CSCI 1515 Applied Cryptography

This Lecture:

- PSI-Sum (Continued)
- Information Theoretic MPC (BGW)
- Introduction to FHE
- SWHE over Integers

Private Set Intersection (PSI)

Input: $X = \{x_1, x_2, \cdots, x_n\}$

$V = \{v_1, v_2, \cdots, v_n\}$

Input: $Y = \{y_1, y_2, \cdots, y_n\}$

PSI: $f(X, Y) = X \cap Y$

PSI-CA: $f(X, Y) = |X \cap Y|$

PSI-SUM: $f((X, V), Y) = |X \cap Y|, \sum_{i : x_i \in Y} v_i$

PSI-CA

PSI-CA: $f(X,Y) = |X \cap Y|$

Input: $X = \{x_1, x_2, \cdots, x_n\}$

$k_A \xleftarrow{\$} \mathbb{Z}_q$

Input: $Y = \{y_1, y_2, \cdots, y_n\}$

$k_B \xleftarrow{\$} \mathbb{Z}_q$

$$\xleftarrow{H(Y)^{k_B} := \{H(y_1)^{k_B}, \cdots, H(y_n)^{k_B}\}}$$

$$\xrightarrow{H(X)^{k_A}, \{H(Y)^{k_A \cdot k_B}\}}$$

Shuffle

$H(X)^{k_A \cdot k_B} \cap H(Y)^{k_A \cdot k_B}$

$\downarrow$

$|X \cap Y|$

PSI-SUM?

PSI-SUM: $f((X, V), Y) = |X \cap Y|, \sum_{i: x_i \in Y} v_i$

Alice

Bob

Pallier Encryption Scheme

Additively Homomorphic Enc

Input: $X = \{x_1, x_2, \cdots, x_n\}$

$V = \{v_1, v_2, \cdots, v_n\}$

$k_A \xleftarrow{\$} \mathbb{Z}_q$

Input: $Y = \{y_1, y_2, \cdots, y_n\}$

$k_B \xleftarrow{\$} \mathbb{Z}_q$

$H(Y)^{k_B} := \{H(y_1)^{k_B}, \cdots, H(y_n)^{k_B}\}$

$H(X)^{k_A}, \{H(Y)^{k_A \cdot k_B}\}$

Shuffle

$H(x_1)^{k_A}, H(x_2)^{k_A}, \cdots, H(x_n)^{k_A}$

$\text{Enc}(v_1), \text{Enc}(v_2), \cdots, \text{Enc}(v_n)$

$H(X)^{k_A \cdot k_B} \cap H(Y)^{k_A \cdot k_B}$

$\downarrow$

$|X \cap Y|$

$\text{Enc}\left(\sum_{i: x_i \in Y} v_i\right)$

Feasibility Results

Computational Security:

Semi-honest Oblivious Transfer (OT)

semi-honest MPC for any function with $t<n$

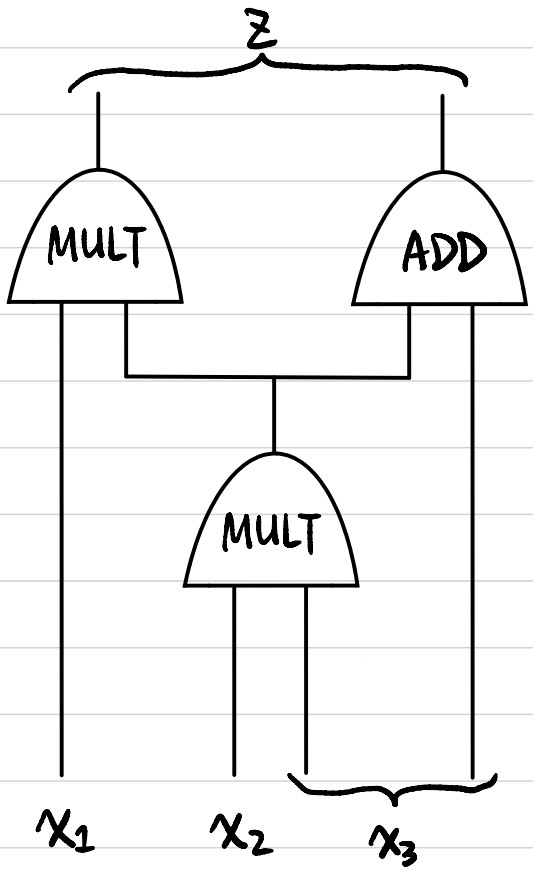
malicious MPC for any function with $t<n$

Information-Theoretic (IT) Security:

semi-honest/malicious MPC for any function with $t<n/2$ (honest majority)

↑ necessary

IT-MPC for any function with $t < n/2$ (BGW)



Throughout the protocol, we keep the invariant:

For each wire w:

If the value of the wire is $v^w \in \mathbb{F}$

the n parties hold a $(t+1)$-out-of-n secret share of v^w

Each party P_i holds a random share $v_i^w \in \mathbb{F}$ s.t.

1) Any $(t+1)$ shares can jointly recover v^w

2) Any t shares information theoretically hide v^w

Shamir Secret Sharing Scheme

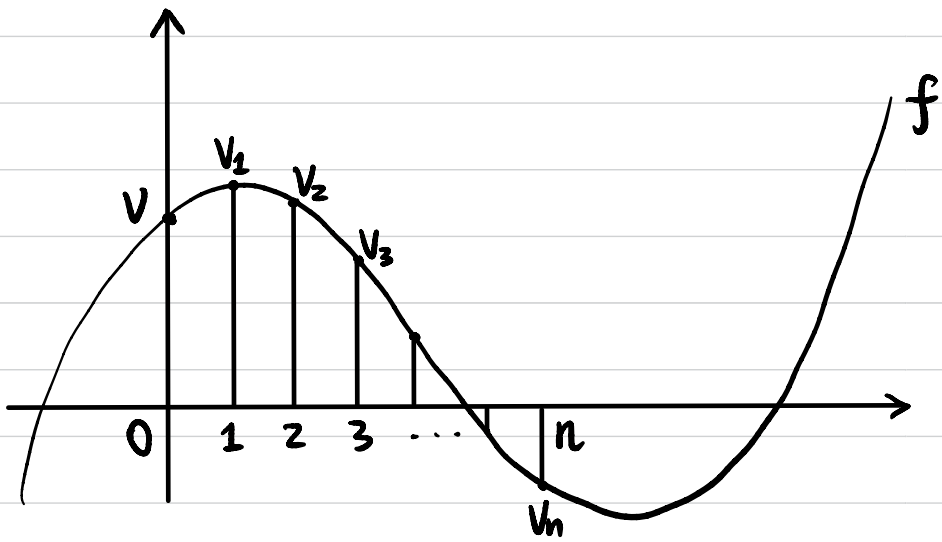
- Secret value $v \in \mathbb{F}$

- **Share**: Sample a random degree-t polynomial $f: \mathbb{F} \to \mathbb{F}$ s.t. $f(0) = v$ ← How?

$$f(x) = a_t x^t + a_{t-1} x^{t-1} + \cdots + a_1 x_1 + a_0$$

($a_t, a_{t-1}, \ldots, a_1 \in \mathbb{F}$; $a_0 = v$)

Party P_i's share $v_i = f(i)$



t shares $\overset{?}{\Rightarrow} v$

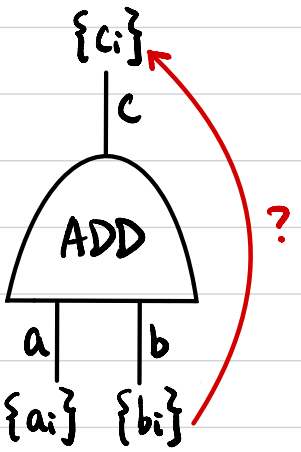
- **Reconstruct**:

Given $(t+1)$ shares: $(\alpha_1, v_{\alpha_1} = f(\alpha_1)), \cdots, (\alpha_{t+1}, v_{\alpha_{t+1}} = f(\alpha_{t+1}))$

How to recover v?

IT-MPC for any function with $t < n/2$ (BGW)

ADD gates:



GIVEN: Deg-t poly $f_a(\cdot)$ s.t. $f_a(0) = a$. Shares $a_i = f_a(i)$

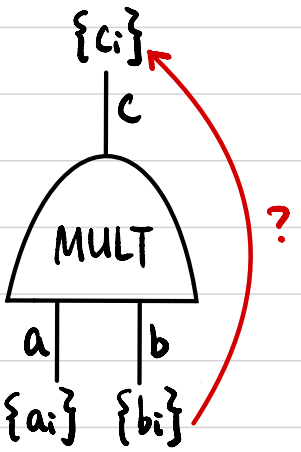
Deg-t poly $f_b(\cdot)$ s.t. $f_b(0) = b$. Shares $b_i = f_b(i)$

WANT: Deg-t poly $f_c(\cdot)$ s.t. $f_c(0) = a + b$. $f_c = f_a + f_b$

Shares $c_i = f_c(i) = f_a(i) + f_b(i) = a_i + b_i$

IT-MPC for any function with $t < n/2$ (BGW)

MULT gates:



GIVEN: Deg-t poly $f_a(\cdot)$ s.t. $f_a(0) = a$. Shares $a_i = f_a(i)$

Deg-t poly $f_b(\cdot)$ s.t. $f_b(0) = b$. Shares $b_i = f_b(i)$

WANT: Deg-t poly $f_c(\cdot)$ s.t. $f_c(0) = a \cdot b$. $f_c = ?$

Shares $c_i = f_c(i) = ?$

Attempt: deg-$2t$ poly

$f_c := f_a \cdot f_b$

$f_c(0) = f_a(0) \cdot f_b(0) = a \cdot b$

$c_i = f_c(i) = f_a(i) \cdot f_b(i) = a_i \cdot b_i$

Problem?

$f_c(0) = \alpha_1 \cdot c_1 + \alpha_2 \cdot c_2 + \cdots + \alpha_n \cdot c_n$

P_1 P_2 P_n

$(t+1)$-out-of-n

Fully Homomorphic Encryption (FHE)

$Enc(m_1)$ → $Enc(m_1 + m_2)$ ← $Enc(m_2)$

Additively Homomorphic

↑

Exponential ElGamal / Pallier

$Enc(m_1)$ → $Enc(m_1 \cdot m_2)$ ← $Enc(m_2)$

Multiplicatively Homomorphic

↑

RSA / ElGamal

Fully Homomorphic: Additively & Multiplicatively Homomorphic

Fully Homomorphic Encryption (FHE)

Def A (public-key) homomorphic encryption scheme

$\Pi = (\text{KeyGen}, \text{Enc}, \text{Dec}, \text{Eval})$ w.r.t. function family F:

- $(pk, sk) \leftarrow \text{KeyGen}(1^\lambda)$
- $ct \leftarrow \text{Enc}_{pk}(m) \qquad m \in \{0,1\}$
- $m \leftarrow \text{Dec}_{sk}(ct)$
- $ct_f \leftarrow \text{Eval}(f, ct_1, \cdots, ct_n) \qquad f: \{0,1\}^n \to \{0,1\}$

- **Correctness:** $ct_i \leftarrow \text{Enc}_{pk}(m_i) \quad \forall i \in [n]$,

  $\forall f \in F,\ ct_f \leftarrow \text{Eval}(f, ct_1, \cdots, ct_n)$

  $\text{Dec}_{sk}(ct_f) = f(m_1, \cdots, m_n)$

- **(CPA) Security:** $(pk, \text{Enc}_{pk}(m_0)) \overset{c}{\approx} (pk, \text{Enc}_{pk}(m_1))$.

Missing Requirement ?

Fully Homomorphic Encryption (FHE)

Def A (public-key) homomorphic encryption scheme $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ w.r.t. function family $\mathcal{F}$:

- $(pk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda)$
- $ct \leftarrow \mathsf{Enc}_{pk}(m) \qquad m \in \{0,1\}$
- $m \leftarrow \mathsf{Dec}_{sk}(ct)$
- $ct_f \leftarrow \mathsf{Eval}(f, ct_1, \cdots, ct_n) \qquad f: \{0,1\}^n \to \{0,1\}$

- So far, it's trivial to construct FHE under the definition.

  $\mathsf{Eval}(f, ct_1, \cdots, ct_n) = (f, ct_1, \cdots, ct_n)$

- **Compactness:** $|ct_f| \leq$ fixed poly (λ)
  Independent of circuit size of f.

- If $\mathcal{F}$ is the set of **all** poly-sized Boolean circuits, then Π is **fully** homomorphic.

Application: Outsourcing Storage & Computation

Server

Client

Data x

Key sk

$ct \leftarrow Enc(x)$

ct

f

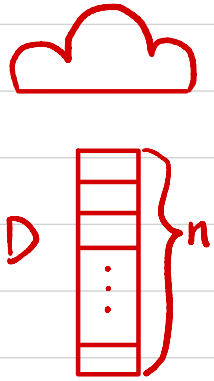
$ct' \leftarrow Eval(f, ct)$

ct'

$f(x) \leftarrow Dec_{sk}(ct')$

Application: Private Information Retrieval (PIR)

Server



1-out-of-n OT:

① Stronger security guarantee

② Communication O(n)

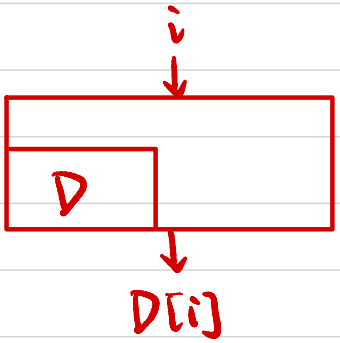
Client

WANT: D[i]

While hiding i against Server

$ct' \leftarrow Eval(f, ct)$

$f_D(i) = D[i]$



Communication o(n) ?

Application: Secure 2PC ?

Alice

Bob

$C(x, y)$

Input: x

Input: y

Key sk

$ct \leftarrow Enc(y)$

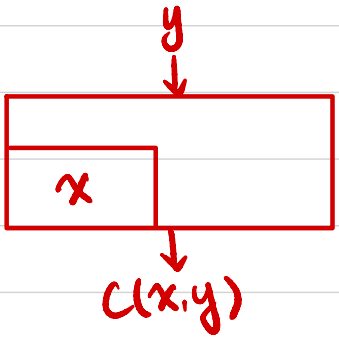
ct

$ct' \leftarrow Eval(f, ct)$

ct'

$f_x(y) = C(x, y)$

$f(y) \leftarrow Dec_{sk}(ct')$



FHE Constructions

Step 1: Somewhat Homomorphic Encryption (SWHE)

- over Integers
- from LWE (GSW)
- from RLWE (BFV)

Step 2: Bootstrapping

SWHE over Integers

Attempt 1 (secret-key)

- secret key: odd number p ← Why odd?
- $\mathrm{Enc}(m)$: $m \in \{0,1\}$
  Sample a random q.
  Output $ct = p \cdot q + m$
  Encryption of 0 is a multiple of p.
- $\mathrm{Dec}(ct)$: $ct \bmod p$
- Eval ADD: $ct \leftarrow ct_1 + ct_2$
  Eval MULT: $ct \leftarrow ct_1 \cdot ct_2$

(CPA) Security?

$\mathrm{GCD}(p \cdot q_1, p \cdot q_2, \cdots) = p$

SWHE over Integers

Attempt 2 (secret-key)

- secret key: odd number p
- $\mathrm{Enc}(m)$: $m \in \{0,1\}$

  Sample a random q. Sample a random $e \ll p$

  Output $ct = p \cdot q + m + 2e$

  Encryption of 0 is small and even modulo p.
- $\mathrm{Dec}(ct)$: $[ct \bmod p] \bmod 2$
- Eval ADD: $ct \leftarrow ct_1 + ct_2$

  Eval MULT: $ct \leftarrow ct_1 \cdot ct_2$

- Approximate GCD Problem:

  Given poly-many $\{x_i = p \cdot q_i + s_i\}$, output p.

Example parameters: $p \sim 2^{O(\lambda^2)}$, $q_i \sim 2^{O(\lambda^5)}$, $s_i \sim 2^{O(\lambda)}$

Best known attacks require 2^λ time.