Equivocable Oblivious Transfer

Donald Beaver*

Transarc Corporation

Abstract. We analyze and enhance Oblivious Transfer (OT) protocols to accommodate security against adaptive attacks. Previous analysis has been static in nature, treating the security of Alice and the security of Bob as separate cases, determined in advance. It remains unclear whether existing protocols are provably secure against adaptive attacks, but we provide enhancements to make them provably secure against attacks by *adaptive* 1-*adversaries*, who can choose *at any time* whether to corrupt Alice or Bob. We determine circumstances under which OT can be executed "in the open," without encrypting the messages, thereby giving simple alternatives to encrypting an entire interaction. We isolate *equivocation* properties that provide enough flexibility for a simulator to handle adaptive attacks. These properties also provide a means for classifying OT protocols and understanding the subtle demands of security against adaptive adversaries, as well as designing protocols that can be *proven* secure against adaptive attacks.

1 Introduction

Oblivious Transfer is one of the most broadly-applicable tools available for building secure interactive protocols. Its applicability is matched only by its simplicity: Alice must send a bit b to Bob without knowing whether it arrives, but with the assurance that it arrives with precisely 50-50 probability [Rab81]. A variety of implementations with varying efficiency and security have been offered, along with (occasional) proofs of security when Alice is bad or Bob is bad [Rab81, EGL82, BCR86a, BCR86b, BM89, KMO89, HL90, Boe91, B92]. Yet security in the most natural adversarial situation – namely in which neither Alice nor Bob need be bad at the outset, yet one *or both* can be corrupted *at any time* – does not follow from existing two-case analyses.

It seems intuitively clear, however, that an eavesdropper Eve overhearing the conversation between A and B should gain no knowledge. If Eve learned anything about whether b arrived, then so could Alice. If Eve learned anything about b, then so could Bob and thus Alice could infer that Bob did gain information about b.

The "flaw" in this reasoning is that it presupposes that Alice is faulty (perhaps only overly curious) or that Bob is faulty, but does not address other goals that Eve may have. For example, both A and B may be honest, and Eve may

* Transarc Corp., 707 Grant St., Pittsburgh, PA 15219; (412) 338-4365; beaver@transarc.com; http://www.transarc.com/~beaver.

U. Maurer (Ed.): Advances in Cryptology - EUROCRYPT '96, LNCS 1070, pp. 119-130, 1996.

be using the overheard conversation to decide whom to corrupt. Eve might wish to corrupt Bob only if he has a good chance of receiving b in that particular conversation. If she does corrupt Bob, Eve is indeed (unlike Alice) entitled to learn whether he received b and to learn b if so.

Of course, it may be possible to address each such motive with a plausible argument against insecurity, but we cannot be sure of listing all of Eve's ulterior motives. The situtation calls for a proper simulation result.

Simulation Results. Building a simulator $\mathcal{S}(A)$ for faulty Alice, and a second $\mathcal{S}(B)$ for faulty Bob, is a straightforward if tedious task for most secure OT protocols (sometimes with suitable modifications, such as the inclusion of zero-knowledge proofs of knowledge).

But attacks by adaptive adversaries are another thing. Based on the information it receives from an *ideal* OT channel, a simulator $\mathcal{S}$ should be able to provide Eve with an appropriate view of the corresponding messages between Alice and Bob, even before any corruptions are made. Should Eve decide to corrupt Alice, $\mathcal{S}$ is entitled to learn b (and if not too late, to decide what value to send), but $\mathcal{S}$ must then augment Eve's current view with a facsimile of Alice's internal history (random tapes, keys, *etc.*). This newly-gained history should look realistic to Eve, given the view she has already seen. The same should hold with respect to Bob, should Eve decide to corrupt him.

Furthermore, the case in which Eve ultimately corrupts *both* Alice and Bob should not be treated as moot. If the OT is part of a larger network protocol, it is certainly plausible that two arbitrary parties may be attacked. A proof of security for the larger protocol should not collapse when a module is "overrun." Only when attacks are *static* – namely when Eve decides *in advance* whom to corrupt – is such analysis indeed moot.

When Alice is corrupt, and Eve bases a decision whether to corrupt Bob based on subsequent interactions, the static analysis is insufficient. In particular, we must demand that $\mathcal{S}$ be able to augment Eve's view of Alice's history with a newly-created view of Bob's past history, should Eve decide to corrupt Bob also. A similar task is required when Bob is initially corrupt and Alice is later compromised.

Insufficiency of Current Work. In stark contrast to what intuition suggests, many if not all "secure" implementations of OT fail these requirements. In many cases, merely overhearing the conversation itself is enough to fix a particular value of b, even though it may be unknown. In other words, $\mathcal{S}$ cannot patch a current simulated view to be consistent with $\overline{b}$ having been sent. Brassard, Chaum and Crépeau's *chameleon blobs* [BCC88] are an appropriate but insufficient idea.[2]

[2] The chameleon property is generally sufficient against *static* adveraries, which is the scenario implicitly considered in [BCC88]. Also, they address commitment, which has weaker objectives than OT.

In other cases, the arrival of b is absolutely fixed by the conversation, so that even though nothing can be inferred (computationally), $\mathcal{S}$ cannot later pretend that b arrived when it didn't (or vice versa).

As a consequence, $\mathcal{S}$ may be able to construct reasonable facsimiles of views, but when a new party is corrupted, $\mathcal{S}$ has no freedom to pretend either a 0 or 1 was sent, or to pretend b arrived or didn't. Such inflexibility precludes a proper simulator-based proof of security against *adaptive* attacks.

Equivocation. Essentially, a simulator must be able to create equivocable views. We say an implementation of OT is **content-equivocable** if views can be generated (whether or not B is already corrupt) so that if Alice is suddenly corrupted, the views can be made consistent with Alice having sent either 0 or 1. We call it **result-equivocable** if a simulator can expand a view consistently to include either "received" or "failed to receive" when Bob is suddenly corrupted.

We discuss several OT implementations and show that these properties vary among them. It should be noted that the "chameleon" property of [BCC88] is distinct: the chameleon property requires that *Bob* be able to open a commitment to 0 or 1, but strictly speaking, an *eavesdropper* may be unable to do so without Bob's private knowledge. Moreover, Bob (*a.k.a.* the simulator augmenting Bob's view) may not be able to open bits of 0 or to 1 if the rest of Alice's information is newly compromised.

Weaker forms of equivocation are also helpful under certain circumstances. We say an implementation of OT is **weakly content-equivocable** if an already-corrupt Bob can make it consistent with either 0 or 1 whenever he has failed to receive b, even if Alice is also subsequently corrupted.[3] Likewise, we say the protocol is **weakly result-equivocable** if an already-corrupt Alice can fill in the view to make it consistent with reception or failure, even if Bob is also subsequently compromised.

Enclosure. If openly transmitting the messages of an OT protocol raises problems, the simple answer *seems* to be to enclose the conversation with an appropriate encryption scheme. An adversary would have to corrupt either Alice or Bob just to gain any meaningful knowledge about the conversation. While this does not quite reduce the adaptive analysis to the static case, it will work in certain cases, namely if the enclosed OT protocol is *weakly* equivocable (both by content and by result). It does not work for many commonly-known OT protocols.

Moreover, using an encryption scheme presents something of a chicken-and-egg dilemma. The encryption scheme itself must be secure against adaptive attacks. Such encryptions can be found in the work of Beaver and Haber [BH92], in the ingenious but impractical very recent scheme by Canetti, Feige, Goldreich and Naor [CFGN96], and most recently in an efficient scheme by Beaver [B96]. Unless erasing is permitted, the weakest assumptions (one-way trapdoor permutations) require third parties to assist in the encryption. Stronger assumptions,

[3] This is nearly the same as the "chameleon" property. It differs in requiring consistency *even* if Alice is then corrupted.

such as RSA or Diffie-Hellman [RSA78, DH76], require no third-party assistance [CFGN96, B96].

Efficiency. The expense of involving third parties for every message bit and the danger of attempting to erase bits completely lead us to seek to minimize the use of such techniques. We show that it is not necessary to protect the entire conversation but that encrypting one or two extra bits suffices. We also show that encrypting one bit securely against *adaptive* attacks is *necessary*.

Contributions. In summary, we provide the following observations and technologies regarding OT:

- Static analysis of security does not show security against adaptive attacks.
- Static security plus encryption is not sufficient to protect OT against adaptive attacks.
- Certain equivocability properties plus encryption do suffice to provide provable security against fully adaptive attacks.
- Equivocability varies among proposed OT schemes (including a new scheme discussed within), but most can be enhanced for security against adaptive 1-adversaries;
- Encrypting two extra bits is a sufficient alternative to encrypting the conversation; encrypting one bit is necessary.

2 Background

Attacks: Static or Adaptive. An adversary is a probabilistic poly-time TM (PPTM) that issues two sorts of messages: "*corrupt* i," "*send* m from i to j." It receives two sorts of responses: "*view of* i," "*receive* m from j to i." Whether its send/receive message is honored depends on whether it has issued a request to corrupt i.

A **static t-adversary** is an adversary who issues up to t *corrupt* requests before the protocol starts. An **adaptive t-adversary** may issue up to t such requests at any time.

OT specification. The **specification protocol for OT** is a three-party protocol consisting of $\hat{A}$, $\hat{B}$, and incorruptible party OT. $\hat{A}$ has input b, which it is instructed to send to OT. OT flips a coin, $?b$, and sends $(?b, ?b \wedge b)$ to $\hat{B}$.[4] The communication channels between $\hat{A}$ and OT and between OT and $\hat{B}$ are absolutely private.

Simulation-based security. The definition of simulator-based static security is the standard approach: find an appropriate simulator for the case in which Alice is bad, and another simulator for when Bob is bad. We focus on the adaptive case.

[4] Thus, $(0,0)$ means "failed," while $(1,b)$ means "received b."

In the adaptive case, there is a single simulator, $\mathcal{S}$, who receives requests from and delivers responses to the attacker, **Adv**, creating an environment for **Adv** as though **Adv** were attacking a given implementation. $\mathcal{S}$ is itself an attacker acting within the specification protocol for OT, which is run with $\hat{A}$ on input b. When **Adv** corrupts player i, $\mathcal{S}$ issues a corruption request and is given $\hat{i}$'s information.[5] $\mathcal{S}$ responds to **Adv** with a facsimile of the "*view of* i" response that **Adv** expects. $\mathcal{S}$ receives all of **Adv**'s "*send* m" requests and provides **Adv** with facsimiles of "*receive* m" responses. Finally, **Adv** (or $\mathcal{S}$ on **Adv**'s behalf) writes its output, $y_{\mathbf{Adv}}$.

Let **Adv**, with auxiliary input $x_{\mathbf{Adv}}$, attack a given OT implementation **OT** in which Alice holds input b. The execution induces a distribution $(A(b), B, \mathbf{Adv}(x_{\mathbf{Adv}}))$ on output triples, $(y_A, y_B, y_{\mathbf{Adv}})$.

Let $\mathcal{S}(\mathbf{Adv}(x_{\mathbf{Adv}}))$ attack the OT specification. The execution induces a distribution $(\hat{A}(b), \hat{B}, \mathcal{S}(\mathbf{Adv}(x_{\mathbf{Adv}})))$ on output triples, $(y_{\hat{A}}, y_{\hat{B}}, y_{\mathcal{S}})$.

An extra, "security parameter" k may be included. This provides a sequence of distributions on output triples in each scenario. Let $\approx$ denote *computational indistinguishability*, a notion whose formal definition is omitted for reasons of space (*cf.* [GMR89]).

The implementation $\mathcal{OT}$ is **secure against adaptive t-adversaries** if, for any adaptive t-adversary **Adv**, there is a PPTM simulator $\mathcal{S}$ such that for any b, $(A(b), B, \mathbf{Adv}(x_{\mathbf{Adv}})) \approx (\hat{A}(b), \hat{B}, \mathcal{S}(\mathbf{Adv}(x_{\mathbf{Adv}})))$. In other words, the simulator maps attacks on the implementation to equivalent attacks on the specification.

Encryption. The **specification protocol for secure channels** is a two-party protocol consisting of $\hat{S}$, $\hat{R}$, in which $\hat{S}$ produces a bit m which is transferred securely to $\hat{R}$. An eavesdropper knows only that a bit was sent, or that one or the other party decided to abort. An **encryption scheme secure against adaptive t-adversaries** is a (two-party) protocol such that, for any adaptive t-adversary **Adv**, there is a PPTM simulator $\mathcal{S}$ such that for any m, $(S(m), R, \mathbf{Adv}(x_{\mathbf{Adv}})) \approx (\hat{S}(m), \hat{R}, \mathcal{S}(\mathbf{Adv}(x_{\mathbf{Adv}})))$.[6] We generally assume that the implementation network provides authenticated, service-undeniable, point-to-point connections. The traffic over the lines is public, however.

OT Variants. We also consider two variants on OT: one-out-of-two OT ($\frac{1}{2}$OT), in which Alice holds (b_0, b_1) and Bob receives (c, b_c) for a random c unknown to Alice [EGL82]; and chosen one-out-of-two OT ($\binom{2}{1}$OT), in which Alice holds (b_0, b_1) and Bob receives b_c for a c of his choice, but unknown to Alice.

3 Equivocation

As described above, content-equivocation involves the simulator adjusting or augmenting a view to be consistent with sending either $b = 0$ or $b = 1$. This

[5] $\hat{i}$ is a player in the specification protocol and is unaware of messages being passed in a given implementation. In particular, $\hat{A}$ knows only its input b (and its message to OT), and $\hat{B}$ knows only its message from OT.

[6] A multiparty implementation is also possible; the formalities are similar.

may be necessary when $\mathcal{S}$ did not corrupt $\hat{A}$ before it started to create views for **Adv**, and thus did not know the bit b sent by $\hat{A}$ to OT in the specification protocol. If or when **Adv** corrupts Alice, $\mathcal{S}$ does obtain b (perhaps already sent to OT, perhaps not) and must bend its views so that the up-until-now honest Alice appears to have used b as its input.

Likewise, result-equivocation involves the simulator augmenting a view to be consistent with having received b or failed to receive b, depending on what $\hat{B}$ may actually have received in the specification protocol.

3.1 Various Implementations

We characterize several implementations according to whether they are content-equivocable (C.E.) or result-equivocable (R.E.). This list is not meant to be exhaustive.

Often, such implementations omit certain requirements such as Alice proving in zero-knowledge that she knows the effective bit b she is sending. Without such ZK proofs of knowledge (ZKPK's), some protocols can be broken (*cf.* [B92]). For the purposes of this paper, we assume that such ZKPK's are included, and postpone an analysis of the equivocability of the ZKPK's themselves.

Rabin. Rabin proposed an implementation based on the intractability of factoring [Rab81]. Let $E_{p,q}(x,r)$ be a probabilistic encryption of x with decryption key (p,q). For example, $E_{p,q}(b,r) = (-1)^b r^2 \pmod{n}$; then the quadratic residuosity of $E_{p,q}(b,r)$ can be determined when p and q are known. (This requires the stronger QRA assumption: determining Quadratic Residuosity is intractable [GM84].)

1. $A \rightarrow B$: $n = pq$ and $E_{p,q}(b,r)$, for a random $r \pmod{n}$.
2. $B \rightarrow A$: $z = x^2 \pmod{n}$, for a random $x \pmod{n}$.
3. $A \rightarrow B$: $w = \sqrt{z} \pmod{n}$, chosen at random from the four square roots $\{\pm x, \pm y\}$ of z.
4. B: if $w = \pm x$, output $(0,0)$; else factor n, decrypt b, and output $(1,b)$.

This protocol is neither C.E. nor weakly C.E., because $E_{p,q}(b,r)$ fixes the value of b. It is, however, R.E., since simulator $\mathcal{S}$ can pretend that Bob internally generated $\pm x$ whenever $\hat{B}$ obtained $?b = 0$ ("failed") in the specification, or pretend that Bob held $\pm y$ whenever $\hat{B}$ obtained $?b = 1$ ("received") in the specification.

Den Boer. In Den Boer's method [Boe91], based on the QRA, Bob generates $n = pq$. Alice sends a couple of residues, which with probability $1/2$ have the same quadratic residuosity (also equal to b), and with probability $1/2$ have different quadratic residuosity (hence Bob learns nothing). Let $Q_n(x) = 0$ if x is a quadratic residue mod n, else let $Q_n(x) = 1$.

1. $B \rightarrow A$: $n = pq$ and a for a random $a \pmod{n}$.

2a. A: choose random bits c, d, and random $r \pmod{n}$. Set $x_0 = (-1)^b r^2 \pmod{n}$, $x_1 = (-1)^c a x_0^{-1} \pmod{n}$.

2b. $A \rightarrow B$: $(x_d, x_{\overline{d}})$.[7]

3. B: receive (y_0, y_1). If $Q_n(y_0) = Q_n(y_1)$, then output $(1, Q_n(y_0))$ ["received $Q_n(y_0)$"]. Else output $(0, 0)$.

This protocol is neither R.E. nor weakly R.E., because the quadratic residuosities of (y_0, y_1) are fixed, once seen. Thus, if they are $(0, 0)$ or $(1, 1)$, then $\mathcal{S}$ cannot pretend that Bob failed to receive the bit. If one witnesses the conversation between honest players, then it will be impossible to change the bit sent when it is forced to arrive, so the protocol is not C.E.. On the other hand, the protocol is weakly C.E., because, having seen that Bob failed to receive the bit, $\mathcal{S}$ can swap the roles of x_0 and x_1, thereby choosing between Alice's having sent $Q_n(x_0)$ or $Q_n(x_1)$ (which is $1 \oplus Q_n(x_0)$). This choice is perfectly acceptable even when Alice is subsequently corrupted.

Novel. In this method, bearing similarity to Den Boer's OT protocol and the commitment schemes of Brassard, Chaum and Crépeau [BCC88], Alice sends b over a channel that, with 50-50 probability, delivers nothing but quadratic residues.

1. $B \rightarrow A$: $n = pq$ and a for a random $a \pmod n$.[8]

2a. A: choose random bit c, and random $r \pmod n$. Set $s = (-1)^c a \pmod n$ and $x = s^b r^2 \pmod n$.

2b. $A \rightarrow B$: (x, c).[9]

3. B: receive (x, c). If $c \neq Q_n(a)$, then output $(1, Q_n(x))$. Else output $(0, 0)$.

Note that when s is a quadratic residue ($c = Q_n(a)$), no information is transmitted. Otherwise, the residuosity of $x = s^b$ determines b. This protocol is neither R.E. nor weakly R.E., because the quadratic residuosity of a and the value of c, once seen, determine whether the bit arrives. If one witnesses the conversation between honest players, then it will be impossible to change the bit sent in those cases that it is forced to arrive, so the protocol is not C.E.. On the other hand, the protocol is weakly C.E., as can be seen by applying the arguments of [BCC88] for their commitment scheme. If $\mathcal{S}$ wishes to reveal a fake b it initially used to construct a view, it can use the original fake r. If $\mathcal{S}$ wishes to patch $b = 0$ to pretend that $b = 1$ was sent, it replaces r with $r/\sqrt{s}$, so that the already-seen value of x, namely $s^0 r^2 \equiv s^1 (r/\sqrt{s})^2$, remains unchanged. Similarly, to patch $b = 1$ to $b = 0$, $\mathcal{S}$ replaces r with $r\sqrt{s}$. These arguments make it evident that the weak-C.E. property is comparable to the chameleon property of [BCC88].

Even-Goldreich-Lempel. Two different implementations of $\frac{1}{2}$OT were given by [EGL82]. In the earlier version, a protocol that assumes Bob does not cheat was given. Because the later version contains an apparently more robust but subtly

[7] Alice must also give a ZKPK that she knows the effective b she is sending, else the protocol is breakable [B92]. We have also slightly modified the protocol to send x_i's in random order.

[8] Along with ZKPK of p, q.

[9] Along with ZKPK of effective b.

breakable protocol, we focus on the simpler approach. The simpler approach permits Bob to gain both bits (b_0, b_1) if he departs from the protocol. Canetti, Feige, Goldreich and Naor, however, have applied this otherwise undesirably vulnerable method to provide an remarkable solution to adaptively-secure encryption [CFGN96]. Thus, the properties of the EGL protocol are of interest.

Let $f(x)$ be a trapdoor one-way permutation, and let Alice hold the trapdoor. Let $B_f(x)$ give a hard bit of x with respect to f. Recall that Alice has input bits b_0, b_1. Bob will receive either b_0 or b_1 along one of two channels. One channel is masked by a bit that Bob knows, the other by a hard bit.

1a. B: choose random bit c and random strings x_0, x_1. If $c = 0$, set $(y_0, y_1) = (f(x_0), x_1)$, set $(y_0, y_1) = (x_0, f(x_1))$.
1b. $B \rightarrow A$: (y_0, y_1).
2a. A: choose random bit d. Set $z_0 = B(f^{-1}(y_d)) \oplus b_0$, $z_1 = B(f^{-1}(y_{\overline{d}})) \oplus b_1$.
2b. $A \rightarrow B$: (z_0, z_1, d).
3. B: receive (z_0, z_1, d). Set $e = c \oplus d$, $\beta = z_e \oplus B(x_c)$. Output (e, β) = "received bit e with value β."

Clearly, malicious Bob can send $(f(x_0), f(x_1))$ and always receive both bits without detection. Let us consider whether views can be generated even when Alice and Bob are honest but curious. $\mathcal{S}$ cheats by indeed using $(f(X_0), f(X_1))$ for random strings X_0,X_1. This permits $\mathcal{S}$ to use $(x_0 = X_0, x_1 = f(X_1))$ or $(x_0 = f(X_0), x_1 = X_1)$ when it reveals Bob's internal history. Such a simulation is *perfectly* indistinguishable from actual histories. This flexibility allows $\mathcal{S}$ to reverse c at the last minute. Since c determines which bit arrives, the protocol is R.E..

On the other hand, (y_0, y_1) determines the hard bits used as masks in step 3, and this cannot be reversed once Bob or an outsider has seen those messages. Thus the protocol is not weakly C.E..

Bellare-Micali. Unlike the previous protocols, this implementation of $\binom{2}{1}$OT seeks to use the intractability of discrete logarithm for protection [BM89]. Although it has subtle correlation problems if used to send more than one message, its use of an alternate intractability assumption makes it of interest. Let p be a prime, g be a primitive element mod p, and let C be a public value mod p whose discrete logarithm is unknown.

1a. A: choose random bit i and random $x \pmod{p}$. If $c = 0$, set $(\beta_0, \beta_1) = (g^x, Cg^{-x})$, else set $(\beta_0, \beta_1) = (Cg^{-x}, g^x)$.
1b. $A \rightarrow B$: (β_0, β_1).[10]
2a. A: choose random $y_0, y_1 \pmod{p-1}$. Set $\alpha_0 = g^{y_0}$, $\alpha_1 = g^{y_1}$. Set $\gamma_0 = \beta_0^{y_0}$, $\gamma_1 = \beta_1^{y_1}$. Set $r_0 = b_0 \oplus \gamma_0$, $r_1 = b_1 \oplus \gamma_1$,
2b. $A \rightarrow B$: $(\alpha_0, \alpha_1, r_0, r_1)$.
3. B: receive $(\alpha_0, \alpha_1, r_0, r_1)$. Set $\gamma_i = \alpha_i^x$. Set $b_i = \gamma_i \oplus r_i$.

[10] Verified by comparing to public C generated by third-party.

With simple modifications, this is easily converted to $\frac{1}{2}$OT or OT. The protocol is neither R.E. nor C.E.. Notice that witnessing (β_0, β_1) fixes x. Simple algebra shows that it is impossible to find an x and $\hat{x}$ such that $(\beta_0, \beta_1) = (g^x, Cg^{-x})$ and $(\beta_0, \beta_1) = (Cg^{-\hat{x}}, g^{\hat{x}})$, unless $C = 1$ (which defeats the protocol). Thus, once an eavesdropper or Bob sees (β_0, β_1), the values of x and c are determined. The α_0, α_1 values determine y_0, y_1 and therefore also determine the masks γ_0, γ_1; thus the simulator cannot get away with changing the bits b_0, b_1.

Summary. Altogether, we observe the following variability among several proposed implementations of OT:

Protocol	Result Eq.(str/wk)	Content Eq. (str/wk)	Comment
Rabin	+/+	-/-	factoring
Den Boer	-/-	-/+	QRA
Novel	-/-	-/+	QRA
EGL-1	+/+	-/-	weak adversary model
BM	-/-	-/-	discrete log

Although the Rabin and EGL implementations are quite different in robustness, they share the same equivocation characteristics.

3.2 General Reductions

We note that reductions among variants of OT can preserve equivocability:

Theorem 1. *The following are equivalent: (1) there exists a result-equivocable implementation of* OT*; (2) there exists a result-equivocable implementation of* $\frac{1}{2}$OT*; (3) there exists a result-equivocable implementation of* $\binom{2}{1}$*OT.*

Proof. Crépeau's reductions suffice [C87]. Consider (1) $\Rightarrow$ (2). Say that Alice transmits $15k$ random bits $\{r_1, \ldots, r_{15k}\}$, and Bob receives those with indices in $R \subseteq \{1, \ldots, 15k\}$. With probability $\geq 1 - 2^{-k}$, Bob can choose a random subset $U_0 \subseteq R$ of size $5k$; Bob also chooses $U_1 \subseteq \overline{R}$. Then Bob knows $\beta_0 = \oplus_{i \in U_0} r_i$ but with probability $\geq 1 - 2^{-k}$ he cannot determine $\beta_1 = \oplus_{i \in U_1} r_i$. By sending (U_0, U_1) to Alice in random order, a half-obscured pair of channels is created; Alice sends (b_0, b_1) along them in random order. For our purposes, it suffices to note that a simulator can reverse the roles of U_0 and U_1 at will, since the underlying OT is result-equivocable. Thus the identity of the arriving bit in the higher-level $\frac{1}{2}$OT protocol can be reversed. Showing (1) $\Rightarrow$ (3) is similar, and the reverse directions are trivial. □

Theorem 2. *The following are equivalent: (1) there exists a content-equivocable implementation of* OT*; (2) there exists a content-equivocable implementation of* $\frac{1}{2}$OT*; (3) there exists a content-equivocable implementation of* $\binom{2}{1}$*OT.*

Proof. Similar to 1. □

4 Enclosure

Given an OT implementation OT and an encryption scheme $\mathcal{E}$, let ENCLOSE(OT,$\mathcal{E}$) be the protocol that requires each message in OT to be encrypted with $\mathcal{E}$.

Theorem 3. *Let $\mathcal{E}$ be an encryption scheme secure against adaptive 2-adversaries. If* OT *is secure against static 1-adversaries, then* ENCLOSE(OT,$\mathcal{E}$) **is secure** *against adaptive 1-adversaries, but it is* **not necessarily secure** *against adaptive 2-adversaries.*

Proof. Let **Adv** be an adaptive 1-adversary. If **Adv** corrupts Alice, then the encrypted messages can be opened to any values, hence $\mathcal{S}$ runs a "static" simulator for Alice, extracts a view up to the moment of corruption, and equivocates the ciphertexts seen by **Adv** to be consistent with encrypting that view. **Adv** has used up its corruptions, hence we needn't worry about ever adjusting the view. A similar analysis holds for Bob.

For the adaptive 2-adversary case, note that even if the messages are encrypted, **Adv** can corrupt Bob to see them in the clear. Rabin's OT protocol does not permit b to be changed after corrupt Bob has seen the conversation and failed to receive b. Upon corrupting Alice, the simulator has at best a 50-50 chance of filling in the view to be consistent with the value b that $\mathcal{S}$ learns by corrupting $\hat{A}$ in the specification protocol. □

With enclosure, the weak equivocation properties suffice, however. (Notice that Rabin's OT failed above because it is not content-equivocable.)

Theorem 4. *Let $\mathcal{E}$ be an encryption scheme secure against adaptive 2-adversaries. If* OT *is secure against static 1-adversaries and both weakly content- and weakly result- equivocable, then* ENCLOSE(OT,$\mathcal{E}$) *is secure against adaptive 2-adversaries.*

Proof. Let **Adv** be an adaptive 2-adversary. As before, when **Adv** makes its first request to corrupt Alice or Bob, $\mathcal{S}$ corrupts $\hat{A}$ or $\hat{B}$ in the specification protocol and calls on the static simulator to generate an appropriate view, then equivocates the ciphertexts **Adv** has seen, to make them consistent with the view.

Now, say Alice is corrupted first. If **Adv** requests to corrupt Bob, $\mathcal{S}$ corrupts $\hat{B}$, determines the bit reception pattern (if the specification protocol is that far along), and based on the result-equivocability property, extends the view to appear as though Bob received the same pattern. If Bob is corrupted first, then upon later corruption of Alice, $\mathcal{S}$ discovers b by corrupting $\hat{A}$ and uses content-equivocability to extend the view to appear as though Alice had been using b all along. □

5 Efficient Enclosure

It turns out that many OT protocols can be made provably robust against 1-adaptive attacks without taking the drastic measure of encrypting the entire conversation.

Theorem 5. *Let $\mathcal{E}$ be an encryption scheme secure against adaptive 1-adversaries. The Rabin, Den Boer, and Novel protocols can be made secure against adaptive 1-adversaries by encrypting at most 2 additional bits while running the rest of the protocol in the open.*

Proof. In each protocol, Alice applies a random bit r instead of b, and she also sends $\mathcal{E}(r \oplus b)$. The Rabin protocol needs no further modification.

The Den Boer protocol is further modified as follows. In step 2b, Alice sends x_d instead of $(x_d, x_{\overline{d}})$, and Bob receives it as y_0. Alice sends $\mathcal{E}(c)$, Bob calculates $y_1 = x_{\overline{d}} = (-1)^c a y_0^{-1}$, and the protocol continues as written. The Novel protocol is further modified as follows. In step 2b, Alice sends $(x, \mathcal{E}(c))$ instead of (x, c). Simulation details are omitted for space. □

Necessity of Encrypting $\Omega(1)$ Bits. Let OT be an implementation of OT secure against adaptive 1-adversaries. By treating OT as a noisy channel, Alice can transmit N bits to Bob using $O(N)$ invocations of OT, in the limit. Because OT is secure, an adversary must corrupt either Alice or Bob to gain any knowledge about the bit stream. Because OT is secure against adaptive 1-adversaries, the bit stream can be equivocated when Alice or Bob is corrupted, to match any desired sequence.

Thus the existence of an OT protocol secure against adaptive 1-adversaries implies the existence of a communication scheme secure against adaptive 1-adversaries.

6 Conclusions

Our results advance toward the 2-adversary case by improving security from static to adaptive 1-adversaries and providing the analytical support to develop stronger methods.

Two intriguing open problems are raised. First, is it possible to combine two protocols with different properties to give a protocol with the combination of those properties? For example, is it possible to combine a weakly R.E. protocol (such as Rabin) with a weakly C.E. protocol (such as Den Boer or Novel) to obtain a hybrid protocol that is weakly R.E. and weakly C.E.?

Second, does there exist an OT protocol that is secure against adaptive 2-adversaries, but which does not require erasing [BH92] or the impractical overhead (numerous third-party assistance and/or $\Omega(k)$ repeated encryptions per bit sent) of [CFGN96]?

A positive answer to the first would enable a positive answer to the second, using results [B96] obtained after the submission of this article. The proofs behind theorems 4 and 5 generalize to show that, if an implementation $\binom{2}{1}$OT of $\binom{2}{1}$OT is secure against static 1-adversaries and both weakly content- and weakly result- equivocable, then $\binom{2}{1}$OT can be made secure against adaptive 2-adversaries by encrypting 3 bits while running the rest of the protocol in the open. The simple and efficient two-party, adaptively-secure encryption scheme of [B96] provides the means to encrypt the additional 3 bits at minimal cost.

References

[B92] D. Beaver. "How to Break a 'Secure' Oblivious Transfer Protocol." *Advances in Cryptology – Eurocrypt '92 Proceedings*, Springer–Verlag LNCS 658, 1993, 285–296.

[B96] D. Beaver. "Adaptively Secure Encryption." Penn State Univ. Tech Report PSU-CSE-96-031, February 7, 1996.

[BH92] D. Beaver, S. Haber. "Cryptographic Protocols Provably Secure Against Dynamic Adversaries." *Advances in Cryptology – Eurocrypt '92 Proceedings*, Springer–Verlag LNCS 658, 1993, 307–323.

[BM89] M. Bellare, S. Micali. "Non-Interactive Oblivious Transfer and Applications." *Advances in Cryptology – Crypto '89 Proceedings*, Springer–Verlag LNCS 435, 1990, 547–557.

[BCR86a] G. Brassard, C. Crépeau, J. Robert. "All or Nothing Disclosure of Secrets." *Advances in Cryptology – Crypto '86 Proceedings*, Springer–Verlag LNCS 263, 1987, 234–238.

[BCR86b] G. Brassard, C. Crépeau, J. Robert. "Information Theoretic Reductions among Disclosure Problems." *Proceedings of the* 27^{th} *FOCS*, IEEE, 1986, 168–173.

[BCC88] G. Brassard, D. Chaum, C. Crépeau. "Minimum Disclosure Proofs of Knowledge." *J. Comput. Systems Sci.* **37**, 1988, 156–189.

[CFGN96] R. Canetti, U. Feige, O. Goldreich, M. Naor. "Adaptively Secure Multiparty Computation." To appear, *Proceedings of the* 28^{th} *STOC*, ACM, 1996.

[C87] C. Crépeau. "Equivalence Between Two Flavours of Oblivious Transfers." *Advances in Cryptology – Crypto '87 Proceedings*, Springer–Verlag LNCS 293, 1988, 350–354.

[Boe91] B. den Boer. "Oblivious Transfer Protecting Secrecy." *Advances in Cryptology – Eurocrypt '91 Proceedings*, Springer–Verlag LNCS 547, 1991, 31–45.

[DH76] W. Diffie, M. Hellman. "New Directions in Cryptography." *IEEE Transactions on Information Theory* **IT-22**, November 1976, 644–654.

[EGL82] S. Even, O. Goldreich, A. Lempel. "A Randomized Protocol for Signing Contracts." *Comm. of the ACM* **28**:6, 1985, 637–647. (Early version: *Proceedings of Crypto 1982*, Springer–Verlag, 1983, 205–210.)

[GM84] S. Goldwasser, S. Micali. "Probabilistic Encryption." *J. Comput. Systems Sci.* **28**, 1984, 270–299.

[GMR89] S. Goldwasser, S. Micali, C. Rackoff. "The Knowledge Complexity of Interactive Proof Systems." *SIAM J. on Computing* **18**:1, 1989, 186–208.

[HL90] L. Harn, H. Lin. "Noninteractive Oblivious Transfer." *Electronics Letters* **26**:10, May 1990, 635–636.

[KMO89] J. Kilian, S. Micali, R. Ostrovsky. "Minimum Resource Zero-Knowledge Proofs." *Proceedings of the* 30^{th} *FOCS*, IEEE, 1989, 1989, 474–479.

[Rab81] M.O. Rabin. "How to Exchange Secrets by Oblivious Transfer." TR-81, Harvard, 1981.

[RSA78] R. Rivest, A. Shamir, L. Adleman. "A Method for Obtaining Digital Signatures and Public Key Cryptosystems." *Communications of the ACM* **21**:2, 1978, 120–126.