Garbling scheme: $(\text{Garble}, \text{Eval})$

$C(x,y) = z$; two n-bit inputs; one n-bit output

$$\left(\{(X_i^0, X_i^1)\}_{i=1}^n, \{(Y_i^0, Y_i^1)\}_{i=1}^n, GC, \{(Z_i^0, Z_i^1)\}_{i=1}^n \right) \leftarrow \text{Garble}(C)$$

$$\{Z_i\}_{i=1}^n \leftarrow \text{Eval}(\{X_i^{x_i}\}, \{Y_i^{y_i}\}, GC)$$

Correctness: for all x, y, if $z = C(x,y)$ then $Z_i = Z_i^{z_i}$ for all i

Security: there is a simulator Sim s.t. the following are computationally indistinguishable:

$$\{\{(X_i^0, X_i^1)\}, \{(Y_i^0, Y_i^1)\}, GC, \{(Z_i^0, Z_i^1)\}\} \leftarrow \text{Garble}(C) : \{X_i^{x_i}\}, \{Y_i^{y_i}\}, GC, \{(Z_i^0, Z_i^1)\}\}_{x,y}$$

and

$$\{\text{Sim}(y, C(x,y))\}_{x,y}$$

$P_1(x)$ $P_2(y)$

$\leftarrow Garble(C)$

$(\{X_i^0, X_i^1\}, \{Y_i^0, Y_i^1\}, GC, \{Z_i^0, Z_i^1\})$

GC →

$\{X_i^{x_i}\}_{i=1}^n$ →

$\{Z_i\} \leftarrow Eval(\{X_i\}, \{Y_i\}, GC)$

$\{(Z_i^0, Z_i^1)\}_{i=1}^n$ →

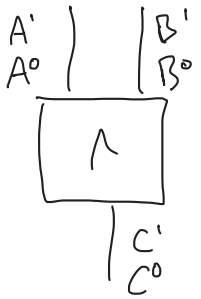
For all i do:

if $Z_i = Z_i^0$, $z_i = 0$

if $Z_i = Z_i^1$, $z_i = 1$

output z

Garbling scheme



$Enc_{A^0}(Enc_{B^0}(C^0))$

$Enc_{A^0}(Enc_{B^1}(C^0))$

$Enc_{A^1}(Enc_{B^0}(C^0))$

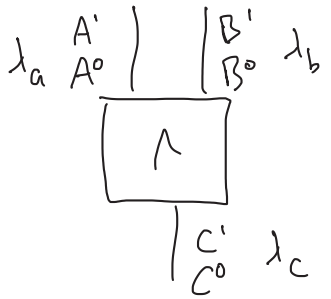
$Enc_{A^1}(Enc_{B^1}(C^1))$

randomly permute

Assume $Dec_K(Enc_{K'}(m)) = \perp$ if $K \neq K'$

Given $A \in \{A^0, A^1\}$ and $B \in \{B^0, B^1\}$, evaluator can compute (the correct) $C \in \{C^0, C^1\}$

Point-and-permute



instead of associating A^0 w/ 0, now associate A^0 w/ λ_a

Give evaluator the label of a key that it learns

$$Enc_{A^0}(Enc_{B^0}(C^{(\lambda_a \wedge \lambda_b) \oplus \lambda_c}, (\lambda_a \wedge \lambda_b) \oplus \lambda_c))$$

$$\vdots$$

$$Enc_{A^1}(Enc_{B^1}(C^{(\bar{\lambda}_a \wedge \bar{\lambda}_b) \oplus \lambda_c}, (\bar{\lambda}_a \wedge \bar{\lambda}_b) \oplus \lambda_c))$$

- no need to randomly permute!
- evaluator knows which row to decrypt based on the labels
  - reduce # of decryptions by evaluator
- no longer need redundancy in encryption scheme
  - ciphertexts can be shorter
  - in particular, i^{th} garbled gate can be computed as

$$F_{A^0}(00||i) \oplus F_{B^0}(00||i) \oplus [C^{(\lambda_a \wedge \lambda_b) \oplus \lambda_c}, (\lambda_a \wedge \lambda_b) \oplus \lambda_c], \text{ etc.}$$

Garbled row reduction

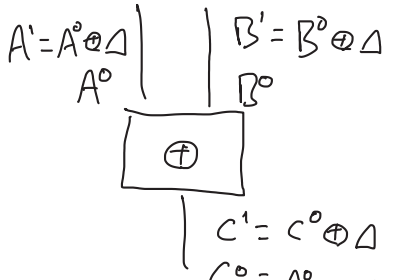
Idea:
- wire keys & labels are random
- garbler can set λ_c and $C^{(\lambda_a \wedge \lambda_b) \oplus \lambda_c}$ s.t. the initial row is all-0
  - I.e., $\left[C^{(\lambda_a \wedge \lambda_b) \oplus \lambda_c}, (\lambda_a \wedge \lambda_b) \oplus \lambda_c\right] = F_{A^0}(00\|i) \oplus F_{B^0}(00\|i)$
- Can avoid sending the 1st row!
- Just 3 ciphertexts per garbled gate

---

half-gates – reduce # ciphertexts per garbled gate to 2

Free-XOR – avoid garbling XOR gates at all

global shift Δ



note $A^a \oplus B^b = A^0 \oplus a\Delta \oplus B^0 \oplus b\Delta$

$= (A^0 \oplus B^0) \oplus (a \oplus b)\Delta$

$= C^0 \oplus (a \oplus b)\Delta$

$= C^{a \oplus b}$