

HOMOMORPHIC CRYPTOSYSTEMS

Dörte K. Rappe
University of Dortmund

February 8, 2001

Outline

1. Definition
2. The pros and cons
3. Examples
4. Applications and situation
5. Encryption of the dihedral group
6. Outlook and open questions

Definition

1978 Rivest, Adleman, Dertouzos

1991 Feigenbaum, Merritt:

"Is there an encryption function $E()$ such that both $E(x + y)$ and $E(xy)$ are easy to compute from $E(x)$ and $E(y)$?"

Definition. Let (\mathcal{M}, \circ) be a semigroup. A cryptosystem $(\mathcal{K}, \mathcal{M}, \mathcal{C}, E, D)$ is called homomorphic, if there exists an efficient algorithm A such that:

$$A(E(x), E(y)) = E(x \circ y).$$

The cryptosystem is called algebraically homomorphic if \mathcal{M} is a ring and if there exist two efficient algorithms Add and $Mult$ such that:

$$\begin{aligned} Add(E(x), E(y)) &= E(x + y) \text{ and} \\ Mult(E(x), E(y)) &= E(x \cdot y). \end{aligned}$$

Example. *RSA is an example for a homomorphic encryption scheme since*

$$E(x \cdot y) = (x \cdot y)^e = x^e \cdot y^e = E(x) \cdot E(y)$$

in $\mathcal{C} := \mathbb{Z}/n\mathbb{Z}$.

It is undecided whether RSA is algebraically homomorphic.

The pros and cons

- **Pros:**
Many applications
- **Cons:**
New attacks are possible.
E.g if RSA is used for signing, a valid signature for $x \cdot y$ can be computed from given signatures for x and y .
- **Possible solution:**
Probabilistic schemes

Examples

The Goldwasser-Micali scheme

1982: Goldwasser, Micali: probabilistic cryptosystems, semantic security.

Goldwasser Micali key generation

1. Generate two primes p and q .
2. Compute $n := pq$.
3. Choose $a \in (\mathbb{Z}/n\mathbb{Z})^*$ with $\left(\frac{a}{n}\right) = 1$ and $a \notin \text{QR}(n)$.
4. The public key is (n, a) and the corresponding private key is (p, q) .

The probabilistic Goldwasser-Micali scheme

Let $\mathcal{M} := \mathbb{Z}/2\mathbb{Z}$ and $\mathcal{C} = \mathcal{Z} := (\mathbb{Z}/n\mathbb{Z})^*$.

Define:

$$E(m, z) := a^m z^2 \pmod n$$

and

$$D(c) := \begin{cases} 0, & \text{if } c \in \text{QR}(n) \\ 1, & \text{if } c \notin \text{QR}(n) \end{cases}$$

Correctness:

If $m = 0$ then $c = z^2 \bmod n \in \text{QR}(n)$. If $m = 1$ then $c = az^2 \bmod n \notin \text{QR}(n)$. Using the private key the receiver can compute $\left(\frac{c}{p}\right)$ and distinguish between quadratic residues and quadratic nonresidues modulo n .

Properties:

- The scheme is semantically secure if the Quadratic Residuosity Assumption holds.
- Due to the properties of quadratic residues the scheme is homomorphic.
- It is inefficient since every bit of the message is extended to $|n|$ bits ciphertext.

More efficient schemes:

- Blum, Goldwasser: $E : ((\mathbb{Z}/2\mathbb{Z})^\ell, +) \rightarrow (\mathbb{Z}/2\mathbb{Z})^\ell \times (\mathbb{Z}/n\mathbb{Z})^*$
- Benaloh: $E : (\mathbb{Z}/m\mathbb{Z}, +) \rightarrow (\mathbb{Z}/n\mathbb{Z})^*$
- Lipton, Sander: $E : (\mathbb{Z}/s\mathbb{Z}, +) \rightarrow (\mathbb{Z}/n\mathbb{Z})^*$
- Paillier: $E : (\mathbb{Z}/n\mathbb{Z}, +) \rightarrow (\mathbb{Z}/n^2\mathbb{Z})^*$
- Okamoto, Uchiyama: $E : (\mathbb{Z}/p\mathbb{Z}, +) \rightarrow (\mathbb{Z}/p^2q\mathbb{Z})^*$
- Naccache, Stern: $E : (\mathbb{Z}/\sigma\mathbb{Z}, +) \rightarrow (\mathbb{Z}/n\mathbb{Z})^*$
-

Applications and situation

- Protection of mobile agents
- Multiparty computations
- Watermarking
- Secret sharing schemes
- Threshold-schemes
- Zero-knowledge proofs
- Election schemes
- ...

Protection of mobile agents

→ executable encrypted programs

→ encryption of $(\mathbb{F}_2, +, \cdot)$

Situation:

It is still unknown if algebraically homomorphic cryptosystems $E : \mathbb{F}_2 \rightarrow \mathcal{C}$ exist.

- negativ:

1996 Boneh, Lipton: every deterministic algebraically homomorphic cryptosystem can be broken in sub-exponential time

- positiv:

Ben-Or, Cleve: $(\mathbb{F}_2, +, \cdot)$ can be encoded in $SL_3(\mathbb{F}_2)$.
→ it remains to encrypt $SL_3(\mathbb{F}_2)$

$SL_3(\mathbb{F}_2)$

$$\begin{array}{ccc}
 (\mathbb{F}_2, +, \cdot) & \xrightarrow{\text{encoding}} & SL_3(\mathbb{F}_2) \\
 x & \mapsto & \begin{pmatrix} 1 & 0 & x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} =: M(x)
 \end{array}$$

1. $M(x)M(y) = M(x + y)$

2. $\exists T : TM(x)T^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix}$

$$\exists S : SM(y)S^{-1} = \begin{pmatrix} 1 & y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\rightarrow M(xy) = \left[\begin{pmatrix} 1 & y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{pmatrix} \right]$$

Proposition.

$$\langle M(x), S, T \rangle = SL_3(\mathbb{F}_2)$$

Computing with encrypted functions:

Alice has an algorithm to compute a function f . Bob has an input x and is willing to compute $f(x)$ for her, but Alice wants Bob to learn nothing substantial about f .

Sander, Tschudin: Solution for polynomials f

Definition 1. A cryptosystem is called mixed multiplicatively homomorphic, if there exists an efficient algorithm *MixedMult* such that:

$$\text{MixedMult}(E(x), y) = E(x \cdot y).$$

Lemma. Every additively homomorphic cryptosystem on $\mathbb{Z}/s\mathbb{Z}$ is also mixed multiplicatively homomorphic.

Proof. Let $y = \sum_{i=0}^{|n|} y_i 2^i$ with $y_i \in \{0, 1\}$. It is

$$E(2x) = \text{Add}(E(x), E(x)) = E(x + x),$$

$$E(4x) = \text{Add}(E(2x), E(2x)), \dots$$

\Rightarrow Using *Add* you can compute $E(2^i x)$ for $1 \leq i \leq |n|$. You get $E(x \cdot y)$ by adding all $E(2^i x)$ where $y_i \neq 0$. \square

Protocol for computing with encrypted polynomials:

1. Alice transforms her cleartext polynomial $f = \sum a_{i_1 \dots i_s} X_1^{i_1} \dots X_s^{i_s}$ into the encrypted polynomial $E(f) := \sum E(a_{i_1 \dots i_s}) X_1^{i_1} \dots X_s^{i_s}$
2. Alice sends $E(f)$ to Bob.
3. Bob evaluates $E(f)$ on his input $x = (x_1, \dots, x_s)$ as follows:
 - Bob evaluates the monomials of f on his input x_1, \dots, x_s and stores the results in a list $L := [\dots, (x_1^{i_1} \dots x_s^{i_s}), \dots]$.
 - Using *MixedMult* he computes the list $M := [\dots, E(a_{i_1 \dots i_s} x_1^{i_1} \dots x_s^{i_s}), \dots]$.
 - Using *Add* he adds up the elements of M .
4. Bob sends the result back to Alice.
5. Alice decrypts the result that equals $E(f(x))$, and obtains $f(x)$.

Situation:

Many homomorphic cryptosystems over $(\mathbb{Z}/s\mathbb{Z}, +)$ exist.

Encryption of the dihedral group

Definition. The group D_N , $N \geq 3$, represented by

$$\begin{aligned} D_N &:= \langle \sigma, \tau \mid \sigma^N = 1 = \tau^2, \tau\sigma\tau^{-1} = \sigma^{-1} \rangle \\ &= \{ \sigma^i \tau^j \mid 0 \leq i < N, j = 0, 1 \}. \end{aligned}$$

is called the dihedral group of order $2N$.

Motivation: We have

$$D_3 = \left\langle \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right\rangle \subset SL_3(\mathbb{F}_2).$$

Encryption:

$$\begin{aligned} E : D_N &\rightarrow GL(6, \mathbb{Z}/n\mathbb{Z}), \quad n = pq \\ \sigma^i \tau^j &\mapsto dS^i T^j \end{aligned}$$

with $d \in (\mathbb{Z}/n\mathbb{Z})^*$ randomly chosen and $S, T \in GL(6, \mathbb{Z}/n\mathbb{Z})$ satisfying the following relation:

$$TST^{-1} = aS^r$$

with $a \in (\mathbb{Z}/n\mathbb{Z})^*$ and $r \in \mathbb{Z}$.

The public key is (N, n, S, T) and the private key is (p, q, r, a)

Some conditions:

$$\det S =: \zeta^2 \in \text{QR}(n), \quad (1)$$

$$\left(\frac{\det T}{p}\right) = \left(\frac{\det T}{q}\right) = -1, \quad (2)$$

$$r^j \bmod N \equiv \begin{cases} 1 & \text{if } j \equiv 0 \pmod{2} \\ -1 & \text{if } j \equiv 1 \pmod{2} \end{cases}, \quad (3)$$

$$\gcd(r - 1, N) = 1, \quad N \text{ odd and small}, \quad (4)$$

$$N|p - 1, \quad N|q - 1, \quad (5)$$

To decrypt an expression of the form dS^iT^j you have to find $i \bmod N$ and $j \bmod 2$:

Decryption:

1. $j \bmod 2$:

- Compute $\det(d \cdot S^iT^j) = d^6 \cdot (\zeta^2)^i \cdot (\det T)^j$.
- Compute the Jacobi symbol

$$\left(\frac{d^6 \cdot (\zeta^2)^i \cdot (\det T)^j}{p} \right) = (-1)^j$$

$$= \begin{cases} -1 \Rightarrow j \equiv 1 \pmod{2} \\ 1 \Rightarrow j \equiv 0 \pmod{2}. \end{cases}$$

2. $i \bmod N$

- Compute the commutator

$$[T, dS^iT^j] = (aS^r)^i.$$

- If x is the first entry of $(aS^r)^i$ and y is the first entry of aS^r then the discrete log problem $y^i = x$ has to be solved.

Variants

Variant with random exponents:

$$\sigma^i \tau^j \mapsto dS^{i+kN}T^{j+2\ell},$$

with $d \in (\mathbb{Z}/n\mathbb{Z})^*$ and $k, \ell \in \mathbb{Z}$ randomly chosen.

Since for decryption you have to compute $i \bmod N$ and $j \bmod 2$, decryption remains nearly unchanged.

Variant with random matrices T_z :

$$\sigma^i \tau^j \mapsto dS^{i+kN}T_z^{j+2\ell},$$

with $d \in (\mathbb{Z}/n\mathbb{Z})^*$, $k, \ell \in \mathbb{Z}$ randomly chosen and $T_z \in \{T \mid TST^{-1} = aS^r\}$.

For the decryption it has to be guaranteed that the random matrices commute. Then decrypting is equal to the first variant.

Outlook and open questions

- Efficiency and security of the encrypted dihedral group?
- Does an algebraically homomorphic encryption of \mathbb{F}_2 exist?
- Encryption of $SL_3(\mathbb{F}_2)$?
- Encryption of other classes of circuits/functions
- Details of applications

NC^1 circuits

Barrington: NC^1 circuits can be coded by the symmetric group S_5 .

→ homomorphic encryption of $S_5 \Rightarrow$ encryption of NC^1 circuits.

Situation:

An homomorphic encryption of the symmetric group S_5 has not been found yet.