Superposition Attack on OT Protocols

Ehsan Ebrahimi*1, Céline Chevalier2, Marc Kaplan3, and Michele Minelli4

1SnT, University of Luxembourg
2CRED, Université Panthéon-Assas, Paris II, France
3VeriQloud, Montrouge, France
4R&D Center Europe Brussels Laboratory, Sony, Belgium

July 1, 2020

Abstract

In this note, we study the security of oblivious transfer protocols in the presence of adversarial superposition queries. We define a security notion for the sender against a corrupted receiver that makes a superposition query. We present an oblivious transfer protocol that is secure against a quantum receiver restricted to a classical query but it is insecure when the receiver makes a quantum query. In addition, we present an OT protocol that resists to the attack presented in this paper. However, we leave presenting a security proof for this protocol as a direction for the future work.

Keywords. Oblivious Transfer, Post-Quantum Security, Superposition Attack.

1 Introduction

The oblivious transfer (OT) [Rab05] is a fundamental cryptographic primitive which allows a receiver to obtain one out of two inputs held by a sender, while the receiver learns nothing on the other input and the sender learns nothing at all (in particular, the input that the receiver receives). Later [Cre87] showed that one-out-of-two OT is equivalent to the more generic case of one-out-of-n OT, where the sender holds n inputs and the receiver receives one of them. The importance of oblivious transfer is exemplified by a result by Goldreich, Micali, and Wigderson [GMW87], where they prove that OT is MPC-complete, meaning

*Corresponding author’s email: ehsan.ebrahimi@uni.lu. Some part of work has been done at École Normale Supérieure, Paris, France.
†Work done while affiliated to ENS, CNRS, INRIA, and PSL Research University, Paris, France.
that it can be used as a building block to securely evaluate any polynomial-time computable function without any additional primitive. Studying the security of this primitive becomes then of paramount importance, especially in light of the advent of quantum computers, that numerous computer scientists and experts consider as imminent. When talking about attacks mounted through a quantum computer, there is usually some ambiguity in the terminology and its meaning. When an assumption is deemed “quantum resistant” or “post-quantum” it means that the underlying problem is supposed to be hard to solve even for a quantum computer. However, building protocols that rely on quantum resistant assumptions might not be sufficient to claim that the protocol itself cannot be broken with a quantum computer. The security essentially and crucially depends on the adversarial model that we consider. One way to look at the problem is imagining that the communication channels that connect the parties involved in the protocol are purely classical, meaning that they can transport only classical information. Indeed, in this case it seems that instantiating the protocol from quantum resistant problems is sufficient to obtain the desired proof of security.

However, in a line of works started in 2010, Kuwakado and Morii [KM10] put forward a new and more general adversarial scenario. In this model, all the communication channels controlled by the malicious parties support the transmission of quantum information while the honest parties uses classical constructions and communication. They show that 3-round Feistel cipher is distinguishable from a random permutation when the adversary has quantum access to the primitive. Subsequently, there have been extensive research works to consider this model to define the security definition for the classical cryptographic constructions and prove the security with the respected definition: quantum secure pseudo-random functions [Zha12, Zha16], encryption schemes [BZ13E, GHS16, MSI16, ATTU16, CEV20, CETU20], message authentication codes and signature schemes [BZ13a, AMRS18], hash functions [Zha15, Unr16], multi-party computation protocols [DFNS13], and etc.

Security in this general model is harder to achieve, as the adversary is no longer limited to attacking the protocol and the underlying problems with a quantum computer, but can also send messages in superposition and try to take advantage of this in order to extract information from the protocol’s transcripts. For instance in [KM10], the authors use Simon’s algorithm [Sim97] to recover the hidden (for a classical adversary) periodicity in 3-round Feistel cipher. Similarly, the Simon’s algorithm has been used in [KM12, KLLN16] to break the security of the Even-Mansour construction and some message authentication codes.

In this paper, we study the security of the OT protocols in the presence of superposition queries. The motivation to consider this general model to prove the security of OT protocols can be similar to the reasons presented in the previous works [DFNS13, ATTU16] that consider this general model: 1) A classical OT protocol can be used as a part of a quantum protocol that actively uses quantum communication. So obviously the OT protocol may be run in superposition. 2) To prove the security of some of classical protocols against a quantum adversary, intermediate games in the security proof may actually
contain honest parties that will run in superposition (for instance the security of zero-knowledge proof systems against a quantum adversary [Unr12, Wat09]). So to prove the security of such a systems, we may need to prove the security of cryptographic constructions in the presence of adversarial superposition queries.

3) The miniaturization of classical devices that may reach a quantum scale and therefore a classical protocol will have some quantum effects, etc.

1.1 Related Works

Unconditionally secure quantum OT protocols. In [Lo98, SSS15], the authors show that an unconditionally secure oblivious transfer protocol is not achievable even using quantum systems. This is in contrast to the key distribution task that is achievable with the unconditional security using quantum communication and systems [BB84]. Therefore, the alternative is to design an OT protocol that is computationally secure and obviously in the light of an adversary with the quantum computing power, the computational assumption needs to be quantum secure.

Computationally secure OT protocols against a quantum adversary. Usually, the security of OT protocols will be proven in an Universal Composability (UC) [Can01] style security model in which a real protocol will be compared with an ideal protocol. The real protocol is secure if there exists a simulator that is interacting with the ideal protocol and it successfully mimics the behaviour of the adversary. The first translation of the UC framework to the quantum setting appears in [Unr10] by Unruh. Later in [LKHB17], the authors prove the security of the oblivious transfer protocol presented in [PVW08] in the Unruh’s model. However, we emphasize that in the Unruh’s model, the adversary is not allowed to make superposition queries to the protocol and the ideal functionality measures the inputs of the adversary in the computational basis. Considering that the adversary can make the superposition queries the UC style security model need to be revisited. In [DFNS13], the authors address this problem. However, they show that simulation based security is not possible for the model that gives more power to the adversary. In more details, they show that the simulation is impossible in the model with supplied response registers by the adversary. They achieve positive result by restricting the adversary. Even considering a restricted adversary, they show that any protocol secure in this model is “non-trivial” that means the protocol can not be proven secure by running the classical simulator in superposition and the simulator has to be “more quantum”.

1.2 Our Contribution

In this paper, we study the security of OT protocols in the presence of adversarial superposition queries. We choose a different approach from [DFNS13] to study the security of OT protocols against superposition queries. We define an indistinguishability based security notion against adversarial superposition
queries.

**Why not UC-style security model?** Ideally, we may want to modify a UC-style security model to guarantee the security against adversarial superposition queries (as in [DFNS13]). This means that a real world protocol may be executed in superposition by the adversary. Therefore to have a meaningful security model, we need to consider an ideal protocol that will be run in superposition too (in other words, the ideal functionality will not measure the quantum queries of corrupted parties as in [Unr10]). Now, we will encounter obstacles to define an ideal OT protocol secure against superposition queries. To illustrate this, let assume an one-out-of-two (1-2) bit OT protocol. Roughly speaking, an ideal functionality for 1-2 bit OT protocol can be define as [CLOS02]:

- Upon receiving messages $m_0, m_1$ from the sender, store the messages.
- Upon receiving a message $b$ from the receiver, send $m_b$ to the receiver (if the messages $m_0, m_1$ are stored) and halt.

We naively run this ideal functionality in superposition considering a corrupted receiver. A corrupted receiver can send a superposition of its inputs using a quantum input register $Q_{in}$ (for instance the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)_{Q_{in}}$) to the ideal functionality. The ideal functionality needs to answer with a superposition of outputs using a quantum register $Q_{out}$ ($\frac{1}{\sqrt{2}}(|m_0\rangle + |m_1\rangle)_{Q_{out}}$, if $Q_{out}$ is initiated with 0 by the ideal functionality). At this stage, a corrupted receiver can posses a superposition of this form:

$$|\Psi\rangle := \frac{1}{\sqrt{2}}(|0\rangle_{Q_{in}} |m_0\rangle_{Q_{out}} + |1\rangle_{Q_{in}} |m_1\rangle_{Q_{out}}).$$

When $m_0 = m_1$, this state $|\Psi\rangle$ can be written as

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)_{Q_{in}} \otimes |m_0\rangle_{Q_{out}}.$$

Therefore, a measurement in the \{|+\rangle, |−\rangle\} basis on $Q_{in}$ register will return $|+\rangle$ with probability 1. But when $m_0 \neq m_1$, this measurement returns $|+\rangle$ or $|−\rangle$ with probability $\frac{1}{2}$. Therefore, overall, the corrupted receiver can guess if the inputs of the sender are the same or not with probability $\frac{3}{4}$. The situation becomes more troublesome if the output register will also be provided by the corrupted receiver. In this case the receiver can execute the Deutsch–Jozsa algorithm [DJ92] to recover if $m_0 = m_1$ or $m_0 \neq m_1$ with probability 1. Obviously, this implementation of the ideal OT functionality leaks the parity of the sender’s inputs to a corrupted receiver.

In contrast, we observe that in the real world, a corrupted receiver may not be able to produce such a superposition state as $|\Psi\rangle$. This is due to the fact that an implementation of a superposition query to a real protocol may produce some auxiliary registers that remains entangled with the input register $Q_{in}$ even
when \( m_0 = m_1 \) (see subsection 3.2 and Appendix A). So the attack sketched above will not work in this case.

So, we may encounter a situation that a real classical OT protocol remains secure against adversarial superposition queries, but, as discussed above the (classical) ideal OT functionality will be insecure against superposition queries. For this reason, in this paper we choose a different approach and study the security of OT protocols against superposition queries using an indistinguishability based security model.

**Our definition and result.** To define an indistinguishability based security definition, first, we need to discuss which party in an OT protocol may be able to break the security of the protocol with a superposition query. Note that an OT protocol is a two party protocol in which the receiver queries the sender and the sender replies to the receiver’s query. Then, the receiver extracts the targeted input from the sender’s answer. Therefore, there is no direct query from the sender to the receiver. So if we consider a malicious sender and a honest receiver, since the receiver’s query is classical all the communication will be classical. However, if we consider a malicious receiver and a honest sender, since the receiver’s query can be in superposition, then the answer of the sender is in superposition too. So a malicious quantum receiver may be able to extract some information about the inputs of the sender from the superposition state. Therefore, we consider the security of the sender against a quantum receiver that makes a superposition query in this paper. Considering an 1-2 bit OT protocol, in our security definition the sender chooses two random bits as inputs. The quantum receiver makes a quantum query to the sender and outputs a bit at the end. We say that the oblivious protocol is secure if the quantum polynomial-time receiver can guess the parity of the sender’s inputs with at most a probability negligibly bigger than \( \frac{1}{2} \). We generalize the security definition to more general OT protocols. (See subsection 3.1.) We show that if the OT functionality will be available to the receiver through an obfuscated program, the receiver can recover the parity of the sender’s inputs with high probability. (See subsection 3.3.) In subsection 3.4, we design an OT protocol based on a fully homomorphic public-key encryption scheme and show that this scheme is secure when the receiver makes a classical query, but, it is insecure when the receiver makes a quantum query. We instantiate the protocol with a lattice based public key encryption scheme that is fully homomorphic. From the discussion in subsection 3.2 we conjecture that the security against a superposition query can be achieved for some OT protocols. Specifically, in the Appendix A, we present an OT protocol in which the direct application of the superposition attack presented in this paper on the protocol will not be successful. However, we leave the proof of the quantum security as an open question and a direction for a future work.

### 1.3 Organization of The Paper

In section 2 we present some preliminaries and notations that are needed in this paper. Next, in section 3 we present our result. This section consists of
a security definition for the sender against a malicious quantum receiver that is permitted to make a superposition query (see subsection 3.1). It consists of a discussion subsection on how a malicious receiver with a superposition access can break the security of an OT protocol. In the positive side, we show that a superposition query to an OT protocol may cause some ancillary registers that are entangled with the input register and therefore they will prevent the attack to go through (an actual protocol that resists to the attack is presented in Appendix A). In the negative side, we present some cases that the attack is successful. Later in subsection 3.4, we present an OT protocol based on a fully homomorphic encryption scheme that is vulnerable when the receiver makes a superposition query. But is is secure against a malicious receiver restricted to a classical query. We finish our paper with a section on conclusion and open problems.

2 Preliminaries

Notation. We say a function $f$ from the natural numbers to the real numbers is negligible if for any positive polynomial $P$ there exists a positive integer $N$ such that for any input $n \geq N$, $|f(n)| \leq \frac{1}{P(n)}$. We use “neg($\eta$)” to show a negligible function in the security parameter $\eta$. The notation $[n]$ depicts the set $\{1, 2, \cdots, n\}$. For two bits $b_0, b_1$, the notation $[b_0 = b_1]$ indicates the parity of two bits. For two distributions $D_1$ and $D_2$ defined over the finite set $X$, the statistical distance between them is define as

$$\Delta(D_1, D_2) = \frac{1}{2} \sum_{x \in X} |\Pr[D_1 = x] - \Pr[D_2 = x]|.$$ 

We say two distributions are statistically close if the statistical distance between them is a negligible function in the security parameter.

Quantum computation. We briefly recall some basic of quantum information and computation needed for our paper below. Interested reader can refer to [NC16] for more information. For two vectors $|\Psi\rangle = (\psi_1, \psi_2, \cdots, \psi_n)$ and $|\Phi\rangle = (\phi_1, \phi_2, \cdots, \phi_n)$ in $\mathbb{C}^n$, the inner product is defined as $\langle \Psi, \Phi \rangle = \sum_i \psi_i^* \phi_i$ where $\psi_i^*$ is the complex conjugate of $\psi_i$. Norm of $|\Phi\rangle$ is defined as $||\Phi|| = \sqrt{\langle \Phi, \Phi \rangle}$. The $n$-dimensional Hilbert space $\mathcal{H}$ is the complex vector space $\mathbb{C}^n$ with the inner product defined above. A quantum system is a Hilbert space $\mathcal{H}$ and a quantum state $|\psi\rangle$ is a vector $|\psi\rangle$ in $\mathcal{H}$ with norm 1. An unitary operation over $\mathcal{H}$ is a transformation $U$ such that $UU^\dagger = U^\dagger U = I$ where $U^\dagger$ is the Hermitian transpose of $U$ and $I$ is the identity operator over $\mathcal{H}$. The computational basis for $\mathcal{H}$ consists of log $n$ vectors $|b_i\rangle$ of length log $n$ with 1 in the position $i$ and 0 elsewhere. With this basis, the unitary CNOT is defined as

$$\text{CNOT} : |m_1, m_2\rangle \rightarrow |m_1, m_1 \oplus m_2\rangle.$$
where $m_1, m_2$ are bit strings. The Hadamard unitary is defined as

$$H : |b\rangle \rightarrow \frac{1}{\sqrt{2}}(|\overline{b}\rangle + (-1)^b|b\rangle),$$

where $b \in \{0, 1\}$. The control-swap unitary is defined as

$$|b\rangle|\psi_0\rangle|\psi_1\rangle \rightarrow |b\rangle|\psi_b\rangle|\psi_{\overline{b}}\rangle,$$

for $b \in \{0, 1\}$. An orthogonal projection $P$ over $H$ is a linear transformation such that $P^2 = P = P^\dagger$. A measurement on a Hilbert space is defined with a family of orthogonal projectors that are pairwise orthogonal. An example of measurement is the computational basis measurement in which any projection is defined by a basis vector. The output of computational measurement on state $|\Psi\rangle$ is $i$ with probability $\|\langle b_i, \Psi\rangle\|^2$ and the post measurement state is $|b_i\rangle$. For two quantum systems $H_1$ and $H_2$, the composition of them is defined by the tensor product and it is $H_1 \otimes H_2$. For two unitary $U_1$ and $U_2$ defined over $H_1$ and $H_2$ respectively, $(U_1 \otimes U_2)(H_1 \otimes H_2) = U_1(H_1) \otimes U_2(H_2)$. Any classical function $f : X \rightarrow Y$ can be implemented as a unitary operator $U_f$ in a quantum computer where $U_f : |x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$. Note that it is clear that $U_f^\dagger = U_f$.

A quantum adversary has “standard oracle access” to a classical function $f$ if it can query the unitary $U_f$. When only the input register will be provided by the adversary and the output register is initiated with $0$ by the oracle, we say the adversary has “embedding oracle access” to the function. That is, the adversary has oracle access to the unitary that maps $|x, 0\rangle \rightarrow |x, f(x)\rangle$ [CETU20].

1-2 oblivious transfer protocol. An 1-2 oblivious transfer is a two party protocol between a sender and a receiver:

- The receiver on input a bit $b$ chooses a randomness $r$ and sends $R_1(b; r)$ to the sender.
- The sender on inputs $m_0, m_1$ chooses a randomness $r'$. Then it sends OT($R_1(b; r), m_0, m_1; r'$) to the receiver.
- The receiver applies a function $R_2$ to OT($R_1(b; r), m_0, m_1; r'$) to extract $m_b$.

Informally, the sender’s security will be satisfied if the input $m_{\overline{b}}$ remains secret to the receiver after execution of the protocol. The receiver’s security will be achieved if the sender does not learn the input of the receiver (the bit $b$). An 1-n oblivious transfer will be defined similarly. In this case, the sender has $n$ inputs $m_0, \cdots, m_n$ and the receiver on input $i \in [n]$ will obtain $m_i$ at the end of the protocol. The security is defined similarly.

Deterministic public-key encryption. A deterministic public key encryption scheme $E$ consists of three polynomial time algorithms (KeyGen, Enc, Dec) as follows:
On input of the security parameter, the randomized algorithm KeyGen returns a pair of keys \((pk, sk)\).

The encryption algorithm \(Enc\) is a deterministic algorithm that on inputs \(pk\) and a message \(m\), returns the ciphertext \(c := Enc_{pk}(m)\).

The decryption algorithm is (possibly randomized) algorithm that on input \(sk\) and the ciphertext \(c := Enc_{pk}(m)\) returns \(m\) (with high probability if the decryption is randomized). For an invalid ciphertext, the decryption returns \(⊥\).

**Fully homomorphic public-key encryption scheme [Gen09].** A fully homomorphic public-key encryption scheme consists of three polynomial-time algorithms (KeyGen, Enc, Dec, Evaluate) as follows:

- On input of the security parameter, the randomized algorithm KeyGen returns a pair of keys \((pk, sk)\).
- The encryption algorithm Enc is a randomized algorithm that on inputs \(pk\) and a message \(m\), chooses a randomness \(r\) and returns the ciphertext \(c := Enc_{pk}(m; r)\).
- The decryption algorithm is (possibly randomized) algorithm that on input \(sk\) and the ciphertext \(c := Enc_{pk}(m)\) returns \(m\) (with high probability if the decryption is randomized). For an invalid ciphertext, the decryption returns \(⊥\).
- The Evaluate algorithm is an (possibly randomized) algorithm that on input any \((pk, sk)\) generated by KeyGen, for any circuit \(C\) and any ciphertexts \(c_i := Enc_{pk}(m_i; r_i)\) for \(i \in [n]\), returns a ciphertext
  \[
  α = Evaluate_{pk}(C, c_1, \cdots, c_n)
  \]
  such that \(Dec_{sk}(α) = C(m_1, \cdots, m_n)\).

**Definition 1.** We say a fully homomorphic encryption scheme is “circuit-private” if for any \((pk, sk)\) generated by KeyGen, any circuit \(C\) and any ciphertexts \(c_i := Enc_{pk}(m_i; r_i)\) for \(i \in [n]\), the two distribution \(Enc_{pk}(C(m_1, \cdots, m_n))\) and \(Evaluate_{pk}(C, c_1, \cdots, c_n)\) are statistically close.

### 3 Our Result

In this section, we define a security definition that takes into consideration adversarial superposition queries made by a malicious receiver. Then, we present a discussion about how general OT protocols may be vulnerable to such queries and what will be a possible solution to avoid such attacks. Later, we present an actual protocol that will be broken in this model.
3.1 Security Definition

We define the security notion for the sender against a malicious receiver. First, we assume that the sender’s database contains two bit entries, i.e., \( m_0, m_1 \in \{0, 1\} \) and we generalize the security notion to bitstrings later. To capture the sender’s security, we define the security definition through the following game. We say a 1-2 bit OT protocol is computationally secure against a malicious quantum receiver if any polynomial-time adversary wins the following game with probability at most \( \frac{1}{2} + \operatorname{negl}(\eta) \).

**Game 1.** \( \text{OT}_{2}^{\text{bit-MR}} \): (MR stands for malicious receiver)

**Sender’s input:** The challenger picks two bits \( m_0, m_1 \) uniformly at random.

**Challenge query:** The adversary on input \( b \in \{0, 1\} \) sends two quantum registers \( Q_{\text{in}}, Q_{\text{out}} \) to the challenger. The challenger applies \( U_{\text{OT}(\cdot, m_0, m_1; r')} \) to quantum registers \( Q_{\text{in}}, Q_{\text{out}} \) and send both registers to the adversary.

**Guess:** The adversary outputs a bit \( \delta \) and wins if \( \delta = [m_0 = m_1] \).

**Definition 2.** We say an 1-2 bit OT protocol is computationally secure against a malicious quantum receiver if any polynomial-time quantum adversary wins the Game 1 with probability at most \( \frac{1}{2} + \operatorname{negl}(\eta) \).

Restricted to an adversary that is only allowed to make a classical query, the definition captures the sender’s security because the adversary can recover the bit \( m_b \) from \( \text{OT}(R_1(b), m_0, m_1; r') \) by the correctness property of the OT protocol. Then learning if the unrecovered bit is the same as the recovered bit or not should be negligibly close to \( \frac{1}{2} \). For completeness, we present the security definition restricted to a classical query below.

**Game 2.** \( \text{OT}_{2}^{\text{bit-MR-Classical Query}} \)

**Sender’s input:** The challenger picks two bits \( m_0, m_1 \) uniformly at random.

**Challenge query:** The adversary on input \( b \in \{0, 1\} \) chooses a randomness \( r \) and sends \( R_1(b; r) \) to the challenger. The challenger chooses a randomness \( r' \) and sends \( \text{OT}(R_1(b; r), m_0, m_1; r') \) to the adversary.

**Guess:** The adversary outputs a bit \( \delta \) and wins if \( \delta = [m_0 = m_1] \).

**Definition 3.** We say an 1-2 bit OT protocol is computationally secure against a malicious quantum receiver restricted to a classical query if any polynomial-time quantum adversary wins the Game 2 with probability at most \( \frac{1}{2} + \operatorname{negl}(\eta) \).

We generalize the security notion for a 1-2 bitstring oblivious transfer. We say a 1-2 bitstring OT protocol is computationaly secure against a quantum malicious receiver if any polynomial-time quantum adversary wins the following game with probability at most \( \frac{1}{2} + \operatorname{negl}(\eta) \).

**Game 3.** \( \text{OT}_{2}^{\text{bitstring-MR}} \)

**Sender’s input:** The challenger picks two bitstrings \( M_0 = (m_0^1, \ldots, m_0^\ell), M_1 = (m_1^1, \ldots, m_1^\ell) \) uniformly at random, that is, \( M_0 \leftarrow \{0, 1\}^\ell, M_1 \leftarrow \{0, 1\}^\ell \).

**Challenge query:** The adversary on input \( b \in \{0, 1\} \) sends two quantum registers \( Q_{\text{in}}, Q_{\text{out}} \) to the challenger. The challenger applies \( U_{\text{OT}(\cdot, M_0, M_1; r')} \) to
quantum registers $Q_{in}, Q_{out}$ and send both registers to the adversary.

**Guess:** The adversary outputs a bit $\delta$ and an index $i \in [\ell]$. The adversary wins if $\delta = [m_i^0 = m_i^1]$.

Roughly speaking, fulfilling the security definition above guarantees that the adversary can not learn even one bit of the unrecovered message.

Similarly we can extend the security notion to the 1-n OT protocols. We assume that the sender’s database contains $n$ bit entries, i.e., $m_1, \ldots, m_n \in \{0,1\}$. To capture the sender’s security, we define the security definition through the following game. We say a 1-n bit OT protocol is computationally secure against a malicious quantum receiver if any polynomial-time quantum adversary wins the following game with probability at most $1/2 + \text{negl}(\eta)$.

**Game 4. $OT_{n}^{\text{bit-MR}}$:**

- **Sender’s input:** The challenger picks $n$ bits $m_1, \ldots, m_n$ uniformly at random.
- **Challenge query:** The adversary on input $b \in \{0,1\}$ sends two quantum registers $Q_{in}, Q_{out}$ to the challenger. The challenger applies $U_{OT}(\cdot, m_1, \ldots, m_n; r')$ to $Q_{in}, Q_{out}$ and then sends both registers to the adversary.
- **Guess:** The adversary outputs a bit $\delta$ and two index $i \neq j \in [n]$. The adversary wins if $\delta = [m_i = m_j]$.

The definition above can be generalized to the bitstrings similarly.

**Security of the receiver.** Let $R_1$ be a randomized function that the receiver applies to its input $b$ and then sends the result to the sender. Let $\mathcal{R}$ be the set of randomness. Defining the security definition against a corrupted sender is straightforward, namely, a malicious quantum polynomial-time sender should not be able to guess the receiver’s input $b$ (that is chosen uniformly at random in the game) from $R_1(b; r)$ with probability non-negligibly more than $1/2$.

**Definition 4.** We say an OT protocol is secure against a quantum malicious sender if for any quantum polynomial-time distinguisher $D$,

$$| \Pr[D(R_1(0; r_0)) = 1; r_0 \xleftarrow{\$} \mathcal{R}] - \Pr[D(R_1(1; r_1)) = 1; r_1 \xleftarrow{\$} \mathcal{R}] | \leq \frac{1}{2} + \text{negl}(\eta).$$

### 3.2 Discussion

In this section, we implement a superposition query to an OT protocol. Note that the purpose of this section is to illustrate the ideas used in the superposition attacks on some specific OT protocols in later sections. This section also explains the challenges that appear when we want to implement such an attack on more general OT protocols and it opens a direction to design a secure OT protocol in the presence of adversarial superposition queries. First, we explain why DJ algorithm may not successfully attack all OT protocols.

**Why DJ algorithm may fail to attack an OT protocol.** Recall that any boolean function $f : X \rightarrow Y$ can be implemented efficiently as a unitary
operator $U_f : |x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle$ using quantum gates $\text{NC16}$. Let $R_1$ be a randomized function that is applied by the receiver on its input. Then, the $U_{R_1}$ is an unitary operation applied by the receiver that maps

$$|b\rangle|y\rangle \rightarrow |b\rangle|y \oplus R_1(b;r)\rangle.$$ 

The $U_{OT}$ is an unitary operation applied by the sender that maps

$$|R_1(b;r)\rangle|y\rangle \rightarrow |R_1(b;r)\rangle|y \oplus OT(R_1(b;r), m_0, m_1; r')\rangle,$$

where $m_0$ and $m_1$ are sender’s inputs. Let $R_2$ is a function applied by the receiver to extract $m_b$ from $OT(R_1(b;r), m_0, m_1; r')$. Then, $U_{R_2}$ maps

$$|b\rangle|R_1(b;r)\rangle|OT(R_1(b;r), m_0, m_1; r')\rangle|y\rangle$$

to

$$|b\rangle|R_1(b;r)\rangle|OT(R_1(b;r), m_0, m_1; r')\rangle|y \oplus m_b\rangle.$$ 

Note that $R_2 \circ OT \circ R_1$ is a function from $\{0, 1\}$ to $\{0, 1\}$ that is constant when $m_0 = m_1$ and it is balanced when $m_0 \neq m_1$. Now one may think of using the Deutsch-Jozsa (DJ) algorithm $[DJ92]$ to decide if the function is constant or balanced with the probability 1 and breaking the security in the sense of Definition 2. But this might not work for all OT protocols. The reason is that the function OT will be applied by the sender and may produce some garbage in an ancillary register. These garbage information can not be undone by the sender and therefore it may interfere the analysis of the DJ algorithm. (In Appendix A, we present an OT protocol in which such a scenario happens and the ancillary register that contains some unknown information from the receiver’s point of view will prevent the OT protocol to be attacked.)

In the following, we illustrate this by implementing the DJ algorithm to attack an OT protocol. The register $Q_{\text{out}}$ contains some unknown information from the receiver point of view and will will interfere the analysis of the DJ algorithm.

One can undone the register $Q_{\text{out}}$ by a second application of OT function as depicted in the circuit below. But since $m_0, m_1$ and the randomness $r'$ are not known to the receiver, this second application also has to be applied by the sender. Therefore, we will end up making two quantum queries to the sender that is trivially useless. We depict the circuit below that uses two queries to the
Some cases that (a variant of) DJ algorithm works. Even though the attack may not work for all OT protocols, there might be some cases that one superposition access will break the security of oblivious transfers. For instance, if the unitary operator $U_{R_2 \circ OT}$ can be applied by the receiver, then the attack will work. We present such a scenario in the subsection 3.3 using the obfuscated program of OT.

Also, we can use a variant of the DJ algorithm to attack an OT protocol that satisfies the following:

- $OT(R_1(0; r), m_0, m_1; r') = OT(R_1(1; r), m_0, m_1; r')$ when $m_0 = m_1$.

In the subsection 3.4 we design an OT protocol that satisfies the property above. We draw the circuit below to attack such an OT protocol.

We compute and analyse the output of the circuit. The output of the circuit right before applying the Hadamard operator is:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle_Q)|0\rangle_Q|励\rangle_{Q_{in}}|OT(R_1(0; r), m_0, m_1; r')\rangle_Q|励out\rangle_{Q_{out}}|励0\rangle_{Q_{Dec}} + |1\rangle_Q|励\rangle_{Q_{in}}|OT(R_1(1; r), m_0, m_1; r')\rangle_Q|励out\rangle_{Q_{out}}|励1\rangle_{Q_{Dec}}).$$

When $m_0 = m_1$. We can write the state $|\Psi\rangle$ as follows where we use only $m_0$ in the state.

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|励\rangle_{Q_{in}}|OT(R_1(0; r), m_0, m_1; r')\rangle_Q|励out\rangle_{Q_{out}}|励0\rangle_{Q_{Dec}}.$$
When $m_0 \neq m_1$. In this case, we cannot write $\ket{\Psi}$ as above and the register $Q_b$ remains entangled with $Q_{Dec}$. So the measurement returns 0 with the probability $\frac{1}{2}$ and it returns 1 with the probability $\frac{1}{2}$.

**Overall probability of success.** Therefore, overall, the attack breaks the security notion in $2$ with the probability $\frac{3}{4}$.

**Remark.** Note that in the attack above, the output register $Q_{out}$ starts with zero. Therefore, the attack works even when the malicious receiver has embedding oracle access to the sender that is a weaker oracle access compared to the standard oracle access. This shows that even measuring the output register by the sender will not help to prevent the superposition attack.

### 3.3 Superposition Attack on Obfuscated OT

Here we show that when the malicious quantum receiver possesses the obfuscated program of $\text{OT}(\cdot, m_0, m_1; r')$ where $m_0, m_1$ are the sender’s input it can break the security of OT protocol. In this case, the receiver can implement the OT protocol on a quantum device and run it on quantum inputs. The attack uses the Deutsch-Jozsa quantum algorithm [DJ92, CEMM98] that distinguishes a constant function from a balanced function by one quantum access to the function. In details, if a function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is promised to be either a constant function (it outputs 0 or 1 for all inputs) or a balanced function (half of the inputs maps to 0 and the other half maps to 1), then Deutsch-Jozsa algorithm finds if the function is constant or balanced with probability 1 and using only one quantum query to $U_f$. We illustrate how the Deutsch-Jozsa quantum algorithm can be used to break the $\text{OT}_{2\text{-MR}}$ security of the obfuscated oblivious transfer protocols. Let assume $R_1$ be the operation that will be done by the receiver on its input $b$. Let $R_2$ be the function to recover $m_b$ from $\text{OT}(R_1(b; r), m_0, m_1; r')$ that is applied by the receiver. Roughly speaking, $R_2 \circ \text{OT} \circ R_1$ is a function from $\{0, 1\}$ to $\{0, 1\}$ that is constant when $m_0 = m_1$ and it is balanced when $m_0 \neq m_1$. Therefore, one superposition query to $U_{R_2 \circ \text{OT} \circ R_1}$ can break $\text{OT}_{2\text{-MR}}$ security with the probability 1. We draw the circuit to attack in the following that is exactly the DJ algorithm.

\[
\begin{array}{c}
Q_{in} : \ket{0} \quad \begin{array}{c}
H \quad U_{R_2 \circ \text{OT} \circ R_1} \quad H \quad X
\end{array} \\
Q_{Dec} : \ket{1} \quad \begin{array}{c}
H
\end{array}
\end{array}
\]

**3.4 A Separation Example**

In this section, we present an OT protocol that is secure against a quantum adversary that is only allowed to make a classical query, but, it is insecure when the adversary makes a quantum query. The high level idea is to design an OT protocol such that $\text{OT}(R_1(0; r), m_0, m_1; r') = \text{OT}(R_1(1; r), m_0, m_1; r')$ when $m_0 = m_1$. We present an OT protocol based on a fully homomorphic lattice-based encryption scheme that satisfies the condition above.
**Protocol 1.** Let $E = (\text{KeyGen}, \text{Enc}, \text{Dec})$ be a fully homomorphic public-key encryption scheme. Let $F$ be a circuit that on input $(b, m_0, m_1)$ returns $(1 - b)m_0 + bm_1$. We define an OT protocol as follows.

- The receiver on input $b \in \{0, 1\}$ runs KeyGen to generate a pair of keys $(pk, sk)$. Then it chooses a randomness $r$ and sends $pk$ and $c_b = \text{Enc}_{pk}(b; r)$ to the sender.

- The sender chooses $r_0, r_1$ uniformly at random and computes $c'_0 = \text{Enc}_{pk}(m_0; r_0)$ and $c'_1 = \text{Enc}_{pk}(m_1; r_1)$. Then it computes $c_{\text{final}} = \text{Evaluate}_{pk}(F, c_b, c'_0, c'_1; r')$ and sends it to the receiver.

- The receiver decrypts $c_{\text{final}}$ using the secret key $sk$ to obtain $m_b$.

**Theorem 1.** On the existence of a fully homomorphic public-key encryption scheme that is circuit-private, the Protocol 1 is secure against a quantum polynomial-time malicious receiver restricted to a classical query. (In the sense of Definition 3).

**Proof.** Since the public-key encryption is circuit-private, then $\text{Evaluate}_{pk}(F, c_b, c'_0, c'_1)$ is statistically close to $\text{Enc}_{pk}(F(b, m_0, m_1))$ that is $\text{Enc}_{pk}(m_b)$. Therefore, $c_{\text{final}}$ is statistically close to $\text{Enc}_{pk}(m_b)$. This finishes the proof because $\text{Enc}_{pk}(m_b)$ is independent of the bit $m_b$. \hfill \square

**Instantiation.** We can instantiate this protocol with a lattice-based public-key encryption scheme that is fully homomorphic and it is circuit-private [Gen09, BPMW10].

### 3.4.1 Superposition Attack

We show that when the receiver makes a quantum query, the Protocol 1 will be broken. Below, we draw the circuit of the attack on the protocol. Then we compute the output of the circuit and the success probability. Note that in the circuit below $c'_0$ and $c'_1$ are classical values that the sender computes by encrypting its inputs. In other words, $c'_0 = \text{Enc}_{pk}(m_0; r_0)$ and $c'_1 = \text{Enc}_{pk}(m_1; r_1)$. Note that $c_0$ and $c_1$ will not pass to the sender. The registers $Q_b, Q_{in}, Q_{out}$ and $Q_{Dec}$ are quantum registers provided by the receiver and all are initiated by 0.
If the measurement returns 0, then the adversary outputs that the inputs of the sender are the same. Otherwise, it outputs that the inputs are different. The output of the circuit right before the application of the Hadamard operator is:

\[ |\Psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle_{Q_b}|0\rangle_{Q_{in}}|\text{Evaluate}_{pk}(F,c_0,c'_0,c'_1;r')\rangle_{Q_{out}}|m_0\rangle +
|1\rangle_{Q_b}|0\rangle_{Q_{in}}|\text{Evaluate}_{pk}(F,c_1,c'_0,c'_1;r')\rangle_{Q_{out}}|m_1\rangle). \]

Let \( R \) be a randomness that is used in \( \text{Evaluate} \) function and it depends on \( r, r_0, r_1 \) and \( r' \). We can write the state \( |\Psi\rangle \) as follows:

\[ |\Psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle_{Q_b}|0\rangle_{Q_{in}}|\text{Enc}_{pk}(m_0;R)\rangle_{Q_{out}}|m_0\rangle +
|1\rangle_{Q_b}|0\rangle_{Q_{in}}|\text{Enc}_{pk}(m_1;R)\rangle_{Q_{out}}|m_1\rangle). \]

The success probability. Note that when \( m_0 = m_1 \) we can write \( |\Psi\rangle \) as follows where we use \( m_0 \) instead of \( m_1 \).

\[ |\Psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)_{Q_b}|0\rangle_{Q_{in}}|\text{Enc}_{pk}(m_0;R)\rangle_{Q_{out}}|m_0\rangle. \]

Now, the state after applying the Hadamard operator is

\[ |0\rangle_{Q_b}|\text{Enc}_{pk}(m_0;R)\rangle_{Q_{out}}|m_0\rangle, \]

and therefore the measurement returns 0 with the probability 1. In the other hands, when \( m_0 \neq m_1 \), we can not write \( |\Psi\rangle \) as \( \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)_{Q_b} \otimes |\phi\rangle \) for some state \( |\phi\rangle \). In other words, the register \( Q_b \) is entangled with other registers and therefore, the measurement returns 0 or 1 with the probability \( \frac{1}{2} \). Overall, the adversary can break the security notion in Definition 2 with the probability \( \frac{3}{4} \).

4 Conclusion and Open Problems

In this paper, we study the security of OT protocols in a scenario when the receiver can make a quantum query to the sender. We define a security notion in this model. We design an OT protocol that is secure against a quantum malicious receiver when it is only allowed to make a classical query. But, the protocol is insecure when the receiver makes a quantum query. Our OT protocol is based on a lattice-based fully homomorphic encryption scheme. The attack works even when the malicious receiver is only allowed to provide the input register and the output register will be initiated with 0 by the sender. We present an OT protocol that resists to the attack presented in this paper, however, we leave as an open question presenting a formal proof of the security for this protocol.

We show that any 1-2 bit OT protocol in which \( \text{OT}(R_1(0;r),m_0,m_1;r') = \text{OT}(R_1(1;r),m_0,m_1;r') \) when \( m_0 = m_1 \) will be attacked. In other words, if the
output of an OT functionality is independent of $R_1(b;r)$, the OT protocol will be attacked. In contrast, when the output of an OT functionality is depend on $R_1(b;r)$, the attack presented in this paper will not work.

Acknowledgement. We would like to thank Elham Kashefi for discussions about this work.

References


A A Possibly Secure OT Protocol?

In this section, we present an OT protocol that is secure against a honest-but-
curious quantum receiver when it is only allowed to make a classical query.
Then, we show that the direct application of the superposition attack presented
in this paper will not work. Our protocol is based on a deterministic lattice-
based public-key encryption scheme. The idea is that the sender encrypts its
inputs with two public keys: a random key that is chosen by himself and a
key that is generated by the receiver. Now, the receiver is able to decrypt the
ciphertext corresponding to his public key, but, he should not be able to decrypt
the other ciphertext. (Similar idea has been used in [PVW08, DvdGMN12])

A.0.1 Our Protocol

We define the protocol abstractly and prove that it is secure if the underling
public-key encryption fulfils some properties. Then, we instantiate the protocol
with a lattice based cryptosystem fulfilling the required properties.

Protocol 2. Let $\mathcal{E} = (\text{KeyGen}, \text{Enc}, \text{Dec})$ is a deterministic public-key encryption scheme. We define an OT protocol as follows.

- The sender picks a randomly chosen public key $pk'$ from the key space and
  sends it to the receiver.

- The receiver runs the KeyGen algorithm to obtain a pair $(pk, sk)$. Then
  on input $b \in \{0, 1\}$, it sets $PK_b = pk$ and $PK_{\bar{b}} = pk \oplus pk'$. It sends $PK_0$
  to the sender.

- The sender sets $PK_1 = PK_0 \oplus pk'$. Then, it chooses two randomness
  $r_0, r_1$ and sends $c_0 = \text{Enc}_{PK_0}(m_0, r_0)$ and $c_1 = \text{Enc}_{PK_1}(m_1, r_1)$ to the
  receiver.

- The receiver decrypts $c_b$ using the secret key $sk$ to obtain $M_b$ and outputs
  the first bit of $M_b$.

Note that a malicious receiver can choose its public key $pk$ depend on $pk'$
in a way that later he be able to decrypt both ciphertexts $c_0, c_1$ partially. We
can overcome this using a commitment scheme. That is, the receiver should
commit to a public key before receiving $pk'$. Since our purpose in presenting
this protocol is to show how the ancillary registers can prevent the superposition
attack presented in this paper to go through, we skip using the commitment.
Instead, we consider the security against a honest-but-curious quantum receiver,
that is, the receiver follows the protocol.

In order that the protocol above be secure against a honest-but-curious re-
ceiver, the cryptosystem $\mathcal{E}$ has to fulfill the following properties:
1. **Ciphertext-Indistinguishability.** For a quantum polynomial-time distinguisher \( \mathcal{D} \), a generated ciphertext by a public key has to be indistinguishable from a random ciphertext. That is, for any message \( m \),

\[
\Pr[\mathcal{D}(pk, c) = 1 : (pk, sk) \leftarrow \text{KeyGen}, c := \text{Enc}_{pk}(m)] - \Pr[\mathcal{D}(pk, c^*) = 1 : c^* \leftarrow C] = \text{neg}.
\]

2. **Key-Indistinguishability.** A public key generated by KeyGen algorithm has to be statistically close to a uniformly at random key from the public key space. That is \( \Delta(PK_{\text{KeyGen}}, U) \leq \text{neg} \) where \( \Delta \) is the statistical distance between two distributions, \( PK_{\text{KeyGen}} \) is a distribution over the public key space corresponding to KeyGen and \( U \) is the uniform distribution over the public key space.

**Theorem 2.** On the existence of a public key encryption scheme that is ciphertext-indistinguishable and key-indistinguishable, the Protocol 2 is secure against a quantum honest-but-curious receiver that is only allowed to make a classical query.

**Proof.** Since the public key encryption is key-indistinguishable, we can replace \( Pk_{\overline{b}} \) with a key \( pk' \) that is generated by KeyGen. That is, \( \text{Enc}_{P K_{\overline{b}}}(m_{\overline{b}}, r_{\overline{b}}) \) is indistinguishable from \( \text{Enc}_{pk'}(m_{\overline{b}}, r_{\overline{b}}) \) for the receiver. Then, since the public key encryption scheme is ciphertext-indistinguishable, the receiver can not distinguish \( \text{Enc}_{pk'}(m_{\overline{b}}, r_{\overline{b}}) \) from a randomly chosen ciphertext \( c^* \). Therefore, the OT protocol is secure respected to the security Definition 2.

**Remark.** Note that by the key-indistinguishability property of the public key encryption scheme, the Protocol 2 is also secure against a malicious sender. However, we do not present a formal proof since we consider the malicious receiver in this paper.

**Instantiation.** We instantiate the Protocol 2 with a public-key encryption scheme that fulfills the required properties above. We use the D-PKE scheme presented in [XXZ12] that is a lattice based encryption scheme. It has been proven in Theorem 1 in [XXZ12] that the D-PKE scheme is "PRIV1-INDr" secure. It is clear that PRIV1-INDr security notion (see Definition 1 in [XXZ12]) implies ciphertext-indistinguishability defined above. Also, by Lemma 4 in [XXZ12] a public key generated by KeyGen is statistically close to a random public key. So the D-PKE scheme fulfills the key-indistinguishability property.

**A.0.2 Direct Implementation of Superposition Attack**

In this section, we show that the direct implementation of the superposition attack in subsection 3.2 on Protocol 2 does not work. We present the attack step by step in the following. Note that all the quantum registers are provided by the receiver. For simplicity, we show any zero string 0\(^n\) with 0. This means
that $|0\rangle$ can be a state of bigger size. At the beginning, the receiver prepares three quantum registers $Q_b$, $Q_{PK_0}$ and $Q_{PK_1}$. The register $Q_b$ contains the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, and the registers $Q_{PK_0}$ and $Q_{PK_1}$ contain the state $|0\rangle$. The receiver applies the following operation to these registers defined over basis.

$$|0\rangle_{Q_b}|0\rangle_{Q_{PK_0}}|0\rangle_{Q_{PK_1}} \rightarrow |0\rangle_{Q_b}|pk\rangle_{Q_{PK_0}}|pk \oplus pk'\rangle_{Q_{PK_1}},$$

and

$$|1\rangle_{Q_b}|0\rangle_{Q_{PK_0}}|0\rangle_{Q_{PK_1}} \rightarrow |1\rangle_{Q_b}|pk \oplus pk'\rangle_{Q_{PK_0}}|pk\rangle_{Q_{PK_1}}.$$ 

Next, the sender encrypts its inputs $m_0, m_1$ and stores them in the registers $Q_{out0}, Q_{out1}$ provided by the receiver.

$$|0\rangle_{Q_b}|pk\rangle_{Q_{PK_0}}|pk \oplus pk'\rangle_{Q_{PK_1}}|0\rangle_{Q_{out0}}|0\rangle_{Q_{out1}} \rightarrow$$

$$|0\rangle_{Q_b}|pk\rangle_{Q_{PK_0}}|pk \oplus pk'\rangle_{Q_{PK_1}}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}.$$ 

and

$$|1\rangle_{Q_b}|pk \oplus pk'\rangle_{Q_{PK_0}}|pk\rangle_{Q_{PK_1}}|0\rangle_{Q_{out0}}|0\rangle_{Q_{out1}} \rightarrow$$

$$|1\rangle_{Q_b}|pk \oplus pk'\rangle_{Q_{PK_0}}|pk\rangle_{Q_{PK_1}}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}.$$ 

Now the receiver uses its secret key to decrypt and outputs the but $m_b$. It prepares two registers $Q_{Dec0}$ and $Q_{Dec1}$ containing $|0\rangle$. We show this operation below.

$$|0\rangle_{Q_b}|pk\rangle_{Q_{PK_0}}|pk \oplus pk'\rangle_{Q_{PK_1}}$$

$$|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}} \rightarrow$$

$$|0\rangle_{Q_b}|pk\rangle_{Q_{PK_0}}|pk \oplus pk'\rangle_{Q_{PK_1}}$$

$$|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}}.$$ 

and

$$|1\rangle_{Q_b}|pk \oplus pk'\rangle_{Q_{PK_0}}|pk\rangle_{Q_{PK_1}}$$

$$|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}} \rightarrow$$

$$|1\rangle_{Q_b}|pk \oplus pk'\rangle_{Q_{PK_0}}|pk\rangle_{Q_{PK_1}}$$

$$|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}}.$$ 

Note that since the receiver knows $pk$ and $pk'$, it can undo the registers $Q_{PK_0}$ and $Q_{PK_1}$, and gets back $|0\rangle$. Therefore, we can consider the following states.

$$|0\rangle_{Q_b}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}} \rightarrow$$

$$|0\rangle_{Q_b}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}}.$$
and

\[ |1\rangle_{Q_b}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|0\rangle_{Q_{Dec0}}|0\rangle_{Q_{Dec1}} \rightarrow |1\rangle_{Q_b}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|\bot\rangle_{Q_{Dec0}}|m_1\rangle_{Q_{Dec1}}. \]

Next, the receiver can apply the control-swap unitary to registers $Q_b$ and $Q_{out0}, Q_{out1}$ and $Q_{Dec0}, Q_{Dec1}$ where the control register is $Q_b$. We show this operation below.

\[ |0\rangle_{Q_b}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|m_0\rangle_{Q_{Dec0}}|\bot\rangle_{Q_{Dec1}} \rightarrow |0\rangle_{Q_b}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|m_0\rangle_{Q_{Dec0}}|\bot\rangle_{Q_{Dec1}} \]

and

\[ |1\rangle_{Q_b}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|\bot\rangle_{Q_{Dec0}}|m_1\rangle_{Q_{Dec1}} \rightarrow |1\rangle_{Q_b}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|m_0\rangle_{Q_{Dec0}}|\bot\rangle_{Q_{Dec1}}. \]

At this point, the final state can be written as follows when we remove the last register that contains $\bot$ in the presentation.

\[ |\Psi\rangle := \frac{1}{\sqrt{2}}(|0\rangle_{Q_b}|\text{Enc}_{pk}(m_0, r_0)\rangle_{Q_{out0}}|\text{Enc}_{pk\oplus pk'}(m_1, r_1)\rangle_{Q_{out1}}|m_0\rangle_{Q_{Dec0}} + |1\rangle_{Q_b}|\text{Enc}_{pk}(m_1, r_1)\rangle_{Q_{out1}}|\text{Enc}_{pk\oplus pk'}(m_0, r_0)\rangle_{Q_{out0}}|m_1\rangle_{Q_{Dec1}}) \]

Now if we apply the Hadamard unitary to the $Q_b$ register and then measure the $Q_b$ register in the computational basis, the probability of getting 0 is equal to the probability of getting 1 in both cases of $m_0 = m_1$ and $m_0 \neq m_1$. This is due to the fact that the randomness $r_0$ and $r_1$ are not equal with high probability and therefore we can not write

\[ |\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)_{Q_b} \otimes |\Phi\rangle \text{ for some state } |\Phi\rangle, \]

for both cases of $m_0 = m_1$ and $m_0 \neq m_1$. Therefore, the direct application of the attack does not work.