A Game-Theoretic Perspective on Oblivious Transfer

Abstract

Asharov, Canetti, and Hazay (Eurocrypt 2011) studied how game-theoretic concepts can be used to capture the cryptographic properties of correctness, privacy, and fairness in two-party protocols in the presence of fail-stop adversaries. Based on their work, we characterize the properties of "two-message" oblivious transfer protocols in terms of game-theoretic concepts. Specifically, we present a single two-player game defined using a two-message OT protocol such that the OT protocol satisfies the cryptographic properties of correctness and privacy in the presence of *malicious* adversaries if and only if the strategy of honestly following the protocol is in a Nash equilibrium in the game.

1 Introduction

1.1 Background

Cryptographic protocols are designed for parties who follow them to guarantee some properties such as correctness and privacy. In many cases, such properties are discussed in a way that if some player honestly follow the protocol, she can achieve some desirable properties even if some of participants of the protocol are controlled by an adversary. Game theory studies the behavior of "rational" parties interacting with each other. One of the interplay between cryptography and game theory is to design cryptographic protocols in the presence of rational parties, who are neither honest nor malicious. A line of work on *rational secret sharing* [10, 14, 1, 7, 12, 13, 16, 15, 3, 6] is in this direction.

Recently, Asharov, Canetti, and Hazay [2] have studied how game-theoretic concepts can be used to capture the cryptographic properties such as correctness, privacy, and fairness. In particular, they characterize these properties in terms of a game theoretic concept, Nash equilibrium, in the setting of secure two-party protocols in the presence of fails-stop adversaries. Cryptographic properties of two-party protocols are characterized in a way that the protocol satisfies a "certain" cryptographic property if and only if the strategy of honestly following the protocol is a Nash equilibrium in a "certain" game defined using the protocol in which the player has a "certain" payoff function. Regarding the cryptographic properties of correctness and privacy, they showed a game together with a payoff function that is equivalent to these properties. Regarding fairness, they introduced a new cryptographic fairness that has an equivalent game-theoretic characterization.

1.2 This Work

Based on the work of Asharov et al. [2], we further explore how the crypotographic properties can be captured by game-theoretic concepts. In particular, we characterize the properties of *oblivious transfer*, which is one of the well-studied two-party protocol, in terms of game-theoretic concepts.

Oblivious transfer (OT) is a protocol between a sender and a receiver. The sender has two inputs x_0 and x_1, and the receiver has an input $c \in \{0, 1\}$. After running the protocol, the receiver obtains x_c while the sender obtains nothing. Privacy is considered both for the sender and the receiver. The sender's privacy requires that the receiver learns nothing about x_{1-c}. The receiver's privacy requires that the sender learns nothing about the choice bit c. In this work, we present a game defined using a *two-message* OT protocol together with the payoff functions of the players, a sender and a receiver, such that the OT protocol satisfies the cryptographic properties of correctness and privacy in the presence of *malicious* adversaries *if and only if* the strategy of honestly following the protocol is in a Nash equilibrium in the game.

Our characterization of two-message OT protocols has several advantages compared to the work of [2]. First, the game defined in our work is played between two rational players while every game defined in [2] is "essentially" played by a single rational player. For example, in [2], the privacy of a protocol is characterized by two games, one for the privacy of player 1 and the other for the privacy of player 2. In each game, the payoff function of only one player is essentially considered. Since game-theoretic concepts are of significant meaning in the presence of multiple rational players, characterization by a single game between two rational players is preferable. Second, we can characterize both correctness and privacy by a single game while each property is characterized by different games in [2]. Third, we consider the setting in the presence of malicious adversaries, who can take any malicious action in the protocol and are stronger than fail-stop adversaries, who are allowed to take only two actions, "continue" and "stop", in each round.

Since we can present a single game between two players that characterize the cryptographic properties of two-message OT, several variations of this game can be considered from a game-theoretic perspective. For example, we can consider games with more complex payoff functions and other solution concepts than Nash equilibrium.

The reason for focusing on "two-message" OT is that there exists an indistinguishability-based definition of privacy for two-message OT in the presence of malicious adversaries [11, 8]. Although the ideal/real simulation paradigm provides strong and desirable security in the cryptographic contexts, the indistinguishability-based definition is fit for a game-theoretic framework. In the indistinguishability-based privacy, a player is asked to predict which of the two inputs of the other player is used in the protocol. Then the payoff of the player can be explicitly defined in a way such that she obtains higher payoff if the prediction is correct, and lower payoff otherwise.

2 Models and Definitions

We review some basic definitions to capture OT protocols as well as the solution concepts from game theory.

A function μ is called negligible if for any polynomial p, there exists a value $N \in \mathbb{N}$ such that for all $n > N$ it holds that $\mu(n) < 1/p(n)$. We describe some negligible function on n as $negl(n)$. Let $X = \{X(a, n)\}_{n\in\mathbb{N}, a\in\{0,1\}^*}$ and $Y = \{Y(a, n)\}_{n\in\mathbb{N}, a\in\{0,1\}^*}$ be distribution ensembles. Then, we say that X and Y are computationally indistinguishable, denoted $X \overset{\mathrm{c}}{\equiv} Y$, if for every non-uniform probabilistic polynomial time (PPT) distinguisher D, it holds that

$$|\Pr[D(X(n, a)) = 1] - \Pr[D(Y(n, a)) = 1]| \leq negl(n).$$

2.1 Cryptographic Security of Oblivious Transfer

An OT protocol π is modeled as a pair of interacting Turing machines. We write $\pi = (S^\pi, R^\pi)$ where S^π is the sender's algorithm and R^π is that of the receiver. We focus on probabilistic polynomial time algorithm S^π and deterministic polynomial time algorithm R^π. The sender's input is a pair of two secret messages (x_0, x_1) and the receiver's is a secret choice bit c. After the completion of π, the receiver outputs the message x_c wheres the sender outputs nothing. We write this execution $\pi(S^\pi(x_0, x_1), R^\pi(c)) = (\lambda, x_c)$. To simplify the analysis, we consider machines that are polynomial in a security parameter, rather than in the length of their inputs. In this paper, $n \in \mathbb{N}$ represents the security parameter, and we omit it when it is obvious.

As mentioned in the introduction, we restrict our focus on two message OT protocols.

We define the view and output of protocols formally. These definitions are widely used in the field of 2-party computation [4, 11].

Definition 2.1 (View). *Let $\pi = (S^\pi, R^\pi)$ be an OT protocol. The* view *of the sender during the execution of π on input pair (x_S, x_R), when the sender and the receiver use S and R as their algorithms respectively, is denoted* $\mathsf{view}_{\pi,S}(S(x_S), R(x_R))$ *and equals (x_S, r_S, m_R, m_S), where r_S is the random coins of the sender, m_R represents the message which the receiver send, and m_S represents the message which the sender send. The* view *of the receiver is defined analogously.*

Definition 2.2 (Output). *Let $\pi = (S^\pi, R^\pi)$ be an OT protocol. The* output *of the receiver after the execution of π on input pair (x_S, x_R) when the sender and the receiver use S and R as their algorithms respectively, is denoted* $\mathsf{output}_{\pi,S}(S(x_S), R(x_R))$.

As mentioned in the introduction, our security definitions is based on computational indistinguishability. Similar definitions are considered in previous works [11, 8]. There are 3 notions, privacy for the receiver, privacy for the sender, and correctness. If a two message OT protocol satisfies all of these notions, then it is called secure in the presence of malicious adversaries.

Definition 2.3 (Cryptographic security for OT protocols). *An OT protocol $\pi = (S^\pi, R^\pi)$ is said to be secure protocol if the following holds:*

Privacy for the receiver *If for every probabilistic polynomial time algorithm S^* and every pair of strings (x_0, x_1) such that $|x_0| = |x_1|$, it holds that*

$$\mathsf{view}_{\pi,S}(S^*(x_0, x_1), R^\pi(0)) \overset{\mathsf{c}}{\equiv} \mathsf{view}_{\pi,S}(S^*(x_0, x_1), R^\pi(1)).$$

Privacy for the sender *If for every deterministic polynomial time algorithm R^* and every tuple of strings (x_0, x_1, x) such that $|x_0| = |x_1| = |x|$, there exists a function Choice such that if $Choice(R^*) = 1$ then*

$$\mathsf{view}_{\pi,R}(S^\pi(x_0, x_1), R^*(c)) \overset{\mathsf{c}}{\equiv} \mathsf{view}_{\pi,R}(S^\pi(x_0, x), R^*(c)),$$

and if $Choice(R^*) = 0$ then

$$\mathsf{view}_{\pi,R}(S^\pi(x_0, x_1), R^*(c)) \overset{\mathsf{c}}{\equiv} \mathsf{view}_{\pi,R}(S^\pi(x, x_1), R^*(c)).$$

Correctness *If for every sender's two input strings $x_0, x_1 \in \{0,1\}^*$ such that $|x_0| = |x_1|$, and for every receiver's choice bit c, it holds that:*

$$\Pr[\mathsf{output}_{\pi,R}(S^\pi(x_0, x_1), R^\pi(c)) = x_c] \geq 1 - negl(n)$$

2.2 Game-Theoretic Concepts

To capture properties and security of OT protocols in the field of game theory, we define the concepts of games, utility functions, and a solution concept called Nash equilibrium. Our definitions are similar to the ones previous works [5, 9].

First, we define non-cooperative 2-player games with incomplete information. Since the players of OT protocols do not know any information about the other player's input, and they require the secrecy of their inputs, the implementation of OT protocols can be defined in terms of non-cooperative 2-player games with incomplete information. Formally, we define such games as follows.

Definition 2.4 (non-cooperative 2-player game with incomplete information)**.** *For some N such that $|N| = 2$, a 2-player Bayesian game is described as $\Gamma = (N, \{A_i, T_i, u_i\}_{i \in N}, \mathcal{D})$, where*

- *N is a set of players. (Let $N = \{0, 1\}$ for simplicity.)*
- *A_i is a set of actions for player $i \in N$. Let $A = A_0 \times A_1$.*
- *T_i is a set of types for player $i \in N$. Let $T = T_0 \times T_1$.*
- *$u_i : A \times T \to \mathbb{R}$ is the utility function for player $i \in N$.*
- *$\mathcal{D}$ is the probability distribution over T. The tuples of the types for each player (t_0, t_1) happens with probability $p_{\mathcal{D}}(t_0, t_1)$ which is defined by $\mathcal{D}$. Each player $i \in N$ with the type t_i believes that the type t_{1-i} of the other player occur with probability $p_{\mathcal{D}}(t_{1-i}|t_i)$.*

$\sigma_i : T_i \to A_i$ is called a strategy for player i. Each player's action could be decided with respect to its type and its strategy.

If the player i knows the types of the both players, it can calculate its own utility with respect to any pair of their strategies. For example, when the pair of their types is (t_0, t_1) and each player has strategy σ_0 and σ_1 respectively, the utility of the player i after the completion of the game is $u_i(\sigma_0(t_0), \sigma_1(t_1), t_0, t_1)$. We write it as $u_i(\sigma_0(t_0), \sigma_1(t_1))$ for simplicity.

However, since each player knows its own type, but do not know the other's, even after the execution of the protocol, each player can't calculate its own utility. Thus, we use the following concept of the expected utility.

Utility functions are the "indicator" when the players select their strategies. Each players rationally select their strategies based on their utility functions, that is, they select the strategies with which they can get the highest value of utility. However, in Bayesian games, each party can not know the other's types or strategies, and they cannot compute their own value of the utility, we use the expectation value of the utility.

Definition 2.5 (Expected utility for 2-player Bayesian games)**.** *Let $\Gamma = (N, \{A_i, T_i, u_i\}_{i \in N}, \mathcal{D})$ be a 2-player Bayesian game, and let $N = \{0, 1\}$. The expected utility of the player 0 with type t_0 for a pair of their strategies (σ_0, σ_1) on the game Γ is*

$$u_0(\sigma_0, \sigma_1) = \sum_{t_1 \in T_1} u_0(\sigma_0(t_0), \sigma_1(t_1))\, p_{\mathcal{D}}(t_{1-i}|t_i).$$

The expected utility of the player 0 is defined analogously.

In a Bayesian game, rational players are seeking to maximize their expected utility. We use Bayesian Nash equilibrium as a solution concept of 2-player Bayesian games. To compensate for the small inevitable imperfections of cryptographic constructs, we take no account of negligible differences of values.

Definition 2.6 (Nash equilibrium for a 2-player Bayesian game)**.** *Let* $\Gamma = (N, \{A_i, T_i, u_i\}_{i \in N}, \mathcal{D})$ *be a 2-player Bayesian game, and let* $N = \{0, 1\}$*. A pair of strategies* (σ_0, σ_1) *is in* N*ash equilibrium if for every player* $i \in N$ *and every strategy* σ'_i *it can take, it holds that*

$$u_i(\sigma_0, \sigma_1) \geq u_i(\sigma''_0, \sigma''_1) - negl(n)$$

where $\sigma''_i = \sigma'_i$ *and* $\sigma''_{1-i} = \sigma_{1-i}$*.*

we use "history" to capture the implementation of protocols in a similar way as in cryptography. Formally, history is defined as follows.

Definition 2.7 (History)**.** *Let* $\pi = (S^\pi, R^\pi)$ *be an OT protocol. The* history *of the sender during the execution of* π *on the input pair* (x_S, x_R) *and the pair of the strategies* (S, R) *is denoted* $\mathsf{history}_{\pi,S}(S(x_S), R(x_R))$*, and equals* (x_S, r_S, m_R, m_S)*, where* r_S *is the random coins of the sender,* m_R *represents the message which the receiver send, and* m_S *represents the message which the sender send. The* history *of the receiver is defined analogously.*

3 Game Theoretic Perspective on Oblivious Transfer

3.1 Game Theoretic Security of Oblivious Transfer

OT protocols are done by 2 players, the sender and the receiver, and the possible actions of each player are defined by the model (e.g. malicious model). Moreover, the types of each player correspond to its input. To capture OT in the field of game theory, we should define types and utility functions of the players', and distribution for games for OT protocols.

We formally define games to capture security of OT protocols. In this game, the sender and the receiver execute an OT protocol, and after that, they guess the other's input from 2 candidates. Thus, both parties choose 2 algorithms, one of which is to execute π and the other is a guess algorithm, as a strategy to participate in the game. Here, the guess algorithm runs on the party's history and 2 candidates of the other's input.

Definition 3.1 (Games to capture security of OT protocols)**.** *Let* $\pi = (S^\pi, R^\pi)$ *be an OT protocol. On input* $((S, G_S), (R, G_R), \mathcal{D})$ *where*

- S *is a probabilistic polynomial time strategy of the sender to execute the protocol* π*,*
- R *is a deterministic polynomial time strategy of the receiver to execute the protocol* π*,*
- G_S *and* G_R *are algorithms that output a binary value,*
- $\mathcal{D} = \{D_{x_0,x_1,x}\}_{x_0,x_1,x \in \{0,1\}^n}$*,*

the game Game^π *runs as follows:*

1. *Choose* $D_{x_0,x_1,x}$ *from the distribution* $\mathcal{D}$*.*

2. *If the receiver's algorithm R is equal to R^{π}, c is chosen from $\{0,1\}$ uniformly at random. Otherwise, a function Choice computes c from the receiver's algorithm R.*

3. *Let $X^0 = (x_0, x_1)$. If $c = 0$ then let $X^1 = (x, x_1)$, and if $c = 1$ then let $X^1 = (x_0, x)$.*

4. *Choose a bit b from $\{0,1\}$ uniformly at random.*

5. *Execute $\pi(S(X^b), R(c))$. The receiver outputs $\mathsf{output}(S,R)$ and $\mathsf{fin}_R(S,R)$. $\mathsf{fin}_R(S,R) = 1$ if the receiver receives the sender's message.*

6. *Let $\mathsf{fin}_S(S,R) = 0$ if the sender sends a message to the receiver. Let $\mathsf{output}(S,R)$ be the output of the receiver, $\mathsf{fin}(S,R) = \mathsf{fin}_S(S,R) \wedge \mathsf{fin}_R(S,R)$ $\mathsf{guess}_S((S,R),G_S) = G_S(\mathsf{history}_{\pi,S}(S(X^b),R(c)))$, and $\mathsf{guess}_R((S,R),G_R) = G_R(\mathsf{history}_{\pi,R}(S(X^b),R(c)),X^0,X^1)$.*

If a protocol aborts by some input, we assume that the protocol outputs $\mathsf{fin}(S,R) = 1$. We can easily modify any protocol into this type.

To investigate whether a protocol π has a certain property, we will check whether for an algorithms G such that outputs a bit uniformly at random, a pair of strategies $((S^{\pi}, G), (R^{\pi}, G))$ is in Nash equilibrium for a game with respect to a set of utility functions.

Definition 3.2 (Nash equilibrium)**.** *For a pair of utility functions (u_S, u_R), we say a pair of strategies $((S^*, G_S^*), (R^*, G_R^*))$ is in Nash equilibrium if for every pair of strategies $((S, G_S), (R, G_R))$, it holds that*

$$u_S((S^*, G_S^*), (R^*, G_R^*)) \geq u_S((S, G_S), (R^*, G_R^*)) - negl(n),$$

and

$$u_R((S^*, G_S^*), (R^*, G_R^*)) \geq u_R((S^*, G_S^*), (R, G_R)) - negl(n).$$

To capture the security of OT protocols, we say a pair of utility functions. u_S and u_R represents a utility function for the sender and the receiver, respectively. Intuitively, each utility function consist of 3 elements, i.e., a part to protect secrecy, run the protocol correctly, and guess the other's secret.

Definition 3.3 (Pair of utility functions for security)**.** *Let $\pi = (S^{\pi}, R^{\pi})$ be an OT protocol. Let α_S, β_S, γ_R, α_R, β_R, γ_R be positive constants. The pair of utility functions for security is denoted*

$\mathcal{U} = (u_S, u_R)$. The utility functions on a pair of strategies $((S, G_S), (R, G_R))$, are defined by:

$$\begin{aligned}
&u_S((S,G_S),(R,G_R)) \\
&= \; -\alpha_S \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{guess}_R((S,R),G_R)=b] - 1/2 \right) \\
&\quad +\beta_S \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{fin}(S,R)=0 \vee \mathsf{output}(S,R)=x_c] - 1 \right) \\
&\quad +\gamma_S \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{guess}_S((S,R),G_S)=c] - 1/2 \right) \\
&u_R((S,G_S),(R,G_R)) \\
&= \; -\alpha_R \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{guess}_S((S,R),G_S)=c] - 1/2 \right) \\
&\quad +\beta_R \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{fin}(S,R)=0 \vee \mathsf{output}(S,R)=x_c] - 1 \right) \\
&\quad +\gamma_R \left(\Pr_{x_0,x_1,x\in\mathcal{D}}[\mathsf{guess}_R((S,R),G_R)=b] - 1/2 \right)
\end{aligned}$$

Using this game Game^π and the pair of utility functions $\mathcal{U} = (u_S, u_R)$, we define game-theoretic security for OT protocols. Intuitively, if both players can get the highest utility when they act "honestly," that is, if a pair of strategies which both players act "honestly" is in Nash equilibrium, then the protocol is called secure. Formally, we define as follows.

Definition 3.4 (Game-theoretic security for OT protocols)**.** *Let $\pi = (S^\pi, R^\pi)$ be an OT protocol. We say that π is* game-theoretically secure *if there exist a function Choice such that the pair of strategies $((S^\pi, G), (R^\pi, G))$ is in Nash equilibrium for a game Game^π with respect to the pair of utility functions $\mathcal{U}$.*

Asharov et al. [2] defined game-theoretic security of 2-party computation in the presence of fail-stop adversaries. We can not discuss the security in the presence of malicious adversaries with their definition. Since the definition "strategies which both players act honestly is in Nash equilibrium" assume that the other player acts honestly, and discuss if a player can get the highest utility when she acts honestly, too. However,with their utility functions, the players have no incentive to act maliciously since they can not get a higher utility even if they get the other's secret. Thus, we give the utility functions an incentive to obtain the other's secret. For example, whether a protocol π is private for the receiver in the presence of malicious adversaries depends on the sender's utility function. If π is private for the receiver, sender cannot get the receiver's secret no matter how she act. Otherwise, there are some actions other than the honest behavior which raise the utility.

We introduce the theorem and the proof in the next section.

3.2 Equivalence of the Two Security Definitions

In this section, we show the equivalence between the cryptographic security we introduced in the section 2. and the game-theoretic security we newly defined in the previous section.

Here we sketch the outline of the proof. First, to prove that cryptographic security implies game-theoretic security, assume that there exists a strategy of a player which can reach a higher utility than both players act honestly. Thus, at least one term of the utility functions reach a higher value than honest strategies, and we show that cryptographic notion related to the term does not hold.

Secondly, to prove that game-theoretic security implies cryptographic security, assume that a certain cryptographic property does not hold and show that the related term of the utility function on honest strategies is less than on the default strategies. However, it is insufficient. Because if the dropped value are compensated by another term, then the honest strategies can be in Nash equilibrium. We show that any other term can not compensate the dropped value, and conclude that if a protocol is not cryptographically secure then it is not game-theoretically secure.

Theorem 3.5. *Let $\pi = (S^\pi, R^\pi)$ be an OT protocol. π is cryptographically secure if and only if π is game-theoretically secure.*

Proof. We begin with the proof that cryptographic security implies game-theoretic security. Assume that an OT protocol $\pi = (S^\pi, R^\pi)$ is not game-theoretically secure, and prove that nor is π cryptographically secure.

There exist the following two cases.

First, assume that there exist a sender's strategy (S^*, G^*_S) and a non-negligible function ϵ such that

$$u_S((S^*, G^*_S), (R^\pi, G)) > u_S((S^\pi, G), (R^\pi, G)) + \epsilon(n).$$

Then, as least one of the next formulae holds where ϵ_1, ϵ_2 and ϵ_3 are non-negligible functions:

$$\Pr[\mathsf{guess}_R((S^*, R^\pi), G) = b] < \Pr[\mathsf{guess}_R((S^\pi, R^\pi), G) = b] - \epsilon_1(n) \tag{1}$$

$$\begin{aligned}&\Pr[\mathsf{fin}(S^*, R^\pi) = 0 \vee \mathsf{output}(S^*, R^\pi) = x_c]\\ &\quad > \Pr[\mathsf{fin}(S^\pi, R^\pi) = 0 \vee \mathsf{output}(S^\pi, R^\pi) = x_c] + \epsilon_2(n)\end{aligned} \tag{2}$$

$$\Pr[\mathsf{guess}_S((S^*, R^\pi), G^*_S) = c] > \Pr[\mathsf{guess}_S((S^\pi, R^\pi), G) = c] + \epsilon_3(n) \tag{3}$$

When formula 1 holds, we have:

$$\begin{aligned}\Pr[\mathsf{guess}_R((S^\pi, R^\pi), G) = b] &> \Pr[\mathsf{guess}_R((S^*, R^\pi), G) = b] + \epsilon_1(n)\\ &\geq \min_{S^*}(\Pr[\mathsf{guess}_R((S^*, R^\pi), G) = b]) + \epsilon_1(n)\\ &= \Pr[\mathsf{guess}_R((S^{\mathsf{default}}, R^\pi), G) = b] + \epsilon_1(n)\\ &= 1/2 + \epsilon_1(n),\end{aligned}$$

where S^{default} represents the sender's strategy not to participate in the protocol. This means that π is not cryptographically private for the sender.

When formula 2 holds, we have:

$$\begin{aligned}&\Pr[\mathsf{fin}(S^\pi, R^\pi) = 0 \vee \mathsf{output}(S^\pi, R^\pi) = x_c]\\ &\quad < \Pr[\mathsf{fin}(S^*, R^\pi) = 0 \vee \mathsf{output}(S^*, R^\pi) = x_c] - \epsilon_2(n)\\ &\quad \leq \max_{S^*}(\Pr[\mathsf{fin}(S^*, R^\pi) = 0 \vee \mathsf{output}(S^*, R^\pi) = x_c]) - \epsilon_2(n)\\ &\quad = \Pr[\mathsf{fin}(S^{\mathsf{default}}, R^\pi) = 0 \vee \mathsf{output}(S^{\mathsf{default}}, R^\pi) = x_c] - \epsilon_2(n)\\ &\quad = 1 - \epsilon_2(n).\end{aligned}$$

This means that π is not cryptographically correct.

And when formula 3 holds, we have:

$$\begin{array}{rcl}\Pr[\mathsf{guess}_S((S^*,R^\pi),G^*_S)=c] & > & \Pr[\mathsf{guess}_S((S^\pi,R^\pi),G)=c]+\epsilon_3(n)\\ & \geq & \min_{R^*}(\Pr[\mathsf{guess}_S((S^\pi,R^*),G)=c]+\epsilon_3(n)\\ & = & \Pr[\mathsf{guess}_S((S^\pi,R^{\mathsf{default}}),G)=c]+\epsilon_3(n)\\ & = & 1/2+\epsilon_3(n)\end{array}$$

This means that π is not cryptographically private for the receiver.

Therefore, π is not cryptographically secure.

Secondly, we assume that there exist a receiver's strategy (R^*,G^*_R) and a non-negligible function ϵ such that

$$u_R((S^\pi,G),(R^*,G^*_R))>u_S((S^\pi,G),(R^\pi,G))+\epsilon(n).$$

In this case, we can prove that π is not cryptographically secure in a similar way as the first case.

That is, if $\pi=(S^\pi,R^\pi)$ is not game-theoretically secure, then π is not cryptographically secure.

We now turn to the proof in which game-theoretic security implies cryptographic security. Assuming that an OT protocol $\pi=(S^\pi,R^\pi)$ is not cryptographically secure, we prove that nor is π game-theoretically secure.

First, assume that π is not cryptographically correct. Then, we have

$$\Pr[\mathsf{output}_{\pi,R}(S^\pi(x_0,x_1),R^\pi(c))=x_c]<1-\epsilon(n)$$

for a non-negligible function ϵ. Thus,

$$\begin{array}{rl} & \Pr[\mathsf{fin}(S^\pi,R^\pi)=0\vee\mathsf{output}(S^\pi,R^\pi)=x_c]\\ = & \Pr[\mathsf{output}(S^\pi,R^\pi)=x_c]\\ < & 1-\epsilon(n)\end{array}$$

Let S^{stop} be a strategy of the sender which stops the protocol after receiving the receiver's message, and we have:

- $\Pr[\mathsf{guess}_R((S^{\mathsf{stop}},R^\pi),G)=b]=1/2\leq\Pr[\mathsf{guess}_R((S^\pi,R^\pi),G)=b]$
- $\Pr[\mathsf{fin}(S^{\mathsf{stop}},R^\pi)=0\vee\mathsf{output}(S^{\mathsf{stop}},R^\pi)=x_c]$
  $=\Pr[\mathsf{fin}(S^{\mathsf{stop}},R^\pi)=0]=1>\Pr[\mathsf{output}(S^\pi,R^\pi)=x_c]+\epsilon(n)$
  $=\Pr[\mathsf{fin}(S^\pi,R^\pi)=0\vee\mathsf{output}(S^\pi,R^\pi)=x_c]+\epsilon(n)$
- $\Pr[\mathsf{guess}_S((S^{\mathsf{stop}},R^\pi),G)=c]=\Pr[\mathsf{guess}_S((S^\pi,R^\pi),G)=c]$

Therefore, we have

$$u_S((S^{\mathsf{stop}},G),(R^\pi,G))>u_S((S^\pi,G),(R^\pi,G))+\beta_S\,\epsilon(n),$$

which means that the pair $((S^\pi,G),(R^\pi,G))$ is not in Nash equilibrium.

Secondly, we assume that π is cryptographically correct, and π is not cryptographically private for the receiver. Then, there exist a sender's strategy (S^*,G^*_S) and a non-negligible function ϵ such that

$$\Pr[\mathsf{guess}_S((S^*,R^\pi),G^*_S)=c]>1/2+\epsilon(n).$$

From this formula, we can say $\Pr[\mathsf{guess}_S((S^\pi, R^\pi), G_S^*) = b] > 1/2 + \epsilon(n)$, since the sender's guess is based on the receiver's message, and the success probability can reach the highest value any time after receiving the receiver's message. Thus, we have $\Pr[\mathsf{guess}_S((S^\pi, R^\pi), G_S^*) = b] - \epsilon(n) > 1/2 = \Pr[\mathsf{guess}_S((S^\pi, R^\pi), G) = b]$ Thus, we have

$$u_S((S^\pi, G_S^*), (R^\pi, G)) > u_R((S^\pi, G), (R^\pi, G)) + \gamma_S\, \epsilon(n),$$

which means that the pair $((S^\pi, G), (R^\pi, G))$ is not in Nash equilibrium.

Finally, we assume that π is cryptographically correct, π is cryptographically private for the receiver, and π is not cryptographically private for the sender. Then, there exist a receiver's strategy (R^*, G_R^*) and a non-negligible function ϵ such that

$$\Pr[\mathsf{guess}_R((S^\pi, R^*), G_R^*) = b] > 1/2 + \epsilon(n).$$

If $R^* = R^\pi$ then we discuss in a similar way as the second case. Now we assume $R^* \neq R^\pi$, and we have:

- $\Pr[\mathsf{guess}_S((S^\pi, R^*), G) = b] = 1/2 = \Pr[\mathsf{guess}_S((S^\pi, R^\pi), G) = b]$
- $\Pr[\mathsf{fin}(S^\pi, R^*) = 0 \vee \mathsf{output}(S^\pi, R^*) = x_c]$
  $= \Pr[\mathsf{fin}(S^\pi, R^*) = 0] = 1 = \Pr[\mathsf{output}(S^\pi, R^\pi) = x_c]$
  $= \Pr[\mathsf{fin}(S^\pi, R^\pi) = 0 \vee \mathsf{output}(S^\pi, R^\pi) = x_c]$
- $\Pr[\mathsf{guess}_R((S^\pi, R^*), G_R^*) = c] - \epsilon(n) = 1/2 = \Pr[\mathsf{guess}_R((S^\pi, R^\pi), G) = c] + negl(n)$

Therefore, we have

$$u_R((S^\pi, G), (R^*, G_R^*)) > u_R((S^\pi, G), (R^\pi, G)) + \gamma_R\, \epsilon(n),$$

which means that the pair $((S^\pi, G), (R^\pi, G))$ is not in Nash equilibrium.

As we showed that the pair $((S^\pi, G_S), (R^\pi, G_R))$ is not in Nash equilibrium in every case, π is not game-theoretically secure.

□

References

[1] Ittai Abraham, Danny Dolev, Rica Gonen, and Joseph Y. Halpern. Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In Eric Ruppert and Dahlia Malkhi, editors, *PODC*, pages 53–62. ACM, 2006.

[2] Gilad Asharov, Ran Canetti, and Carmit Hazay. Towards a game theoretic view of secure computation. In Kenneth G. Paterson, editor, *EUROCRYPT*, volume 6632 of *Lecture Notes in Computer Science*, pages 426–445. Springer, 2011.

[3] Gilad Asharov and Yehuda Lindell. Utility dependence in correct and fair rational secret sharing. *J. Cryptology*, 24(1):157–202, 2011.

[4] Yan Zong Ding, Danny Harnik, Alon Rosen, Ronen Shaltiel, and Ronen Shaltiel. Constant-round oblivious transfer in the bounded storage model. In *J. Cryptology*, pages 165–202, 2007.

[5] Yevgeniy Dodis and Tal Rabin. Cryptography and game theory. In Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani, editors, *Algorithmic game theory*, chapter 8, pages 181–205. Cambridge University Press, 2007.

[6] Georg Fuchsbauer, Jonathan Katz, and David Naccache. Efficient rational secret sharing in standard communication networks. In Daniele Micciancio, editor, *TCC*, volume 5978 of *Lecture Notes in Computer Science*, pages 419–436. Springer, 2010.

[7] S. Dov Gordon and Jonathan Katz. Rational secret sharing, revisited. In Roberto De Prisco and Moti Yung, editors, *SCN*, volume 4116 of *Lecture Notes in Computer Science*, pages 229–241. Springer, 2006.

[8] Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. *J. Cryptology*, 25(1):158–193, 2012.

[9] Joseph Y. Halpern, Rafael Pass, and Rafael Pass. Game theory with costly computation: Formulation and application to protocol security. In *ICS*, pages 120–142, 2010.

[10] Joseph Y. Halpern and Vanessa Teague. Rational secret sharing and multiparty computation: extended abstract. In László Babai, editor, *STOC*, pages 623–632. ACM, 2004.

[11] Carmit Hazay and Yehuda Lindell. *Efficient Secure Two-Party Protocols: Techniques and Constructions*. Information Security and Cryptography Series. Springer-Verlag, 2010.

[12] Gillat Kol and Moni Naor. Cryptography and game theory: Designing protocols for exchanging information. In Ran Canetti, editor, *TCC*, volume 4948 of *Lecture Notes in Computer Science*, pages 320–339. Springer, 2008.

[13] Gillat Kol and Moni Naor. Games for exchanging information. In Cynthia Dwork, editor, *STOC*, pages 423–432. ACM, 2008.

[14] Anna Lysyanskaya and Nikos Triandopoulos. Rationality and adversarial behavior in multiparty computation. In Cynthia Dwork, editor, *CRYPTO*, volume 4117 of *Lecture Notes in Computer Science*, pages 180–197. Springer, 2006.

[15] Silvio Micali and Abhi Shelat. Purely rational secret sharing (extended abstract). In Reingold [17], pages 54–71.

[16] Shien Jin Ong, David C. Parkes, Alon Rosen, and Salil P. Vadhan. Fairness with an honest minority and a rational majority. In Reingold [17], pages 36–53.

[17] Omer Reingold, editor. *Theory of Cryptography, 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009. Proceedings*, volume 5444 of *Lecture Notes in Computer Science*. Springer, 2009.