

CSCI 1515 Applied Cryptography

This Lecture:

- Yao's Garbled Circuit
- Putting it All Together: Semi-Honest 2PC
- Optimizations of Garbled Circuits
- Cut-and-Choose for Garbled Circuits

Secure Two-Party Computation (2PC)

Alice



x



Bob



y

$$z = f(x, y)$$

Allowed adversarial behavior:

- **Semi-honest:** Follow the protocol description honestly, but try to extract more information by inspecting transcript.
- **Malicious:** Can deviate arbitrarily from the protocol description.

Semi-honest OT
(OT protocol that's secure
against semi-honest adv.)

Yao's Garbled
Circuit

GMW

Semi-honest ZPC
for any function

Semi-honest MPC
for any function

cut-and-choose
with commitments

GMW Compiler
with ZKP

Malicious ZPC
for any function

Malicious MPC
for any function

Oblivious Transfer (OT)

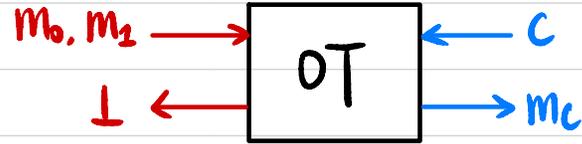
Sender

Input: $m_0, m_1 \in \{0, 1\}^l$

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

$$k_0 := H(B^a)$$

$$k_1 := H\left(\left(\frac{B}{A}\right)^a\right)$$



Receiver

Input: $c \in \{0, 1\}$

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

$$\xrightarrow{A = g^a}$$

$$\xleftarrow{B = g^b \cdot A^c}$$

$$\xrightarrow{\begin{array}{l} ct_0 \leftarrow \text{Enc}_{k_0}(m_0) \\ ct_1 \leftarrow \text{Enc}_{k_1}(m_1) \end{array}}$$

Output:

$$k_c := H(A^b) = H(g^{ab})$$

$$m_c := \text{Dec}_{k_c}(ct_c)$$

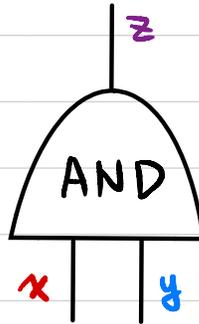
Example: Private Dating

Alice



$x \in \{0, 1\}$

$$f(x, y) = x \wedge y$$



Bob



$y \in \{0, 1\}$

Truth Table:

$$x=0 \quad y=0 \Rightarrow z=0$$

$$x=0 \quad y=1 \Rightarrow z=0$$

$$x=1 \quad y=0 \Rightarrow z=0$$

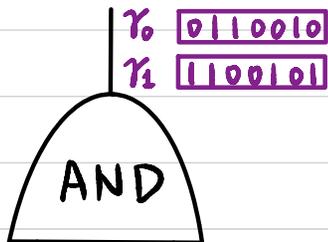
$$x=1 \quad y=1 \Rightarrow z=1$$

Example: Private Dating

Garbler



$x \in \{0, 1\}$



$\alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1 \leftarrow \{0, 1\}^n$
(labels)

Evaluator



$y \in \{0, 1\}$

Garbled Gate

$Enc_{\alpha_0}(Enc_{\beta_0}(\gamma_0))$
$Enc_{\alpha_0}(Enc_{\beta_1}(\gamma_0))$
$Enc_{\alpha_1}(Enc_{\beta_0}(\gamma_0))$
$Enc_{\alpha_1}(Enc_{\beta_1}(\gamma_1))$

Garbled Truth Table:

α_0	β_0	\Rightarrow	γ_0
α_0	β_1	\Rightarrow	γ_0
α_1	β_0	\Rightarrow	γ_0
α_1	β_1	\Rightarrow	γ_1

Evaluator only gets α_x & β_y
 \Rightarrow Which γ label(s)?

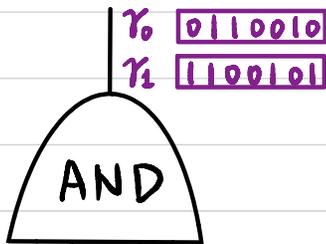
Example: Private Dating

Evaluator only gets α_x & β_y

Garbler

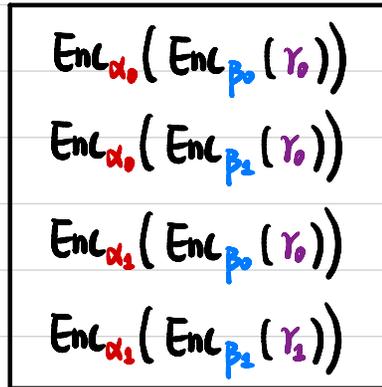


$X \in \{0,1\}$



$\alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1 \leftarrow \{0,1\}^n$
(labels)

Garbled Gate



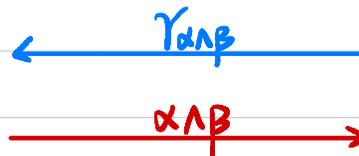
Input label for x : α_x

Evaluator



$y \in \{0,1\}$

Input label for y (β_y)?



Which ciphertext to decrypt?

Arbitrary Function \rightarrow Represent it as a Boolean circuit

Alice

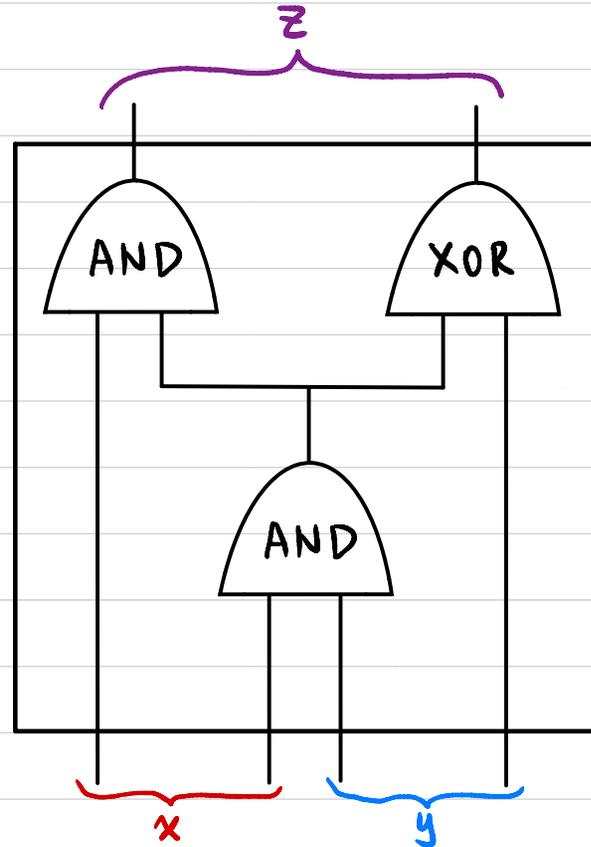


$x \in \{0,1\}^2$

Bob



$y \in \{0,1\}^2$

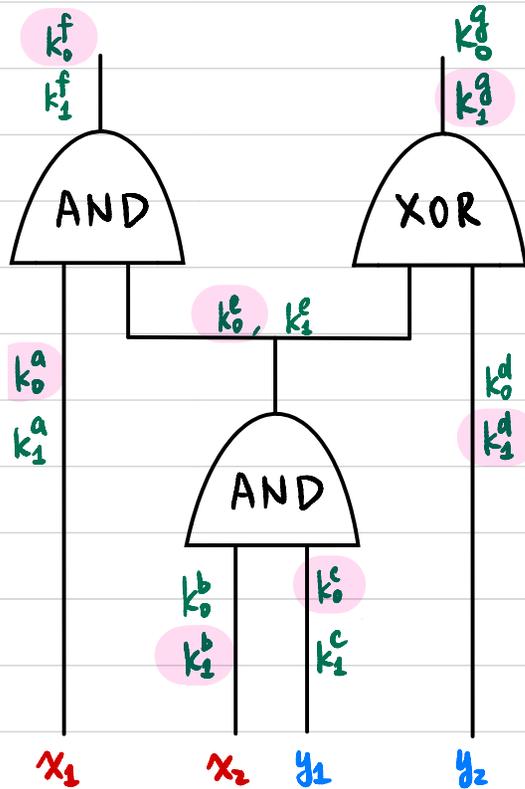


$$z = f(x, y)$$

Yao's Garbled Circuit

Evaluator gets one label per wire

$Enc_{k_0^a} (Enc_{k_0^e} (k_0^f))$
 $Enc_{k_0^a} (Enc_{k_2^e} (k_0^f))$
 $Enc_{k_1^a} (Enc_{k_0^e} (k_0^f))$
 $Enc_{k_1^a} (Enc_{k_1^e} (k_1^f))$



$Enc_{k_0^e} (Enc_{k_0^d} (k_0^g))$
 $Enc_{k_0^e} (Enc_{k_2^d} (k_1^g))$
 $Enc_{k_1^e} (Enc_{k_0^d} (k_1^g))$
 $Enc_{k_1^e} (Enc_{k_1^d} (k_0^g))$

Each label $\leftarrow \{f_{0,2}^j\}^{\lambda}$

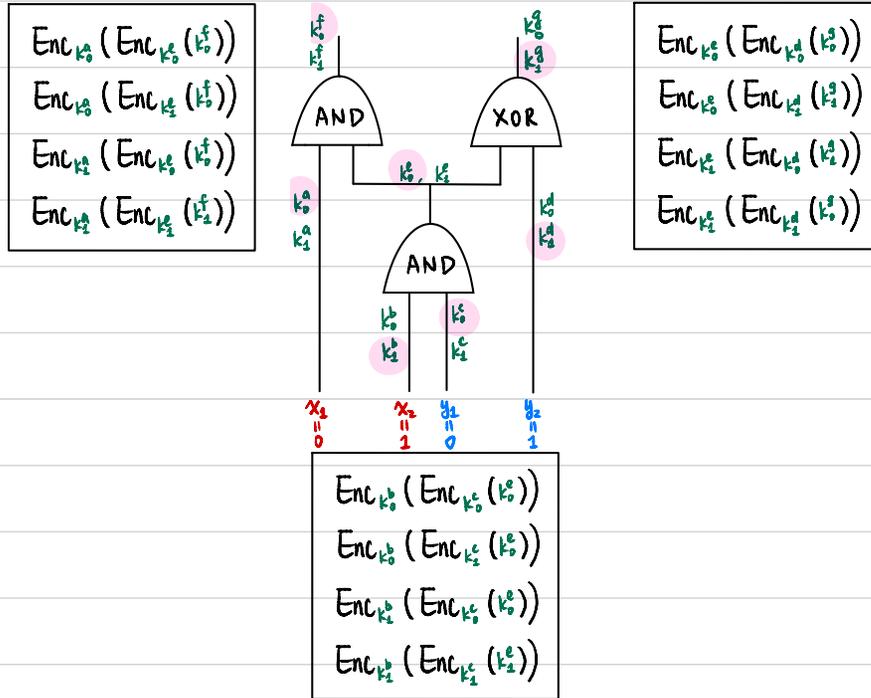
$Enc_{k_0^b} (Enc_{k_0^c} (k_0^e))$
 $Enc_{k_0^b} (Enc_{k_2^c} (k_0^e))$
 $Enc_{k_1^b} (Enc_{k_0^c} (k_0^e))$
 $Enc_{k_1^b} (Enc_{k_1^c} (k_1^e))$

Secure 2PC

Evaluator gets one label per wire

Alice (Garbler)

$x \in \{0,1\}^2$



Bob (Evaluator)

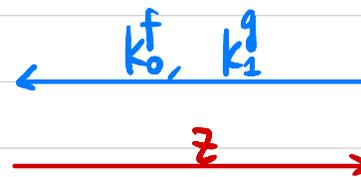
$y \in \{0,1\}^2$

Garbled Circuit
(Garbled Gates) →

Input labels for x →

Input labels for y ?

Which ciphertexts to decrypt?

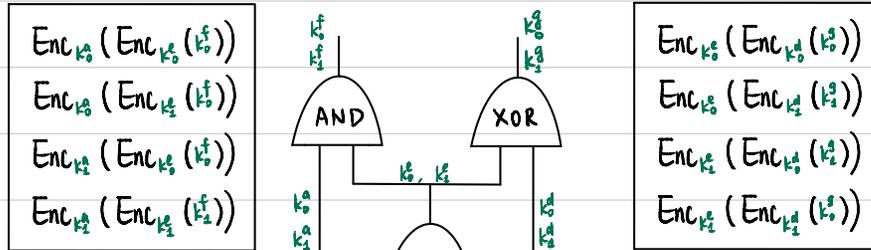


Putting it All Together: Semi-Honest ZPC

What could go wrong against malicious adversaries?

Alice (Garbler)

$x \in \{0,1\}^2$

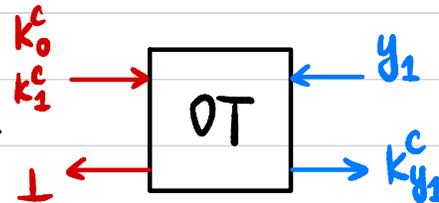


Bob (Evaluator)

$y \in \{0,1\}^2$

Garbled Circuit
(Garbled Gates)

Input labels for x



Input labels for y : OT

Output labels k^f, k^g

Output z

Communication cost?

Computation cost?

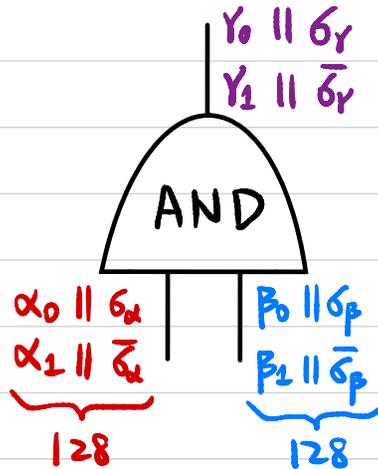
- $Enc_{k_a^0}(Enc_{k_b^0}(k_a^0))$
- $Enc_{k_a^0}(Enc_{k_b^1}(k_a^0))$
- $Enc_{k_a^1}(Enc_{k_b^0}(k_a^1))$
- $Enc_{k_a^1}(Enc_{k_b^1}(k_a^1))$

- Shuffle
- $H(k_a^0 || k_b^0) \oplus (k_a^0 || 0 \dots 0) \rightarrow$ garbage
 - $H(k_a^0 || k_b^1) \oplus (k_a^0 || 0 \dots 0) \rightarrow$ garbage
 - $H(k_a^1 || k_b^0) \oplus (k_a^0 || 0 \dots 0) \rightarrow k_a^0 || 0 \dots 0$
 - $H(k_a^1 || k_b^1) \oplus (k_a^1 || 0 \dots 0) \rightarrow$ garbage
- 256 128 128

Optimization 1: Point-and-Permute

4 · 128 bits / gate

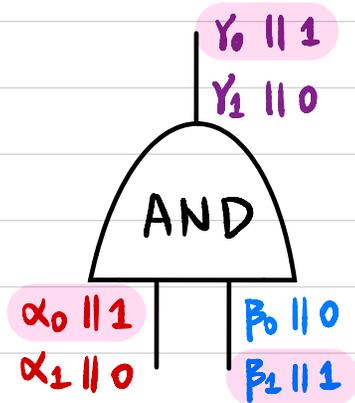
$\sigma_\alpha, \sigma_\beta, \sigma_\gamma \in \{0, 1\}$
(signal bits)



$$\begin{aligned} \sigma_\alpha \quad \sigma_\beta & : \text{Enc}_{\alpha_0} (\text{Enc}_{\beta_0} (Y_0 \parallel \sigma_\gamma)) \\ \sigma_\alpha \quad \bar{\sigma}_\beta & : \text{Enc}_{\alpha_0} (\text{Enc}_{\beta_1} (Y_0 \parallel \sigma_\gamma)) \\ \bar{\sigma}_\alpha \quad \sigma_\beta & : \text{Enc}_{\alpha_1} (\text{Enc}_{\beta_0} (Y_0 \parallel \sigma_\gamma)) \\ \bar{\sigma}_\alpha \quad \bar{\sigma}_\beta & : \text{Enc}_{\alpha_1} (\text{Enc}_{\beta_1} (Y_1 \parallel \bar{\sigma}_\gamma)) \end{aligned}$$

Example:

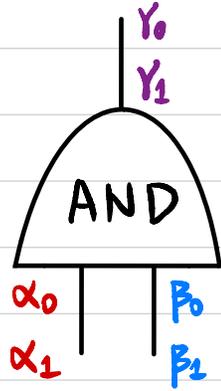
$\sigma_\alpha = 1 \quad \sigma_\beta = 0 \quad \sigma_\gamma = 1$



$$\begin{aligned} 0 \quad 0 & : \text{Enc}_{\alpha_1} (\text{Enc}_{\beta_0} (Y_0 \parallel 1)) \\ 0 \quad 1 & : \text{Enc}_{\alpha_1} (\text{Enc}_{\beta_1} (Y_1 \parallel 0)) \\ 1 \quad 0 & : \text{Enc}_{\alpha_0} (\text{Enc}_{\beta_0} (Y_0 \parallel 1)) \\ 1 \quad 1 & : \text{Enc}_{\alpha_0} (\text{Enc}_{\beta_1} (Y_0 \parallel 1)) \end{aligned}$$

Optimization 2: Row Reduction

3 · 128 bits/gate



$$\cancel{H(\alpha_0 \parallel \beta_0) \oplus \gamma_0} := 0 \quad \gamma_0 := H(\alpha_0 \parallel \beta_0)$$

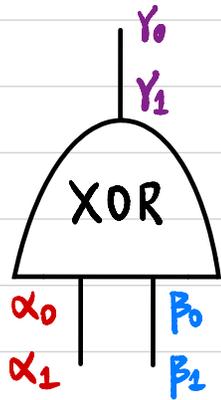
$$H(\alpha_0 \parallel \beta_1) \oplus \gamma_0$$

$$H(\alpha_1 \parallel \beta_0) \oplus \gamma_0$$

$$H(\alpha_1 \parallel \beta_1) \oplus \gamma_1$$

↑
Random Oracle

Optimization 3: Free XOR



Sample a global $\Delta \leftarrow \{0, 1\}^\lambda$

hidden to evaluator

$$\alpha_1 := \alpha_0 \oplus \Delta$$

$$\beta_1 := \beta_0 \oplus \Delta$$

$$\gamma_1 := \gamma_0 \oplus \Delta$$

$$\gamma_0 := \alpha_0 \oplus \beta_0$$

Other Optimizations:

- Half-gates: 2λ bits per AND gate + free XOR
- Three halves: $\sim 1.5\lambda$ bits per AND gate + free XOR

Cut-and-Choose for Garbled Circuits

Alice (Garbler)

$x \in \{0,1\}^2$

Bob (Evaluator)

$y \in \{0,1\}^2$

λ Garbled Circuits
(Garbled Gates)

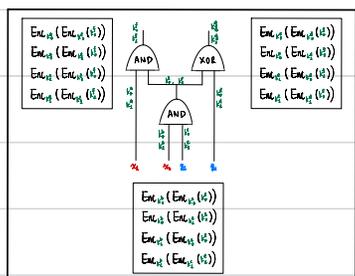
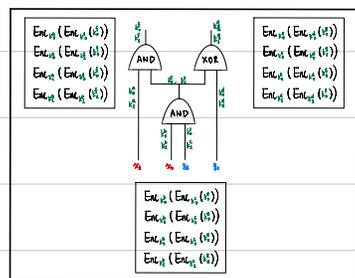
← Randomly pick $(\lambda-1)$ of them

Reveal all the labels
for the $(\lambda-1)$ GCs

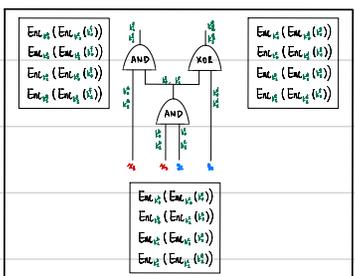
Verify correctness

↓
Continue with the remaining GC

If Alice generated 1 GC incorrectly,
 $\Pr[\text{Alice passes verification}] = ?$



⋮



Cut-and-Choose for Garbled Circuits

Alice (Garbler)

$x \in \{0,1\}^2$

Bob (Evaluator)

$y \in \{0,1\}^2$

2λ Garbled Circuits
(Garbled Gates)

← Randomly pick λ of them

Reveal all the labels
for the λ GCs

Verify correctness

Continue with the remaining λ GCs \Rightarrow Take majority

If Alice generated $\geq \frac{\lambda}{2}$ GCs incorrectly,
 $\Pr[\text{Alice passes all verification}] \leq ?$

