m_1 + m_2 \pmod{n}$$
$$D(a \times_E c) = am \pmod{n}$$

Security

One-way encryption

ciphertext $c \nrightarrow$ plaintext m

$\iff$

Computational Composite Residuosity Assumption

$$w \nrightarrow [\![w]\!]_g$$

Semantic security (IND-CPA)

$$m_0, m_1 \rightarrow b \in_R \{0,1\}$$
$$\text{guess } b\,? \leftarrow c_b := \text{enc}(m_b)$$

$\iff$

Decisional Composite Residuosity Assumption

$$w, x \nrightarrow x \stackrel{?}{=} [\![w]\!]_g$$

Chosen-ciphertext security (IND-CCA)

$$c_0, \ldots, c_k \neq c_b \rightarrow e_i := \text{dec}(c_i)$$
$$\text{guess } b\,? \leftarrow e_0, \ldots, e_k$$

NO!

$$c_0 = 2 \times_E c_b \rightarrow 2m_B$$

Remarks - Encryption

Encryption: plaintext $m \in \mathbb{Z}_n$

1. Pick $r \in_R \mathbb{Z}_n$.
2. Compute the ciphertext $c := \mathcal{E}_g(m, r) = g^m r^n \pmod{n^2}$.

Remarks:

- Smart choice of g: take it small.
- Pre-processing techniques for g^m (g is constant).
- Choose r and compute r^n in advance.

Remarks - Decryption

Decryption: ciphertext $c \in \mathbb{Z}_{n^2}$

1. Compute the plaintext $m := \dfrac{L(c^\lambda \mod n^2)}{L(g^\lambda \mod n^2)} \pmod{n}$.

Remarks:

- Pre-compute $L(g^\lambda \mod n^2)^{-1} \mod n$ once for all.
- Turn the division by n in the function L into a mulitplication by n^{-1} mod $2^{|n|}$, which can be pre-computed once for all.
- Make the computation mod p and mod q with the respective L_p and L_q functions, then apply the CRT.

Bibliography

R. Gennaro and S. Goldfeder.
Fast Multiparty Threshold ECDSA with Fast Trustless Setup.
In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 1179–1194, 2018.

P. Paillier.
Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.
In *International conference on the theory and applications of cryptographic techniques*, pages 223–238. Springer, 1999.