

CSCI 1515 Applied Cryptography

This Lecture:

- Yao's Garbled Circuits
- Construction of Oblivious Transfer

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

Yao's Garbled
Circuit

Semi-honest ZPC
for any function

Cut-and-choose
with commitments

Malicious ZPC
for any function

GMW

Semi-honest MPC
for any function

GMW Compiler
with ZKP

Malicious MPC
for any function

Oblivious Transfer (OT)

Sender



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

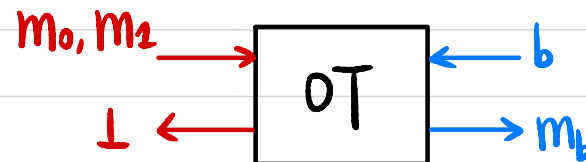
Output: $\perp$

Receiver



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

Output: m_b



Secure Two-Party Computation (2PC)

Alice

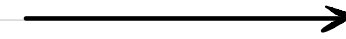


x

Bob



y



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

Adversary's Power

Allowed adversarial behavior:

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

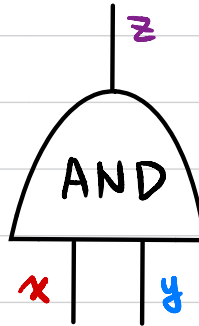
Example: Private Dating

Alice



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

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



Bob



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

Truth Table:

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

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

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

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

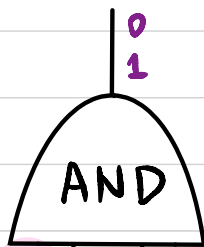
Example: Private Dating

Garbler



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

$x = 0$



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

Garbled Gate

$Enc_{\alpha_0}(Enc_{\beta_0}(0))$

$Enc_{\alpha_0}(Enc_{\beta_1}(0))$

$Enc_{\alpha_1}(Enc_{\beta_0}(0))$

$Enc_{\alpha_1}(Enc_{\beta_1}(1))$

Garbled Truth Table:

$\alpha_0 \quad \beta_0 \Rightarrow 0$

$\alpha_0 \quad \beta_1 \Rightarrow 0$

$\alpha_1 \quad \beta_0 \Rightarrow 0$

$\alpha_1 \quad \beta_1 \Rightarrow 1$

Evaluator



$y = 1$

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

Evaluator only gets α_x & β_y

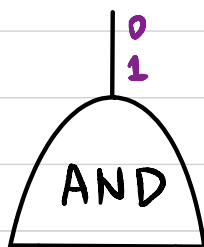
Example: Private Dating

Evaluator only gets α_x & β_y

Garbler



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



$\alpha_0, \alpha_1, \beta_0, \beta_1 \xleftarrow{\$} \{0, 1\}^\lambda$
(labels)

Garbled Gate

Shuffle

$Enc_{\alpha_0}(Enc_{\beta_0}(0))$
 $Enc_{\alpha_0}(Enc_{\beta_1}(0))$
 $Enc_{\alpha_1}(Enc_{\beta_0}(0))$
 $Enc_{\alpha_1}(Enc_{\beta_1}(1))$

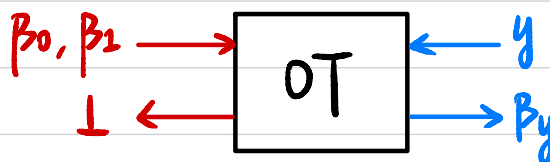
Input label for x : α_x

Evaluator



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

Input label for y (β_y)?



z



Arbitrary Function $\rightarrow$ Represent it as a Boolean circuit

Alice

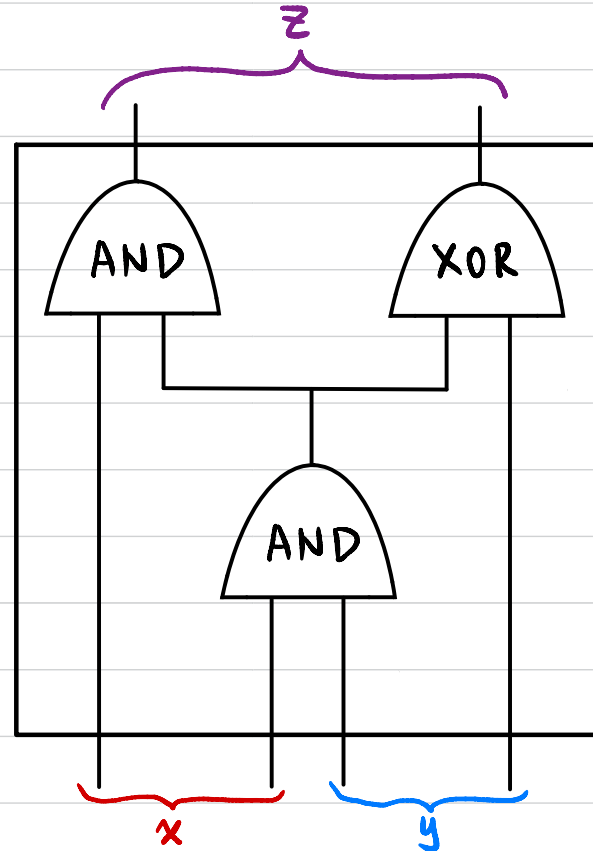


$x \in \{0,1\}^2$

Bob



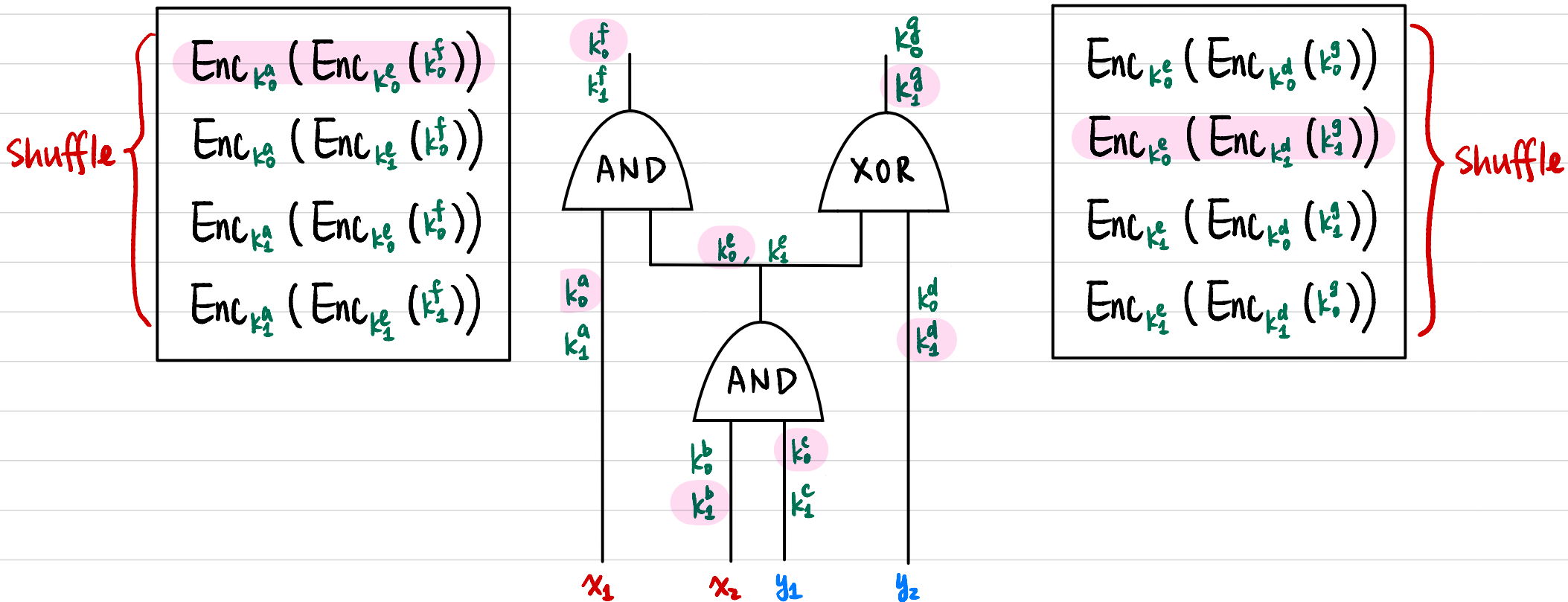
$y \in \{0,1\}^2$



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

Yao's Garbled Circuit

Evaluator gets one label per wire



Each label $\leftarrow \{0,1\}^\lambda$

Shuffle

$$\begin{aligned} & \text{Enc}_{k_0^b}(\text{Enc}_{k_0^c}(k_0^e)) \\ & \text{Enc}_{k_0^b}(\text{Enc}_{k_2^c}(k_0^e)) \\ & \text{Enc}_{k_1^b}(\text{Enc}_{k_0^c}(k_0^e)) \\ & \text{Enc}_{k_1^b}(\text{Enc}_{k_1^c}(k_0^e)) \end{aligned}$$

$$\text{Enc}_{k_0^b}(\text{Enc}_{k_0^c}(k_0^e \parallel \overbrace{0 \dots 0}^\lambda))$$

$$H(k_0^b \parallel k_0^c) \oplus (k_0^e \parallel 0 \dots 0)$$

Secure 2PC

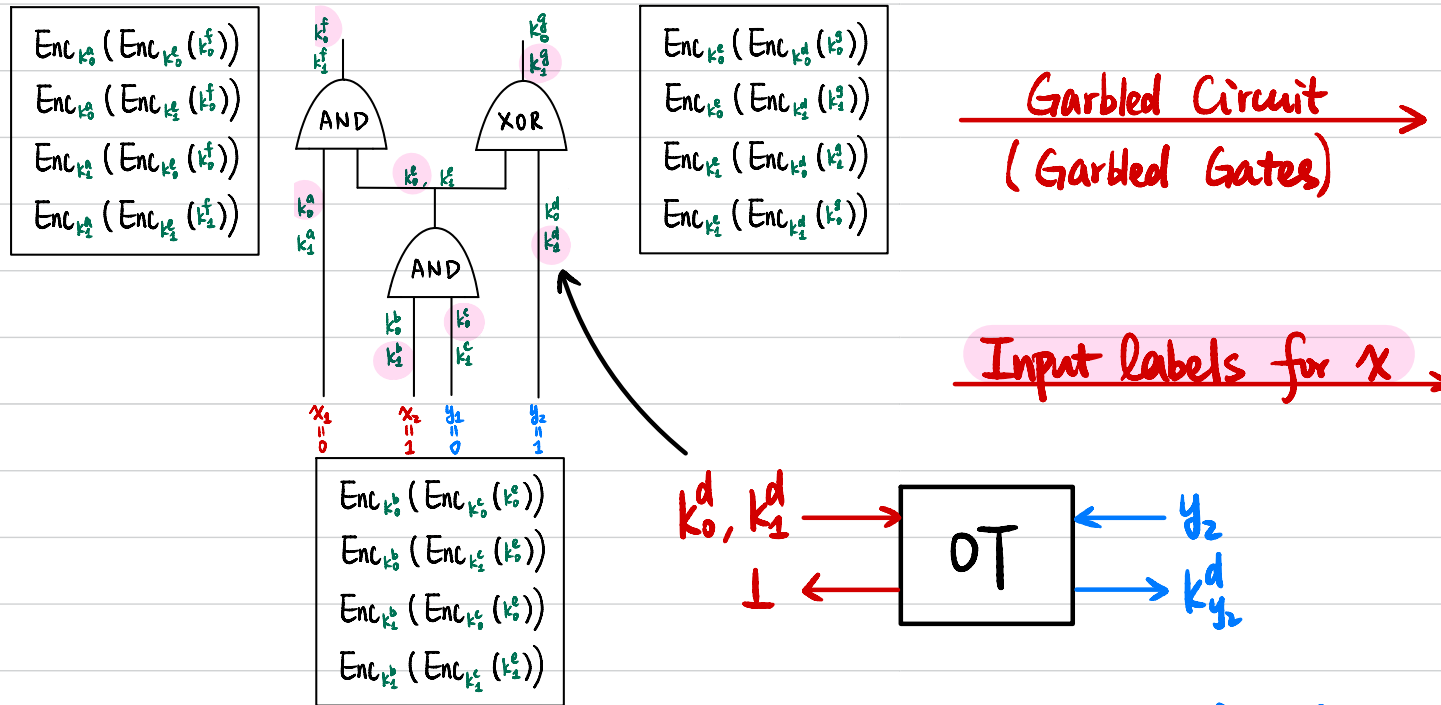
Evaluator gets one label per wire

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?

• Output labels?

① $k_0^f \rightarrow 0 \quad k_1^f \rightarrow 1$

② Bob sends output labels back to Alice

• How to decide which ciphertext to decrypt?

k_0^f, k_1^g
 z

Oblivious Transfer (OT)

$$\text{CDH: } g^a, g^b \stackrel{?}{\Rightarrow} g^{ab}$$

$$g^a \stackrel{?}{\Rightarrow} g^{a^2}$$

Sender

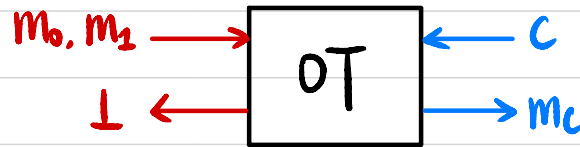
Input: $m_0, m_1 \in \{0, 1\}^L$

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

$$g^{ab} \cdot g^{a^2}$$

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

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



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

$$\xleftarrow[B = g^b \cdot A^c]{C=1}$$

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

Receiver

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

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

Output:

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

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

Why is it correct?

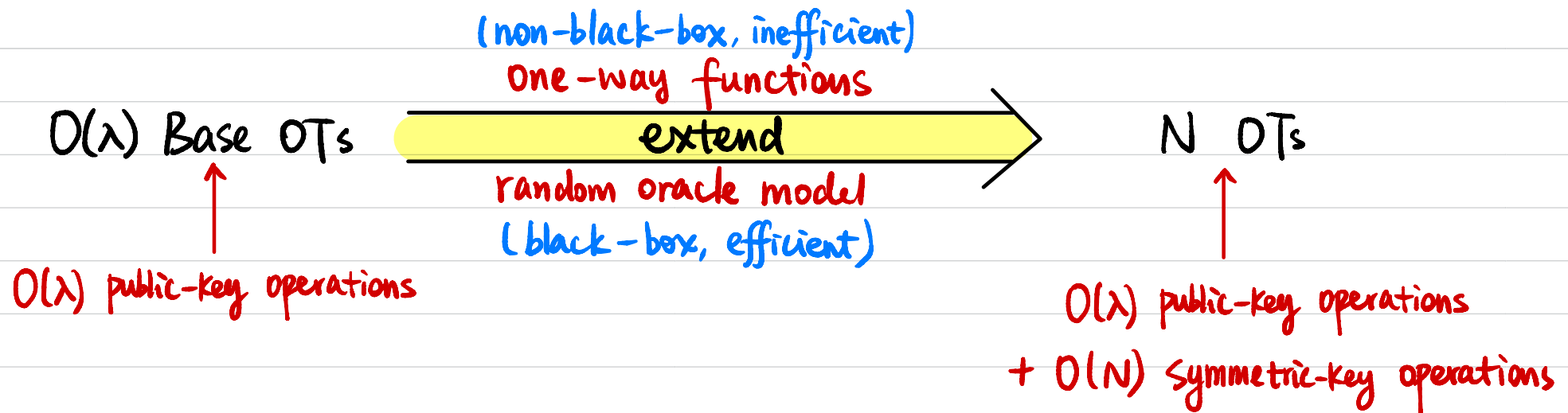
Why is it secure against semi-honest Sender?

Why is it secure against semi-honest Receiver?

OT Extension

Can we construct OT from symmetric-key primitives only?

Unlikely! (theoretical impossibility)

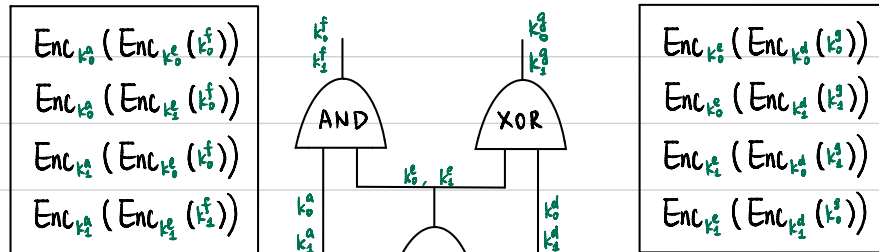


Putting it All Together: Semi-Honest ZPC

What could go wrong against malicious adversaries?

Alice (Garbler)

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



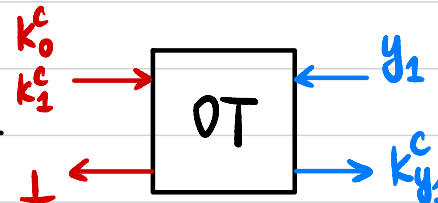
Garbled Circuit
(Garbled Gates)

Input labels for x

Bob (Evaluator)

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

Input labels for y: OT



Output labels k^f, k^g

Output z

Communication cost?

$4 \cdot 256 \cdot |C|$ bits

Computation cost?

$4 \cdot |C|$

