# Post-quantum secure oblivious transfer

**Emmanuela Orsini**

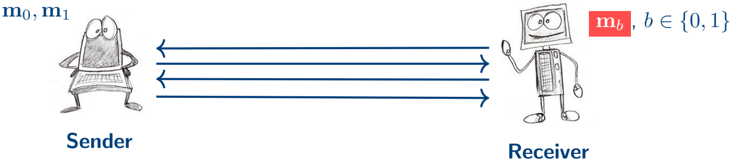
imec-COSIC, KU Leuven

# Talk outline

- Oblivious transfer: definition, motivation, security
- Efficient, non-PQ secure OT protocols
- Examples of PQ-secure OT

# Oblivious transfer – Definition

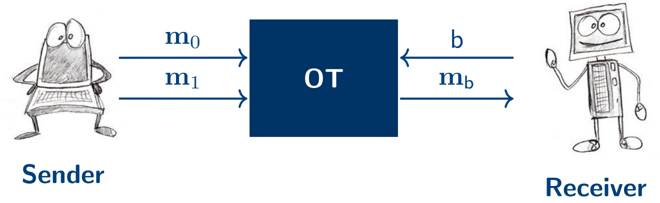
Oblivious Transfer (OT) is a ubiquitous cryptographic primitive designed to transfer specific data based on the receiver's choice.




No further information should be learned by any party

<u>Why we care?</u>: Complete for secure 2-party and multi-party computation, used as a building block in many cryptographic protocols etc.

# 1-out-of-2 oblivious transfer




Security for the receiver : The sender should not learn anything about the bit b

Security for the sender : The receiver should not learn anything about $\mathbf{m}_{1-b}$

# Many flavours of OT


**Standard OT and COT functionality**


**1-out-of-2 OT and 1-out-of-$n$ OT**

# Oblivious transfer – General results and security

- First introduced by M. Rabin in 1981 (based on RSA)
- Previously described by Wiesner in 1975 (as multiplexing)

# Oblivious transfer – General results and security

- First introduced by M. Rabin in 1981 (based on RSA)
- Previously described by Wiesner in 1975 (as multiplexing)

* OT cannot achieve *information theoretic security* for both parties over a standard, noiseless communication channel
* If a noisy channel of certain form is available between the sender and the receiver, OT can be constructed with unconditional security.

# Oblivious transfer – General results and security

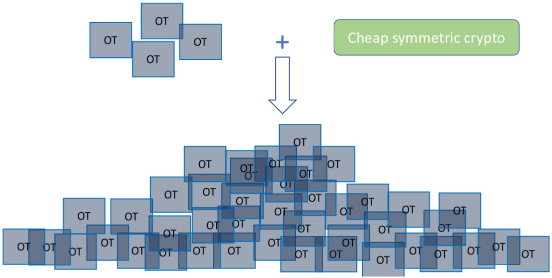
- First introduced by M. Rabin in 1981 (based on RSA)
- Previously described by Wiesner in 1975 (as multiplexing)

* OT cannot achieve *information theoretic security* for both parties over a standard, noiseless communication channel
* If a noisy channel of certain form is available between the sender and the receiver, OT can be constructed with unconditional security.

- Impagliazzo, Rudich [IR98]
  Black-box separation result $\rightarrow$ OT is impossible without public-key primitives (?)
- We cannot construct OT from PKE in a black box way
  + Enhanced trapdoor permutation
  + DDH, RSA, lattices, error-correcting codes, isogenies etc.

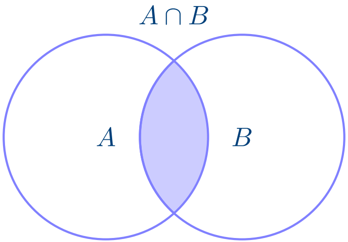
## Oblivious transfer – Efficiency

- Impagliazzo, Rudich [IR98]
  Black-box separation result → OT is impossible without public-key primitives (?)
- Beaver [Beaver96]: OT can be extended



# Oblivious transfer – Applications

**Private Set Intersection (PSI)**: Given two parties Alice and Bob with two set of items $A = \{a_1, \ldots, a_k\}$ and $B = \{b_1, \ldots, b_m\}$.
The *goal* is to design a protocol by which Alice and Bob obtain the intersection $A \cap B$, such that nothing is revealed but the items that are in the intersection.



# Oblivious transfer – Applications

- DNA analysis
- Contact discovery
- Remote diagnostic
- Record linkage
- Measuring the effectiveness of online advertising
- and many more

# Part II: Building OT from cryptographic assumption

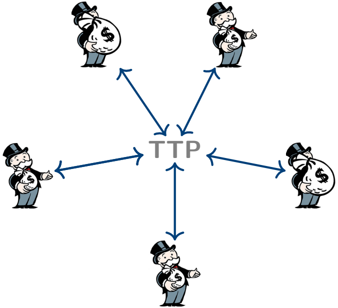
# Oblivious transfer – Security

- **Semi-honest**: adversary running the correct protocol cannot learn anything
- **Malicious**: adversary running any protocol cannot learn anything

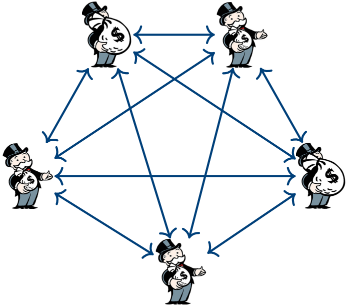
⋆ The strongest form of security we can hope for is universal composability (UC).
- Very difficult and expensive to achieve
- [PVW Crypto 2008] *A framework for efficient and composable oblivious transfer*

**Disclaim:** We are going to talk about security in a *very* informal way.

# Security definition

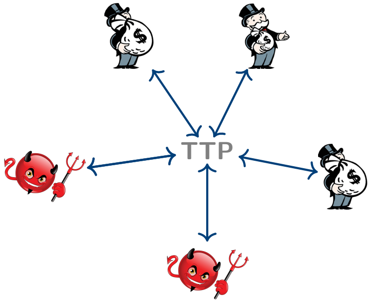



**Ideal world**

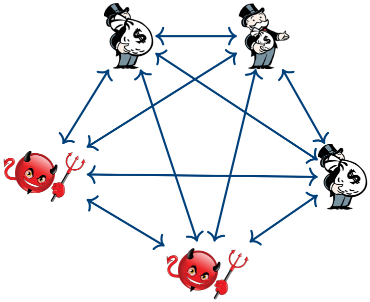


**Real world**

# Security definition




**Ideal world**



**Real world**

What OT protocols (or PKE) we will use in 100 years?

# Oblivious transfer – Security

- Shor's algorithm
  - Integer factorization: RSA broken
  - Discrete logarithm: (EC-)DSA, (EC-)DH,... broken
- Quantum computers
  - Theoretically viable, engineering effort to scale sizes
  - NIST has started a "PQ Standardization Process" which has recently entered the third round
    - Key encapsulation, PK encryption, digital signatures

# Families of post-quantum secure algorithms (so far...)

- Code-based
- Isogeny-based
- Hash-based
- Lattice-based
- Multivariate-systems based

# NIST PQ candidates 3rd round

**Table 2.1:** NIST Round 3 candidates

| Scheme | Enc/Sig | Family | Hard Problem |
|---|---|---|---|
| **Round 3 Finalists** | | | |
| Classic McEliece | Enc | Code-Based | Decoding random binary Goppa codes |
| Crytals-Kyber | Enc | Lattice-Based | Cyclotomic Module-LWE |
| NTRU | Enc | Lattice-Based | Cyclotomic NTRU Problem |
| Saber | Enc | Lattice-Based | Cyclotomic Module-LWR |
| Crystals-Dilithium | Sig | Lattice-Based | Cyclotomic Module-LWE and Module-SIS |
| Falcon | Sig | Lattice-Based | Cyclotomic Ring-SIS |
| Rainbow | Sig | Multivariate-Based | Oil-and-Vinegar Trapdoor |
| **Round 3 Alternate Candidates** | | | |
| BIKE | Enc | Code-Based | Decoding quasi-cyclic codes |
| HQC | Enc | Code-Based | Coding variant of Ring-LWE |
| Frodo-KEM | Enc | Lattice-Based | LWE |
| NTRU-Prime | Enc | Lattice-Based | Non-cyclotomic NTRU Problem or Ring-LWE |
| SIKE | Enc | Isogeny-Based | Isogeny problem with extra points |
| GeMSS | Sig | Multivariate-Based | 'Big-Field' trapdoor |
| Picnic | Sig | Symmetric Crypto | Preimage resistance of a block cipher |
| SPHINCS+ | Sig | Hash-Based | Preimage resistance of a hash function |

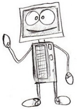
# Oblivious transfer from DH key exchange – 1

**Common input:** A group $\mathbb{G}$ of prime order $q$ and a generator $g$

**Alice**

$M, a$

**Bob**



$b$

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

$$\xleftarrow{B = g^b}$$

$$\xrightarrow{e = \mathsf{Enc}_K(M)}$$

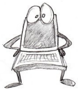
$$K = H(B^a) = H(g^{ba})$$

$$K = H(A^b) = H(g^{ab})$$
$$M = \mathsf{Dec}_K(e)$$

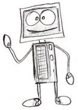
**Security:** Computational DH. Fixed $\langle g \rangle = \mathbb{G}$ and given $(g, g^a, g^b)$, with $a, b$ randomly chosen, it is hard to compute $g^{ab}$.

# Oblivious transfer from ECDH key exchange – 1

**Common input:** An elliptic curve $E$ over a finite field $K$, a subgroup of prime order $q$ of $E(K)$, a generator $P$



$M, a \in \mathbb{Z}/q\mathbb{Z}$



$b \in \mathbb{Z}/q\mathbb{Z}$

$$\xrightarrow{A = [a]P}$$

$$\xleftarrow{B = [b]P}$$

$$\xrightarrow{e = \mathsf{Enc}_K(M)}$$

$K = H([a]B) = H([ab]P)$

$K = H([b]P) = H([ab]P)$
$M = \mathsf{Dec}_K(e)$

# CDH and DDH

**Computational Diffie–Hellman (CDH) problem.** Fixed $E, P$ as before and given the tuple

$$(P, P_a, P_b) = (P, [a]P, [b]P),$$

with $a, b$ randomly chosen, it is hard to compute

$$[ab]P$$

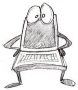
**Decision Diffie–Hellman (DDH) problem.** Is given the tuple

$$(P, P_a, P_b, P_c) = (P, [a]P, [b]P, [c]P)$$

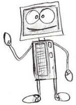
where $c$ is selected with probability $1/2$ to be uniformly random, and with probability $1/2$ to be equal to $ab \pmod{q}$. Then determine which case you are in.

# OT from key-exchange [T. Chou and C. Orlandi]

**Common input:** A group $\mathbb{G}$ of prime order $q$ and a generator $g$



$m_0, m_1, a$



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

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

$x = 0 \; B = g^b$

$x = 1 \; B = Ag^b$

$$\xleftarrow{B}$$

$K_0 = H(B^a))$

$K_1 = H((\frac{B}{A})^a))$

$K = H(A^b)$

$$\xrightarrow[e_1 = \mathsf{Enc}_{K_1}(m_1)]{e_0 = \mathsf{Enc}_{K_0}(m_0)}$$

$m_x = \mathsf{Dec}_K(e_x)$

# OT from key-exchange – Correctness and security intuition

**Security for the sender**

- $x = 0,\ B = g^b$. The sender (Alice) computes

$$K_0 = H(B^a)) = H(g^{ba}) \qquad K_1 = H((\frac{B}{A})^a)) = H(g^{ba-a^2})$$

  Bob computes $K = H(g^{ab}) = K_0$

- $x = 1,\ B = Ag^b$. The sender computes

$$K_0 = H(B^a)) = H(g^{a^2+ab}) \qquad K_1 = H((\frac{B}{A})^a)) = H(g^{ba})$$

  Bob computes $K = H(g^{ab}) = K_1$

**Security for the receiver** The sender is not able to get any information about $x$ from $B$

# OT from key-exchange – Correctness and security intuition

Security for the sender

- $x = 0,\ B = g^b$. The sender (Alice) computes

$$K_0 = H(B^a)) = H(g^{ba}) \qquad K_1 = H((\frac{B}{A})^a)) = H(g^{ba-a^2})$$

  Bob computes $K = H(g^{ab}) = K_0$

- $x = 1,\ B = Ag^b$. The sender computes

$$K_0 = H(B^a)) = H(g^{a^2+ab}) \qquad K_1 = H((\frac{B}{A})^a)) = H(g^{ba})$$

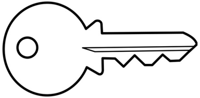
  Bob computes $K = H(g^{ab}) = K_1$

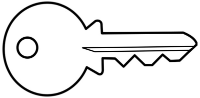
Security for the receiver The sender is not able to get any information about $x$ from $B$

$\star$ This protocol is **NOT UC-secure against malicious adversary**

# Oblivious transfer via lossy encryption - 1 [PVW08]

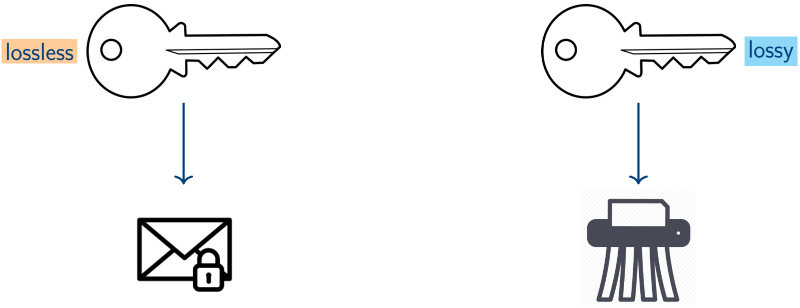
lossless



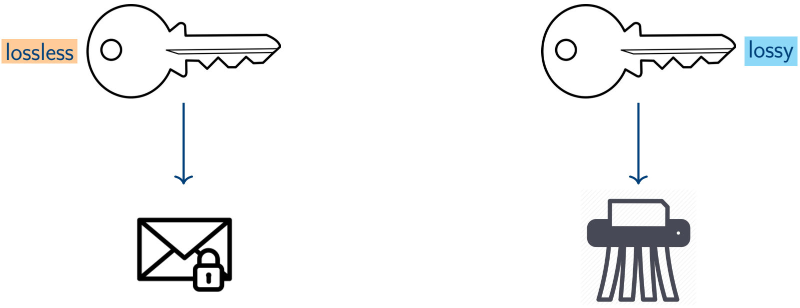


lossy

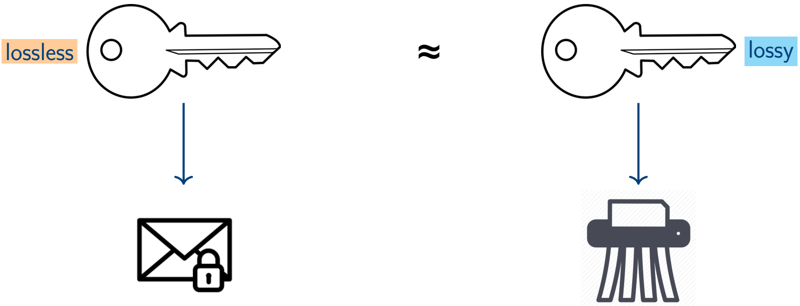
# Oblivious transfer via lossy encryption - 1 [PVW08]



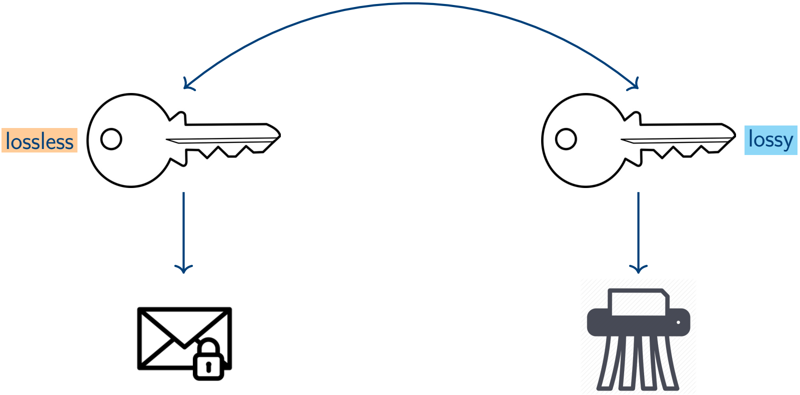
# Oblivious transfer via lossy encryption - 1 [PVW08]



# Oblivious transfer via lossy encryption - 1 [PVW08]



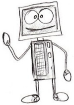
# Oblivious transfer via lossy encryption - 1 [PVW08]



# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information

$m_0, m_1$



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

$\xleftarrow{\mathsf{pk}}$

$x = 0$: pk lossless

$x = 1$: pk lossy

$\widetilde{\mathsf{pk}}$ in reverse mode

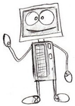
$$\xrightarrow[e_1 = \mathsf{Enc}_{\widetilde{\mathsf{pk}}}(m_1)]{e_0 = \mathsf{Enc}_{\mathsf{pk}}(m_0)}$$

$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information

$m_0, m_1$



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

$x = 0$: pk lossless

$$\xleftarrow{\mathsf{pk}}$$

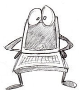
$\widetilde{\mathsf{pk}}$ in reverse mode

$$\xrightarrow[e_1 = \mathsf{Enc}_{\widetilde{\mathsf{pk}}}(m_1)]{e_0 = \mathsf{Enc}_{\mathsf{pk}}(m_0)}$$

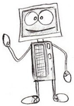
$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information



$m_0, m_1$



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

$x = 0$: pk lossless

$$\xleftarrow{\quad \mathsf{pk} \quad}$$

$\widetilde{\mathsf{pk}}$ in reverse mode

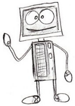
$$\xrightarrow[e_1 = \mathsf{Enc}_{\widetilde{\mathsf{pk}}}(m_1)]{e_0 = \mathsf{Enc}_{\mathsf{pk}}(m_0)}$$

$$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information

$m_0, m_1$



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

pk

$x = 0$: pk lossless

$\widetilde{\mathsf{pk}}$ in reverse mode

$e_0 = \mathsf{Enc}_{\mathsf{pk}}(m_0)$

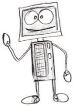
$e_1 = \mathsf{Enc}_{\widetilde{\mathsf{pk}}}(m_1)$

$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information

$m_0, m_1$



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

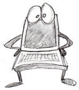
$\xleftarrow{\text{pk}}$

$x = 1$: pk lossy

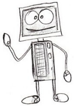
$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information



$m_0, m_1$



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

pk

$x = 1$: pk lossy

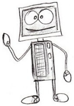
$\widetilde{\text{pk}}$ in reverse mode

$m_x = \text{Dec}_{\text{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 2

**CRS**: Lossy encryption scheme and other information

$m_0, m_1$



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

pk

$\longleftarrow$

$x = 1$: pk lossy

$\widetilde{\mathsf{pk}}$ in reverse mode

$e_0 = \mathsf{Enc}_{\mathsf{pk}}(m_0)$

$\longrightarrow$

$e_1 = \mathsf{Enc}_{\widetilde{\mathsf{pk}}}(m_1)$

$m_x = \mathsf{Dec}_{\mathsf{pk}}(e_x)$

# Oblivious transfer via lossy encryption - 3

- Concrete construction from DDH, QR, LWE

⋆ LWE-based scheme has weaker security guarantees compared to their group-based or number-theoretic counterparts.

1. Only achieves computational receiver security
2. Each CRS can only be securely used a bounded number of times
3. It allows for essentially single-bit transfers.

⋆ **A brief history of failure**: we tried to design a more efficient OT protocol from lossy encryption schemes based on Ring-LWE ... but we failed!

# Oblivious transfer via lossy encryption - 3

- Concrete construction from DDH, QR, LWE

⋆ LWE-based scheme has weaker security guarantees compared to their group-based or number-theoretic counterparts.

1. Only achieves computational receiver security
2. Each CRS can only be securely used a bounded number of times
3. It allows for essentially single-bit transfers.

⋆ **A brief history of failure**: we tried to design a more efficient OT protocol from lossy encryption schemes based on Ring-LWE ... but we failed!

# Isogeny-based oblivious transfer

∗ *Semi-Commutative Masking (SCM), a Framework for Isogeny-based Protocols*, Delpech de Saint Guilhem, O., Petit, Smart

---

- $q = p^2$
- Take supersingular elliptic curves $E_1, E_2$ elliptic curve over a finite field $\mathbb{F}_q$
- **Isogeny:** rational map (non-constant) over $\mathbb{F}_q$

$$\phi : E_1 \to E_2,$$

  that is a group homomorphism from $E_1(\mathbb{F}_q)$ to $E_2(\mathbb{F}_q)$
- For every prime $\ell$, there exists $\ell + 1$ isogeny class originating from any given supersingular curve
- Given a finite subgroup $K < E(\mathbb{F}_q)$, there is a unique isogeny class $\phi$ with kernel $K$, we write

$$\phi : E \to E/K$$

- We work with subgroups of torsion group $E[m]$ for $m \in \mathbb{N}$.

# Diffie-Hellman instantiations

| | **DH** | **ECDH** | **SIDH** |
|---|---|---|---|
| **elements** | integers $g$ mod a prime | points $P$ in curve group | curve $E$ in isogeny class |
| **secrets** | exponent $x$ | scalar $k$ | isogenies $\phi$ |
| **computations** | $g, x \mapsto g^x$ | $k, P \mapsto [k]P$ | $\phi, E \mapsto \phi(E)$ |
| **hard problems** | given $g, g^x$, find $x$ | given $P, [k]P$, find k | given $E, \phi(E)$, find $\phi$ |

# SIDH - Supersingular isogenies key-exchange (1)

## Setup and communication

- Fix starting curve $E/\mathbb{F}_{p^2}$.
- Prime $p = \ell_A^{e_A} \cdot \ell_B^{e_B} \cdot f \pm 1$ for small primes $\ell_A, \ell_B$ and small $f$.
- Let $\{P_A, Q_A\}$ be a basis of $E[\ell_A^{e_A}]$; similarly for $\{P_B, Q_B\}$.

$$\xrightarrow{E_A, \{\phi_A(P_B), \phi_A(Q_B)\}}$$

$$\xleftarrow{E_B, \{\phi_B(P_A), \phi_B(Q_A)\}}$$

$$\begin{tikzcd}
 & E_A = E/K_A \\
E \arrow[ur, "\phi_A"] \arrow[dr, "\phi_B"'] & \\
 & E_B = E/K_B
\end{tikzcd}$$

$K_A = [\alpha_A]P_A + [\beta_A]Q_A$

$K_B = [\alpha_B]P_B + [\beta_B]Q_B$

# SIDH - Supersingular isogenies key-exchange (1)

## Common key

- Alice compute $K_{AB} = [\alpha_A]\phi_B(P_A) + [\beta_A]\phi_B(Q_A)$
- Bob compute $K_{BA} = [\alpha_B]\phi_A(P_B) + [\beta_B]\phi_A(Q_B)$

$$\begin{tikzcd}
 & \begin{array}{c} E_A = E/K_A \\ {\scriptstyle K_A = [\alpha_A]P_A + [\beta_A]Q_A} \end{array} \arrow[dr, dashed, "\phi'_B"] & \\
E \arrow[ur, "\phi_A"] \arrow[dr, "\phi_B"'] & & \begin{array}{l} E_{AB} = E/K_{AB} \\ E_{BA} = E/K_{BA} \end{array} \\
 & \begin{array}{c} E_B = E/K_B \\ {\scriptstyle K_B = [\alpha_B]P_B + [\beta_B]Q_B} \end{array} \arrow[ur, dashed, "\phi'_A"'] &
\end{tikzcd}$$

$j(E_{AB}) = j(E_{BA}) \implies$ equal keys

# A 2-round oblivious transfer protocol

Constraint: **exponentiation-only mechanism**

$$
\begin{array}{ccc}
(\boxed{g_0, g_1};\ \boxed{a}) & & (\boxed{g_0, g_1};\ \boxed{x; b}) \\
 & \xleftarrow{\quad B = g_x^b \quad} & (g_x)^b \\
\begin{array}{l} m_0 = (g_0)^a \\ m_1 = (g_1)^a \end{array} & \xrightarrow{\quad B^a = g_x^{ab} \quad} & (g_x^{ab})^{1/b} = g_x^a
\end{array}
$$

Security proof against *passive* adversary in the UC framework.

# 2 round OT from SI

### Setup and receiver's message

- Fix starting two curves $E_0/\mathbb{F}_{p^2}$ and $E_1/\mathbb{F}_{p^2}$.
- Prime $p = \ell_A^{e_A} \cdot \ell_B^{e_B} \cdot f \pm 1$ for small primes $\ell_A, \ell_B$ and small $f$.
- Let $\{P_A^b, Q_A^b\}_{b\in\{0,1\}}$ be a basis of $E_b[\ell_A^{e_A}]$; similarly for $\{P_B^b, Q_B^b\}_{b\in\{0,1\}}$.

$$\xleftarrow{E_c^{B,1}, (\phi_B(P_A^c), \phi_B(Q_A^c)), (P_{B,1}^c, Q_{B,1}^c)}$$

$$E_c \xrightarrow{\phi_B} E_c^{B,1} = E_c/K_B$$

$K_B = [\alpha_B]P_B^c + [\beta_B]Q_B^c$

$(P_{B,1}, Q_{B,1})$ is a random basis of $E_c^{B,1}[\ell_B^{e_B}]$

# 2 round OT from SI

**Sender's message**

$$E_c^{B,1}, \underleftarrow{(\phi_B(P_A^c), \phi_B(Q_A^c)), (P_{B,1}, Q_{B,1})}\}$$

$$E_c^{A,2}, \underrightarrow{(\phi'_A(P_{B,1}), \phi'_A(Q_{B,1})), (c_0, c_1)}$$

$$E_{1-c} \xrightarrow{\phi_A^{1-c}} E_{1-c}^A$$

- **Sender:**

  $c_{1-c} = \mathsf{Enc}(m_{1-c}, j(E_{1-c}^A)), \quad c_c = \mathsf{Enc}(m_c, j(E_c^A))$

- **Receiver:** Can compute the dual isogeny $\widehat{\phi'_B}$, reaching a curve that is isomorphic to $E_c^A$. Compute $j(E_c^A)$ and retrieve $m_c$

$$E_c \xrightarrow{\phi_A^c} E_c^A = E_c/K_A^c$$

$$E_c \xrightarrow{\phi_B} E_c^{B,1} = E_c/K_B \xrightarrow{\phi'_A} E_c^{A,2}$$

$$K_B = [\alpha_B]P_B^c + [\beta_B]Q_B^c$$

# 2 round OT from SI

### Sender's message

$$E_c^{B,1}, (\phi_B(P_A^c), \phi_B(Q_A^c)), (P_{B,1}, Q_{B,1})\}$$

$$\longleftarrow$$

$$E_c^{A,2}, (\phi'_A(P_{B,1}), \phi'_A(Q_{B,1})), (c_0, c_1)$$

$$\longrightarrow$$

$$E_{1-c} \xrightarrow{\phi_A^{1-c}} E_{1-c}^A$$

- **Sender:**

  $c_{1-c} = \mathsf{Enc}(m_{1-c}, j(E_{1-c}^A)), \quad c_c = \mathsf{Enc}(m_c, j(E_c^A))$

- **Receiver:** Can compute the dual isogeny $\widehat{\phi'_B}$, reaching a curve that is isomorphic to $E_c^A$. Compute $j(E_c^A)$ and retrieve $m_c$

$$\begin{tikzcd}
 & E_c^A = E_c/K_A^c & \\
E_c \arrow[ur, "\phi_A^c"] \arrow[dr, "\phi_B"'] & & E_c^{A,2} \arrow[ul, dashed, "\widehat{\phi'_B}"'] \\
 & E_c^{B,1} = E_c/K_B \arrow[ur, "\phi'_A"'] &
\end{tikzcd}$$

$K_B = [\alpha_B]P_B^c + [\beta_B]Q_B^c$

# A 2-round OT from SI

- $3$-round OT extension protocol
- $2$-round OT with UC security in the semi-honest setting
- Compiling with [DGHMW20] 2-round OT with UC security in the malicious setting
- Post-quantum assumption

# Thank you!