Cryptographic Protocols

Chapter 4

Homomorphic Encryption

4.1 Grand Goal

Computation with encrypted data. No interaction between user and cloud. User holds sk (private key).
Intriguing problem, first posed in 1978

4.2 Single-Homomorphic Encryption

Encryption scheme supports an operation $\oplus$ such that:

$$\textsc{Enc}(x) \ \oplus \ \textsc{Enc}(y) \ = \ \textsc{Enc}(x+y)$$

OR an operation $\otimes$ such that:

$$\textsc{Enc}(x) \ \otimes \ \textsc{Enc}(y) \ = \ \textsc{Enc}(x \cdot y)$$

where $x, y \in \mathbb{GF}(2)$ or $\mathbb{GF}(p)$
→ such schemes exist and are efficient
→ Additively kommutative ElGamal; Paillier scheme

4.3 Fully-Homomorphic Encryption

Encryption scheme supports an operation $\oplus$ such that:

$$\textsc{Enc}(x) \ \oplus \ \textsc{Enc}(y) \ = \ \textsc{Enc}(x+y)$$

AND an operation $\otimes$ such that:

$$\textsc{Enc}(x) \ \otimes \ \textsc{Enc}(y) \ = \ \textsc{Enc}(x \cdot y)$$

where $x, y \in \mathbb{GF}(2)$ or $\mathbb{GF}(p)$
→ exist since 2009, Gentry
→ not very practical

4.4 Examples

4.4.1 ElGamal-based Single-Homomorphic Encryption

Recall textbook ElGamal in $\mathbb{G} =< g >, \mid \mathbb{G} \mid= q$:

$\textsc{KeyGen}()$	$\textsc{Enc}(m, y)$	$\textsc{Dec}(x, (R, C))$
$x \leftarrow \mathbb{Z}_q$	$r \leftarrow \mathbb{Z}_q$	$\hat{m} \leftarrow c/R^x$
$y \leftarrow g^x$	$R \leftarrow g^r$	return $\hat{m}$
return (y, x)	$C \leftarrow m \cdot y^r$	
	return (R, C)	

- already multiplication homomorphic
- but mapping of integers to and from elements of $\mathbb{G}$ can be hard
- but addition is preferred

4.4.2 Additively homomorphic ElGamal

- use textbook ElGamal
- works only for small plaintexts, numbers e.g. from 0 to max

$\textsc{KeyGen}()$	AH-$\textsc{Enc}(y, a)$	AH-$\textsc{Dec}(x, (R, C))$
$x \leftarrow \mathbb{Z}_q$	return $\textsc{Enc} y, g^a$	$h \leftarrow \textsc{Dec}(x, (R, C))$
$y \leftarrow g^x$		// brute-force search for $i = 0, 1, ..., max$
return (y, x)		if $g^i = h$ then:
		return i
		return $\textsc{Error}$

$$\begin{aligned} \text{AH-}\textsc{Enc}(y, a) \oplus \text{AH-}\textsc{Enc}(y, b) &= (R_a, C_a) \oplus (R_b, C_b) \\ &= (R_a \cdot R_b, C_a \cdot C_b) \\ &= \text{AH-}\textsc{Enc}(y, (a + b) \ mod \ q) \end{aligned}$$

... $q \approx 2^{256}$

... $max \ \approx 10^6$

... $a, b \in [-max/2, \ max/2]$ using $a' = q + a$ for $a < 0$

4.5 Voting protocol using Additional Homomorphic Encryption

1. Parties $P_1, ..., P_n$
2. Each party P_i votes $v_i \in \{-1, 1\}$
3. One authority $\mathbb{A}$

1. $\mathbb{A}$ generates
   $(pk, sk) \leftarrow \textsc{KeyGen}()$
   $\mathbb{A}$ sends pk to $P_1, ..., P_n$

2. $\mathbb{A}$ computes
   $c_0 \leftarrow \textsc{Enc}(pk, 0)$
   and sends c_0 to P_1
   for $i = 1, ..., n$ do
   P_i receives c_{i-1}
   P_i computes
   $c_i \leftarrow c_{i-1} \oplus \textsc{Enc}(y, v_i)$
   P_i sends c_i to P_{i+1} // P_n sends to $\mathbb{A}$
3. $\mathbb{A}$ receives c_n from P_n computes
   $z \leftarrow \textsc{Dec}(x, c_n)$ // $z = \sum_{i=1}^{n} v_i$
   $\mathbb{A}$ publishes z

Remarks:

- sending must use secure channels
- not robust (against malicious $\widetilde{P}_i$)
  a) $\widetilde{P}_i$ encrypt $+n$
  b) $\widetilde{P}_i$ refuse to send c_i
  c) $\mathbb{A}$ refuses to decrypt
- defenses exist for all these:
  a) use zero-knowledge proofs
  b) use public bulletin board for communication
  c) distributed implementation of $\mathbb{A}$ using a group of admins
- Helios implements most of this (heliosvoting.org)

4.6 Zero-Knowledge Proofs

- How to prove a statement is true without any more information
- How to prove knowledge of "password" without giving information about it?
- Many application in cryptographic protocols
- Goldwasser & Micali received Turing Award for it

4.6.1 Two kinds of proofs

- Proofs for statements
  - Griven a Boolean formula Ψ in n variables, there exists an assignment s.t. $\Psi \equiv \textsc{true}$
  - Given two graphs $\mathbb{G}_0$ and $\mathbb{G}_1$, they are isomorphic
  - Given a graph $\mathbb{G}$, there exists a Hamiltonian circuit in $\mathbb{G}$
  - Given a graph $\mathbb{G}$, there is a 3-coloring of $\mathbb{G}$
- Proofs of knowledge
  - Given y (a public key DH$\sim$) "I know" x such that $y = g^x$
  - Given $h \in \{0,1\}^k$, "I know" an x s.t. $\mathbb{H}(x) = h$ for some one-way function $\mathbb{H}$
  - Given N, "I know" P, Q s.t. $N = P \cdot Q$

4.7 Model (Proofs of Statement)

Requirements

1. **Completeness:** If statement S holds, prover $\mathbb{P}$ correct and $\mathbb{V}$ correct, then $\mathbb{V}$ accepts.
2. **Soundness:** If S is FALSE, then (honest) $\mathbb{V}$ will reject with at least a constant probability (no matter what the malicious $\mathbb{P}$ does)
3. **Zero-Knowledge:** $\mathbb{V}$ learns only that S holds (and not more).
(V could also have simulated the whole protocol with itself)

4.8 ZKP for Graph Isomorphism

- $\mathbb{P}$ and $\mathbb{V}$ are given two graphs $G_0 = (V, E_0)$ and $G_1 = (V, E_1)$.
- $\mathbb{P}$ knows an isomorphism betwenn G_0 and G_1, i.e. bijective function:

$$f : V \to V$$

s.t. $\forall v, w \in V : \ (v, w) \in E_0 \ \Leftrightarrow \ (f(v), f(w)) \in E_1$

Protocol