Oblivious Transfer based on Exponential El Gamal.

$G, p = 47, q = 23, g = 17$, $G = (17, 7, 25, 2, 34, \dots, 19, 24, 37, \cancel{27}, 30, 1)$ ©

P_1

$r_0, r_1 = \text{rand}(\mathbb{Z}_q^2) = (13, 4)$

Secrets

$m_0 = 14 \in \mathbb{Z}_p \wedge m_1 = 15 \in \mathbb{Z}_p$

$r' = \text{rand}(\mathbb{Z}_p) = 3$ $\qquad h = pk$

$X = \text{Enc}(h) = (24; 9)$

$Y = X^{45} \% 47 = (2, 21)$

$Z = (g^{r} \% p, g^{1} \cdot h^{r} \% p)$

$= (17^3 \% 47, 17 \cdot 4^3 \% 47) = (25, 7)$

(MOD p)

$c_0 = [(25, 7) \cdot (2, 21)]^{14} \cdot (24, 9)^{13}$

$= (25^{14} \cdot 2^{14} \cdot 24^{13} \% 47,$

$7^{14} \cdot 21^{14} \cdot 9^{13} \% 47)$

$= (1, 9)$

$c_1 = (\cancel{25} 24^{15} \cdot 25^4 \cdot 2^4, 9^{15} \cdot 7^4 \cdot 21^4) \% 47 = (28, 12)$

See verso for details!!

$pk = 4$; $\text{Enc}_{pk}(b) = (24; 9)$ ←

$(c_0, c_1) = ((1, 9), (28, 12))$ →

$c_0 = E$

P_2

$\alpha = sk = 8 \in \mathbb{Z}_p$

$pk = h = g^{\alpha} \% p = 17^8 \% 47 = 4$

$b = \text{rand}(\mathbb{Z}_2) = \text{rand}(\{0; 1\}) = 0$

$\text{Enc}_{pk}(b) = \begin{cases} r = \text{rand}(\mathbb{Z}_q) = 19;\ c_1 = g^{r} \% p = 17^{19} \% 47 = 2 \\ c_2 = g^{m} \cdot h^{r} \% p = 17^0 \cdot 4^{19} \% 47 = 9 \end{cases}$

As $b == 0$ chooses c_0

$D = c_{00}^{\alpha} \% p = 1^8 \% 47 = 1$

$D^{-1} = \cancel{D^{45} \%}\ D^{p-2} \% p = 1^{45} \% 47 = 1$

$D' = c_{01} \cdot D^{-1} \% p = 9 \cdot 1 \% 47 = 9$

$\text{Dec}_{sk}(c_{00}, c_{01}) = \log_{g=17}(9)^{\% 47} = 14 = m_0$

$$c_0 = E((1-b)m_0 + r_0 b) = E((1-b)m_0) \cdot E(r_0 b) = E(1-b)^{m_0} \cdot E(b)^{r_0} = [E(1) \cdot E(b)^{-1}]^{m_0} \cdot E(b)^{r_0} \pmod p$$

$$c_1 = E(b m_1 + r_1(1-b)) = E(b m_1) \cdot E(r_1(1-b)) = E(b)^{m_1} \cdot E(1-b)^{r_1} = E(b)^{m_1} \cdot [E(1) \cdot E(b^{-1})]^{r_1} \pmod p$$

$X = E(b) \in \mathbb{Z}_p$

$Y = E(b)^{-1} = X^{p-2} \% p$

$Z = E(1) = (g^r \% p, g^1 \cdot h^r \% p)$

$r = \text{round}(\mathbb{Z}_p)$

$$\begin{cases} c_0 = [Z \cdot Y]^{m_0} \cdot X^{r_0} \% p \\ c_1 = X^{m_1} \cdot [Z \cdot Y]^{r_1} \% p \end{cases}$$

$$\Leftrightarrow \begin{cases} c_0 = (Z_0^{m_0} \cdot Y_0^{m_0} \cdot X_0^{r_0}, Z_1^{m_0} \cdot Y_1^{m_0} \cdot X_1^{r_0}) \% p \\ c_1 = (X_0^{m_1} \cdot Z_0^{r_1} \cdot Y_0^{r_1}, X_1^{m_1} \cdot Z_1^{r_1} \cdot Y_1^{r_1}) \% p \end{cases}$$

Note if b is a vector:

$b^{-1} \% p = (b_0^{-1} \bmod p, \dots, b_n^{-1} \bmod p)$

$b^l \% p = (b_0^l \bmod p, \dots, b_n^l \bmod p)$

$b \cdot b' \% p = (b_0 \cdot b_0' \% p, \dots, b_n \cdot b_n' \% p)$

Yao's Garbled Circuits: (WIKIPEDIA)

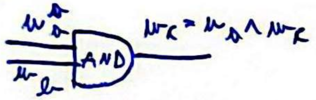
Garbler ~~(a~~ P_0 encrypts a boolean circuit to ~~of~~ obtain a garbled circuit.

$k \equiv$ ~~Aes~~ security parameter

(1) GARBLING

Boolean circuit (ex AND):

a	b	c
0	0	0
0	1	0
1	0	0
1	1	1



Encrypts the output entries of truth table.

a	b	c
X_a^0	X_b^0	X_c^0
X_a^0	X_b^1	X_c^0
X_a^1	X_b^0	X_c^0
X_a^1	X_b^1	X_c^1

$X_a^0 \equiv$ A label representing 0 for the wire a that only P_0 understands the meaning.

Garbled table:

Encrypted table $= T = [w, x, y, z]$

$[\underbrace{Enc_{X_a^0 X_b^0}(X_c^0)}_{*}, \underbrace{Enc_{X_a^0 X_b^1}(X_c^0)}_{*}, \underbrace{Enc_{X_a^1 X_b^0}(X_c^0)}_{*}, \underbrace{Enc_{X_a^1 X_b^1}(X_c^1)}_{*}]$

* All elements are concatened with ending 0 so $X_c^{k_i} || 0^n$ allowing P_1 to detect decryption errors thanks to pattern matching.

$Enc_K(X) \equiv$ Double-key symmetric encryption where K is the key.

Garbled table $= T' =$ shuffle$(T) = [z, w, x, y]$

(2) DATA TRANSFER

P_0		P_1
Compute garbled table T_i for each bit i	T' $\rightarrow$	$b = b_0 b_1 b_2 b_3 b_4 = 10100$
P_0's input $a = a_0 a_1 a_2 a_3 a_4 = 01101$	${}_0X_a^0\, {}_1X_a^1\, {}_2X_a^1\, {}_3X_a^0\, {}_4X_a^1 = A$ $\rightarrow$	
$m_0 = {}_iX_b^0$	OT, $\forall i$	Get $X_b^{b_i}$ Actually …
$m_1 = {}_iX_b^1$	$\leftarrow$ Enc(b_i)	
	c_0, c_1 $\rightarrow$	$\Rightarrow {}_0X_b^1\, {}_1X_b^0\, {}_2X_b^1\, {}_3X_b^0\, {}_4X_b^0 = B$
	where	DECODE

* $w =$ Number that has invalid pattern (no n ending zeros) so invalid.

(3) EVALUATION $(A = X_a, B = X_b, C = X_c)$

$\forall i: {}_iT' = [{}_iX_c^{AND(1,1)}, {}_iX_c^{AND(0,0)}, {}_iX_c^{AND(0,1)}, {}_iX_c^{AND(1,0)}]$

$C = Dec_{{}_iA, {}_iB}({}_iT'[i]) \mid i \in \{0;1;2;3\}$

$$\Rightarrow C = \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} w & w & X_c^1 & w & w \\ w & w & w & X_c^0 & w \\ X_c^0 & w & w & w & w \\ w & X_c^0 & w & w & X_c^0 \end{bmatrix} \end{array}$$

$A, B = [(X_a^0, X_b^1), (X_a^1, X_b^0), (X_a^1, X_b^1), (X_a^0, X_b^0), (X_a^1, X_b^0)]$

Output $= [{}_0X_c^0, {}_1X_c^0, {}_2X_c^1, {}_3X_c^0, {}_4X_c^0]$

(4) REVEALING OUTPUT

P_0 $\xleftarrow{\text{output}}$ P_1

P_0 ~~does map~~ * unmaps: $c = 0,0,1,0,0$

Output $= [{}_0X_c^0, {}_1X_c^0, {}_2X_c^1, {}_3X_c^0, {}_4X_c^0]$

* P_0 $\forall i$:
Ask(index $= i$, label $= {}_iX_c^{b_i}$) $= b_i$

Conclusion: Both P_0 and P_1 have secret inputs that cannot be revealed to the other.

P_0 creates a garbled map table mapping each encrypted bit $E(b_i)$ sent by P_1 to the associated $X_b^{b_i}$ thanks to Oblivious Transfer: P_0 has no idea of b_i and P_1 gets $X_b^{b_i}$ without understanding it.

P_1 cannot map 0 or 1 to X_b^0 or X_b^1 as for every bit b_i Alice generate a new collection $({}_iX_a^0, {}_iX_a^0, {}_iX_b^0, {}_iX_b^1, {}_iX_c^0, {}_iX_c^1)$.

$${}_iX_b^{b_i} \neq {}_jX_b^{b_i} \text{ if } i \neq j$$

~~P_1 has a secret~~ P_1 has computed a secret function only understood by P_0 using its secret input and he doesn't understand the result.

Each $Enc_{pk}(b_i)$ is different even if $b_i = b_j$ because each bit is sent with a random r.