Secure Computation - II
(Garbled Circuits)

Lecture 14
Secure Two-Party Computation from Garbled Circuits
Garbled Circuits

[Yao86]
Garbled Circuits

[Yao86]
**Goal**: “Garble” (Circuit=\(C\), input=\(x\)) s.t. (\(C_{\text{garble}}, x_{\text{garble}}\)) only reveals \(C(x)\)
Syntax

- **Two algorithms**: \((\text{Garble}, \text{Eval})\)
  - \(\text{Garble}(C)\) outputs \((C_{\text{garble}}, x_{\text{garble}})\)
  - \(\text{Eval}(C_{\text{garble}}, x_{\text{garble}})\) outputs a value \(z\)
Syntax

- **Two algorithms**: \((\text{Garble}, \text{Eval})\)
  - \(\text{Garble}(C)\) outputs \((C_{\text{garble}}, x_{\text{garble}})\)
  - \(\text{Eval}(C_{\text{garble}}, x_{\text{garble}})\) outputs a value \(z\)

- Correctness:

  For every \((C, x)\),

  \[
  \Pr[C(x) = \text{Eval}(C_{\text{garble}}, x_{\text{garble}}) \mid (C_{\text{garble}}, x_{\text{garble}}) = \text{Garble}(C, x)] = 1 - \text{negl}(n)
  \]
Security

There exists a PPT simulator \( S \) s.t. for every \((C,x)\),

\[
(C_{\text{garble}},x_{\text{garble}}) \sim S(1^n, C, C(x))
\]

where \((C_{\text{garble}},x_{\text{garble}}) = \text{Garble}(C,x)\)
There exists a PPT simulator $S$ s.t. for every $(C,x)$,

$$(C_{\text{garble}}, x_{\text{garble}}) \sim S(1^n, C, C(x))$$

where $(C_{\text{garble}}, x_{\text{garble}}) = \text{Garble}(C, x)$

- Hiding $C$: Use universal circuit and pass $C$ as input to the universal circuit
Garbled Circuits
[Yao86]

**Garble**($C, x$):
- Pick random **labels** $W_0, W_1$ for each wire
**Garbled Circuits**

[Yao86]

Garble(C,x):
- Pick random **labels** $W_0, W_1$ for each wire
**Garbled Circuits**

[ Yao86 ]

**Garble(C,x):**

- Pick random **labels** $W_0, W_1$ for each wire
- “Encrypt” truth table of each gate

---

- $\text{AND}$
- $\text{OR}$
- $\text{NOT}$

- $A_0, A_1$
- $B_0, B_1$
- $C_0, C_1$
- $D_0, D_1$
- $E_0, E_1$
- $F_0, F_1$
- $G_0, G_1$
- $H_0, H_1$
- $I_0, I_1$
Garbled Circuits

[Yao86]

Garble(C,x):

- Pick random **labels** W_0,W_1 for each wire
- “Encrypt” truth table of each gate
- **Garbled Circuit** C_{garble} = all encrypted gates
Garbled Circuits

[Yao86]

Garble(C,x):

- Pick random labels $W_0, W_1$ for each wire
- “Encrypt” truth table of each gate
- **Garbled Circuit** $C_{\text{garble}}$ = all encrypted gates
- **Garbled Input** $x_{\text{garble}}$ = one label per wire
Garbled Circuits
[Yao86]

Garble(C,x):
• Pick random labels \( W_0, W_1 \) for each wire
• “Encrypt” truth table of each gate

• Garbled Circuit \( C_{\text{garble}} \) = all encrypted gates
• Garbled Input \( x_{\text{garble}} \) = one label per wire

Eval(\( C_{\text{garble}}, x_{\text{garble}} \)):
• Only one ciphertext per gate is decryptable
Garbled Circuits

[Yao86]

Garble\(C, x\):
- Pick random labels \(W_0, W_1\) for each wire
- “Encrypt” truth table of each gate
- **Garbled Circuit** \(C_{\text{garble}}\) = all encrypted gates
- **Garbled Input** \(x_{\text{garble}}\) = one label per wire

\[\text{Eval}(C_{\text{garble}}, x_{\text{garble}}):\]
- Only one ciphertext per gate is decryptable
- Result of decryption = value on outgoing wire
**Garbled Circuits**

[Yao86]

- **Garble(C,x):**
  - Pick random **labels** \(W_0, W_1\) for each wire
  - “Encrypt” truth table of each gate
  - **Garbled Circuit** \(C_{\text{garble}}\) = all encrypted gates
  - **Garbled Input** \(x_{\text{garble}}\) = one label per wire

- **Eval\((C_{\text{garble}}, x_{\text{garble}})\):**
  - Only one ciphertext per gate is decryptable
  - Result of decryption = value on outgoing wire
**Garbled Circuits**

- **AND**, **OR**, and **NOT**

\[ \begin{align*}
A_0, A_1 & \rightarrow E_0, E_1 \\
B_0, B_1 & \rightarrow F_0, F_1 \\
C_0, C_1 & \rightarrow G_0, G_1 \\
D_0, D_1 & \rightarrow H_0, H_1
\end{align*} \]

- **Garble** \((\text{C}, \text{x})\):
  - Pick random **labels** \(W_0, W_1\) for each wire
  - “Encrypt” truth table of each gate
  
  - **Garbled Circuit** \(C_{\text{garble}}\) = all encrypted gates
  
  - **Garbled Input** \(x_{\text{garble}}\) = one label per wire

- **Eval** \((C_{\text{garble}}, x_{\text{garble}})\):
  - Only one ciphertext per gate is decryptable
  - Result of decryption = value on outgoing wire

\[ \text{Yao86} \]
Garbled Circuits

[Yao86]

**Garble(C,x):**
- Pick random labels $W_0,W_1$ for each wire
- “Encrypt” truth table of each gate
- **Garbled Circuit** $C_{\text{garble}}$ = all encrypted gates
- **Garbled Input** $x_{\text{garble}}$ = one label per wire

**Eval(C_{\text{garble}},x_{\text{garble}}):**
- Only one ciphertext per gate is decryptable
- Result of decryption = value on outgoing wire
The finer details

Privacy (intuition):

• For each wire (including input wires), adversary only sees one label $W_b$
The finer details

Privacy (intuition):

• For each wire (including input wires), adversary only sees one label $W_b$
• The 4 entries in each encrypted table are in random order
The finer details

Privacy (intuition):

- For each wire (including input wires), adversary only sees one label $W_b$
- The 4 entries in each encrypted table are in random order
- Adversary tries to decrypt each entry. Only one decryption succeeds.
The finer details

Privacy (intuition):

- For each wire (including input wires), adversary only sees one label $W_b$
- The 4 entries in each encrypted table are in random order
- Adversary tries to decrypt each entry. Only one decryption succeeds.
- Adversary has no idea whether $b=0$ or $b=1$ for any label $W_b$
The finer details

Privacy (intuition):

• For each wire (including input wires), adversary only sees **one** label $W_b$

• The 4 entries in each encrypted table are in random order

• Adversary tries to decrypt each entry. Only one decryption succeeds.

• Adversary has no idea whether $b=0$ or $b=1$ for any label $W_b$

Interpreting the output:

• For every output wire, reveal the mappings $(b, W_b)$
Secure Computation from Garbled Circuits

**Goal:** Compute $f(x, y)$

$x$

$y$
Secure Computation from Garbled Circuits

Goal: Compute $f(x, y)$

1. Garbled circuit $f_{\text{garble}}$
2. Garbled input $x_{\text{garble}}$
Secure Computation from Garbled Circuits

Goal: Compute $f(x, y)$

1. Garbled circuit $f_{\text{garble}}$
2. Garbled input $x_{\text{garble}}$

Problem: How to transmit $y_{\text{garble}}$?
Secure Computation from Garbled Circuits

**Goal:** Compute $f(x, y)$

1. Garbled circuit $f_{\text{garble}}$
2. Garbled input $x_{\text{garble}}$

All Labels for 2$^{nd}$ input

$y_{\text{garble}}$
Secure Computation from Garbled Circuits

Goal: Compute $f(x,y)$

1. Garbled circuit $f_{\text{garble}}$
2. Garbled input $x_{\text{garble}}$

Want:
- Alice learns nothing about $y$
Secure Computation from Garbled Circuits

Goal: Compute $f(x,y)$

1. Garbled circuit $f_{\text{garble}}$
2. Garbled input $x_{\text{garble}}$

Want:
- Alice learns nothing about $y$
- Bob does not learn the other labels
Oblivious Transfer

$W_0, W_1 \rightarrow b \rightarrow W_b \rightarrow b$

[Rabin81, Even-Goldreich-Lempel-85]
Oblivious Transfer

Want:
- Alice does not learn $b$
- Bob does not learn $W_{1-b}$

[Rabin81, Even-Goldreich-Lempel-85]