Batch Somewhat Homomorphic Encryption Over Integer[1]

G. Droandi

Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche,
University of Siena, ITALY

19 dicembre 2013

Somewhat homomorphic

Chiavi

- *Segreta*: scelgo p dispari
- *Pubblica*: $x_0 = pq_0$, $\{x_i = pq_i + r_i; i \in [0, \tau]\}$,

cifratura

$m \in \{0, 1\}$ messaggio scelgo $S \subseteq \{1, \ldots \tau\}$

$$c = \left[m + 2r + 2 \sum_{i \in S} x_i \right]_{x_0}$$

decifratura

$$m = \Big[[c]_p \Big]_2$$

Chinese remainder Theorem

Chinese remainder Theorem

Dati $a_1, a_2, \ldots, a_k$ interi e $p_1, \ldots, p_k$ interi a due a due coprimi ($MCD(p_i, p_j) = 1$) allora esiste un intero x tale per ogni $i < k$

$$x \equiv a_i \mod p_i$$

Chinese remainder Theorem

Chinese remainder Theorem

Dati $a_1, a_2, \ldots, a_k$ interi e $p_1, \ldots, p_k$ interi a due a due coprimi ($MCD(p_i, p_j) = 1$) allora esiste un intero x tale per ogni $i < k$

$$x \equiv a_i \mod p_i$$

Tutte le soluzioni x sono congruenti modulo $\pi = p_1 \cdot p_2 \cdot \ldots p_k$.

Chinese remainder Theorem

Chinese remainder Theorem

Dati $a_1, a_2, \ldots, a_k$ interi e $p_1, \ldots, p_k$ interi a due a due coprimi ($MCD(p_i, p_j) = 1$) allora esiste un intero x tale per ogni $i < k$

$$x \equiv a_i \mod p_i$$

Tutte le soluzioni x sono congruenti modulo $\pi = p_1 \cdot p_2 \cdot \ldots p_k$.
$x \equiv y(\mod p_i)$ per ogni $1 \leq i \leq k$ se e solo se $x \equiv y(\mod \pi)$.

Chinese remainder Theorem

Chinese remainder Theorem

Dati $a_1, a_2, \ldots, a_k$ interi e $p_1, \ldots, p_k$ interi a due a due coprimi ($MCD(p_i, p_j) = 1$) allora esiste un intero x tale per ogni $i < k$

$$x \equiv a_i \mod p_i$$

Tutte le soluzioni x sono congruenti modulo $\pi = p_1 \cdot p_2 \cdot \ldots p_k$.
$x \equiv y(\mod p_i)$ per ogni $1 \leq i \leq k$ se e solo se $x \equiv y(\mod \pi)$.
Nel seguito tale unico numero sarà indicato con
$\mathrm{CRT}_{p_1,\ldots p_k}(a_1, \ldots, a_k)$

Schema Simmetrico

Idea base:

dati $p_0, \ldots, p_{l-1}$ interi coprimi allora i messaggi $m_0, \ldots, m_{l-1} \in \mathbb{Z}_2$ possono essere cifrati contemporaneamente come:

$$c = q \cdot \prod_{i=0}^{l-1} p_i + \mathrm{CRT}_{p_0,\ldots p_{l-1}}(2r_0 + m_0, \ldots, 2r_{l-1} + m_{l-1}).$$

Schema Simmetrico

Idea base:

dati $p_0, \ldots, p_{l-1}$ interi coprimi allora i messaggi $m_0, \ldots, m_{l-1} \in \mathbb{Z}_2$ possono essere cifrati contemporaneamente come:

$$c = q \cdot \prod_{i=0}^{l-1} p_i + \mathrm{CRT}_{p_0,\ldots p_{l-1}}(2r_0 + m_0, \ldots, 2r_{l-1} + m_{l-1}).$$

Per decifrare basta

$$m_i = [c \mod p_i]_2$$

Schema Simmetrico

Idea base:

dati $p_0, \ldots, p_{l-1}$ interi coprimi allora i messaggi $m_0, \ldots, m_{l-1} \in \mathbb{Z}_2$ possono essere cifrati contemporaneamente come:

$$c = q \cdot \prod_{i=0}^{l-1} p_i + \mathrm{CRT}_{p_0,\ldots p_{l-1}}(2r_0 + m_0, \ldots, 2r_{l-1} + m_{l-1}).$$

Per decifrare basta

$$m_i = [c \mod p_i]_2$$

Si ottiene il bit esatto per ogni $i < l$ in quanto

- $\left[q \cdot \prod_{i=0}^{l-1} p_i\right]_{p_i} = 0$
- $\left[\mathrm{CRT}_{p_0,\ldots p_{l-1}}(2r_0 + m_0, \ldots, 2r_{l-1} + m_{l-1})\right]_{p_i} = 2r_i + m_i$ per definizione di CRT.

Schema asimmetrico

$$c = \left[\sum_{i=0}^{l-1} m_i \cdot y_i + 2r + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Schema asimmetrico

$$c = \left[\sum_{i=0}^{l-1} m_i \cdot y_i + 2r + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Dove:

- $x_i \mod p_j = r_{i,j}$
- $y_i \mod p_j = \delta_{i,j} + 2t_{i,j}$
- $\delta_{i,j} = 1$ se $i = j$,
  $\delta_{i,j} = 0$ se $i \neq j$.
- $x_0 = p_0 \cdot p_1 \ldots p_{l-1} \cdot q_0 = \pi \cdot q_0$

per ogni j $[c \mod p_j]_2 = m_i$ come richiesto.

Schema asimmetrico

$$c = \left[\sum_{i=0}^{l-1} m_i \cdot y_i + 2r + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Problema:

La sicurezza dello schema originale DGHV si basa sul fatto che viene aggiunto un ulteriore errore random $2r$.
In questo caso invece l'errore $2r$ è lo stesso termine randomico modulo ciascun p_i.
Perché il sistema sia semanticamente sicuro ogni p_i dovrebbe avere un termine randomico indipendente dagli altri.

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$\tag{2}$$

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$\tag{2}$$

Con

- $x_0 = p \cdot q_0$
- $x_i = pq_i + r_i$
- $q = 2 \sum q_i \mod q_0$ e
  $R = 2 \sum r_i$

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$m + p \cdot q + 2R \tag{2}$$

Con

- $x_0 = p \cdot q_0$
- $x_i = pq_i + r_i$
- $q = 2 \sum q_i \mod q_0$ e
  $R = 2 \sum r_i$

Per la sicurezza q e R devono essere random e indipendentemente distribuiti.

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$m + p \cdot q + 2R \tag{2}$$

Per la sicurezza q e R devono essere random e indipendentemente distribuiti.
Per la sicurezza vorremmo poter ridurre R modulo un qualche intero ω

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$m + p \cdot q + 2R \tag{2}$$

Per la sicurezza q e R devono essere random e indipendentemente distribuiti.
Per la sicurezza vorremmo poter ridurre R modulo un qualche intero ω
Invece di ridurre R mod ω si aggiunge un multiplo random molto grande di ω in fase di cifratura.

Variante DGHV per un solo bit:

$$c = \left[m + 2 \sum_{i \in S} x_i \right]_{x_0} = \tag{1}$$

$$m + p \cdot q + 2R \tag{2}$$

Per la sicurezza q e R devono essere random e indipendentemente distribuiti.
Per la sicurezza vorremmo poter ridurre R modulo un qualche intero ω
Invece di ridurre $R \mod \omega$ si aggiunge un multiplo random molto grande di ω in fase di cifratura. Si aggiunge un intero Π alla chiave pubblica tale che $\Pi \mod p = \omega$

Nuova cifratura

Nuovo sistema di cifratura di un bit:

$$c = \left[m + 2b \cdot \Pi + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Per qualche largo intero random b.

Nuova cifratura

Nuovo sistema di cifratura di un bit:

$$c = \left[m + 2b \cdot \Pi + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Per qualche largo intero random b.
In questo modo:
$c \mod p = R + b \cdot \omega$

Nuova cifratura

Nuovo sistema di cifratura di un bit:

$$c = \left[m + 2b \cdot \Pi + 2 \sum_{i \in S} x_i \right]_{x_0}$$

Per qualche largo intero random b.
In questo modo:
$c \mod p = R + b \cdot \omega$
$R + b \cdot \omega$ questo se $R = 2 \sum_{i \in S} r_i$ non é piú grande di ω allora R mod ω sottrae un piccolo multiplo di ω, irrilevante rispetto a $b \cdot \omega$.

chiave segreta:

l interi primi $p_0, \ldots, p_{l-1}$.

chiave pubblica:

- $x_0 = q_0 \cdot \pi$, q_0 non ha fattori primi piú piccoli di 2^{λ^2}
- Interi x_i, y_i, Π_i distribuiti in $[0, x_0)$. Tali che per $0 \leq j < l$

$$
\begin{array}{lll}
0 \leq i \leq \tau & x_i \mod p_j = 2r_{ij} & r_{ij} \in \mathbb{Z} \cap (-2^{\rho_1 - 1}, 2^{\rho_1 - 1}) \\
0 \leq i \leq l-1 & y_i \mod p_j = 2t_{ij} + \delta_{ij} & t_{ij} \in \mathbb{Z} \cap (-2^{\rho}, 2^{\rho}) \\
0 \leq i \leq l-1 & \Pi_i \mod p_j = 2\omega_{ij} + \delta_{ij} \cdot 2^{\rho_1 + 1} & \omega_{ij} \in \mathbb{Z} \cap (-2^{\rho}, 2^{\rho})
\end{array}
$$

cifratura

Per cifrare il vettore $\mathbf{m} \in \{0,1\}^l$. scelti due vettori di elementi casuali

- $\mathbf{b} = (b_i)_{1 \leq i \leq \tau} \in (-2^{\alpha}, 2^{\alpha})^{\tau}$
- $\mathbf{d} = (d_i)_{1 \leq i \leq l-1} \in (-2^{\alpha_1}, 2^{\alpha_1})^{l}$

$$c = \left[\sum_{i=0}^{l-1} m_i \cdot y_i + \sum_{i=0}^{l-1} d_i \cdot \Pi_i + \sum_{i=0}^{\tau} b_i \cdot x_i \right]_{x_0}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

Verifichiamo:

$$[c]_{p_j} =$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

Verifichiamo:

$$[c]_{p_j} =$$

$$\left[\left[\sum_{i=0}^{l-1} m_i \cdot y_i + \sum_{i=0}^{l-1} d_i \cdot \Pi_i + \sum_{i=0}^{\tau} b_i \cdot x_i \right]_{x_0} \right]_{p_j}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$\left[\sum_{i=0}^{l-1} m_i \cdot y_i + \sum_{i=0}^{l-1} d_i \cdot \Pi_i + \sum_{i=0}^{\tau} b_i \cdot x_i - K x_0 \right]_{p_j} \tag{3}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$\sum_{i=0}^{l-1} m_i \cdot (2t_{ij} + \delta_{ij} + K_1 p_j) + \tag{3}$$

$$\sum_{i=0}^{l-1} d_i \cdot (2\omega_{ij} + \delta_{ij} \cdot 2^{\rho_1+1} + K_2 p_j) + \tag{4}$$

$$\sum_{i=0}^{\tau} b_i \cdot (K_3 p_j + 2r_{ij}) - K q_0 \pi \tag{5}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$\sum_{i=0}^{l-1} m_i \cdot (2t_{ij} + \delta_{ij}) \tag{3}$$

$$+p_j \left[\sum m_i K_1 + \sum d_i K_2 + \sum b_i K_3 - K \frac{\pi}{p_j} q_0) \right] + \tag{4}$$

$$\sum_{i=0}^{l-1} d_i \cdot (2\omega_{ij} + \delta_{ij} \cdot 2^{\rho_1+1}) + \sum_{i=0}^{\tau} b_i \cdot 2r_{ij} \tag{5}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$\sum_{i=0}^{l-1} m_i \cdot (2t_{ij} + \delta_{ij}) + \tag{3}$$

$$\sum_{i=0}^{l-1} d_i \cdot (2\omega_{ij} + \delta_{ij} \cdot 2^{\rho_1+1}) + \sum_{i=0}^{\tau} b_i \cdot 2r_{ij} \tag{4}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$2\sum_{i=0}^{l-1} m_i \cdot t_{ij} + \sum_{i=0}^{l-1} m_i \delta_{ij} + \tag{3}$$

$$2\sum_{i=0}^{l-1} d_i \cdot \omega_{ij} + 2\sum_{i=0}^{l-1} d_i \cdot \delta_{ij} \cdot 2^{\rho_1} + \tag{4}$$

$$2\sum_{i=0}^{\tau} b_i \cdot r_{ij} (\mod 2) \tag{5}$$

Decifratura

si ottiene il vettore $\mathbf{m} = (m_0, \ldots, m_{l-1})$
con

$$m_j = [c]_{p_j} \mod 2$$

$$\sum_{i=0}^{l-1} m_i \delta_{ij} = \begin{cases} m_j & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases}$$

Operazioni

dati c_1, c_2 testi cifrati allora

- Addizione $c_1 + c_2 \mod x_0$
- Moltiplicazione $c_1 \cdot c_2 \mod x_0$

Parametri

- l numero bit cifrati
- λ parametro di sicurezza.
- $\rho \geq 2\lambda$ rumore sulle chiavi pubbliche.
- α rumore nel vettore **b**
- $\eta \geq \rho \cdot \lambda$ bit chiave segreta.
- $\gamma = \eta \cdot \lambda$ bit del testo cifrato
- $\rho_1 \geq \rho + \lambda$, $\alpha_1 \geq \alpha + \lambda$ bit rumore di **y** e bit di ciascun elemento del vettore **d**.
- $\alpha \cdot \tau \geq \gamma + \lambda$
- $\tau \geq l \cdot (\rho_1 + 2) + \lambda$

Parametri

Possibili scelte

- $\rho = 2\lambda$ rumore in **x**.
- $\alpha = \lambda^2$ rumore nel vettore **b**
- $\eta = 2\lambda^2$ bit chiave segreta.
- $\gamma = \lambda^5$ bit del testo cifrato
- $\rho_1 = 3\lambda$, $\alpha_1 = \lambda^2$ bit rumore di **y** e bit di ciascun elemento del vettore **d**.
- $l = \lambda^2$
- $\tau = \lambda^3$

Bibiografia

Cheon, Jung Hee and Coron, Jean-Sébastien and Kim, Jinsu and Lee, Moon Sung and Lepoint, Tancrede and Tibouchi, Mehdi and Yun, Aaram *Batch fully homomorphic encryption over the integers*, Advances in Cryptology–EUROCRYPT 2013 pages 315–335, year 2013, Springer

Van Dijk, M. and Gentry, C. and Halevi, S. and Vaikuntanathan, V., *Fully homomorphic encryption over the integers*, Advances in Cryptology–EUROCRYPT 2010, pages 24-43, year 2010, Springer

Grazie per l'attenzione.