WIKIPEDIA

Oblivious transfer

In cryptography, an **oblivious transfer** (**OT**) protocol is a type of protocol in which a sender transfers one of potentially many pieces of information to a receiver, but remains oblivious as to what piece (if any) has been transferred.

The first form of oblivious transfer was introduced in 1981 by Michael O. Rabin.[1] In this form, the sender sends a message to the receiver with probability 1/2, while the sender remains oblivious as to whether or not the receiver received the message. Rabin's oblivious transfer scheme is based on the RSA cryptosystem. A more useful form of oblivious transfer called **1–2 oblivious transfer** or "1 out of 2 oblivious transfer", was developed later by Shimon Even, Oded Goldreich, and Abraham Lempel,[2] in order to build protocols for secure multiparty computation. It is generalized to "1 out of *n* oblivious transfer" where the user gets exactly one database element without the server getting to know which element was queried, and without the user knowing anything about the other elements that were not retrieved. The latter notion of oblivious transfer is a strengthening of private information retrieval, in which the database is not kept private.

Claude Crépeau showed that Rabin's oblivious transfer is equivalent to 1–2 oblivious transfer.[3]

Further work has revealed oblivious transfer to be a fundamental and important problem in cryptography. It is considered one of the critical problems in the field, because of the importance of the applications that can be built based on it. In particular, it is complete for secure multiparty computation: that is, given an implementation of oblivious transfer it is possible to securely evaluate any polynomial time computable function without any additional primitive.[4]

Contents

Rabin's oblivious transfer protocol

In Rabin's oblivious transfer protocol, the sender generates an RSA public modulus $N=pq$ where p and q are large prime numbers, and an exponent e relatively prime to $\lambda(N) = (p-1)(q-1)$. The sender encrypts the message m as $m^e \bmod N$.

1. The sender sends N, e, and $m^e \bmod N$ to the receiver.
2. The receiver picks a random x modulo N and sends $x^2 \bmod N$ to the sender. Note that $\gcd(x,N) = 1$ with overwhelming probability, which ensures that there are 4 square roots of $x^2 \bmod N$.
3. The sender finds a square root y of $x^2 \bmod N$ and sends y to the receiver.

If the receiver finds y is neither x nor $-x$ modulo N, the receiver will be able to factor N and therefore decrypt m^e to recover m (see Rabin encryption for more details). However, if y is x or $-x$ mod N, the receiver will have no information about m beyond the encryption of it. Since every quadratic residue modulo N has four square roots, the probability that the receiver learns m is 1/2.

1–2 oblivious transfer

In a 1–2 oblivious transfer protocol, the sender has two messages m_0 and m_1, and the receiver has a bit b, and the receiver wishes to receive m_b, without the sender learning b, while the sender wants to ensure that the receiver receives only one of the two messages. The protocol of Even, Goldreich, and Lempel (which the authors attribute partially to Silvio Micali), is general, but can be instantiated using RSA encryption as follows.

Alice				Bob		
Calculus	**Secret**	**Public**		**Public**	**Secret**	**Calculus**
Messages to be sent	m_0, m_1					
Generate RSA key pair and send public portion to Bob	d	N, e	$\Rightarrow$	N, e		Receive public key
Generate two random messages		x_0, x_1	$\Rightarrow$	x_0, x_1		Receive random messages
					k, b	Choose $b \in \{0, 1\}$ and generate random k. **Here:** $gcd(k, N) = 1$
		v	$\Leftarrow$	$v = (x_b + k^e) \mod N$		Compute the encryption of k, blind with x_b and send to Alice
One of these will equal k, but Alice does not know which.	$k_0 = (v - x_0)^d \mod N$ $k_1 = (v - x_1)^d \mod N$					
Send both messages to Bob		$m'_0 = m_0 + k_0$ $m'_1 = m_1 + k_1$	$\Rightarrow$	m'_0, m'_1		Receive both messages
					$m_b = m'_b - k$	Bob decrypts the m'_b since he knows which x_b he selected earlier.

1. Alice has two messages, m_0, m_1, and wants to send exactly one of them to Bob. Bob does not want Alice to know which one he receives.
2. Alice generates an RSA key pair, comprising the modulus N, the public exponent e and the private exponent d
3. She also generates two random values, x_0, x_1 and sends them to Bob along with her public modulus and exponent.
4. Bob picks b to be either 0 or 1, and selects either the first or second x_b.
5. He generates a random value k and blinds x_b by computing $v = (x_b + k^e) \mod N$, which he sends to Alice.
6. Alice doesn't know (and hopefully cannot determine) which of x_0 and x_1 Bob chose. She applies both of her random values and comes up with two possible values for k: $k_0 = (v - x_0)^d \mod N$ and $k_1 = (v - x_1)^d \mod N$. One of these will be equal to k and can be correctly decrypted by Bob (but not Alice), while the other will produce a meaningless random value that does not reveal any information about k.
7. She combines the two secret messages with each of the possible keys, $m'_0 = m_0 + k_0$ and $m'_1 = m_1 + k_1$, and sends them both to Bob.
8. Bob knows which of the two messages can be unblinded with k, so he is able to compute exactly one of the messages $m_b = m'_b - k$

1-out-of-*n* oblivious transfer and *k*-out-of-*n* oblivious transfer

A 1-out-of-*n* oblivious transfer protocol can be defined as a natural generalization of a 1-out-of-2 oblivious transfer protocol. Specifically, a sender has *n* messages, and the receiver has an index *i*, and the receiver wishes to receive the *i*-th among the sender's messages, without the sender learning *i*, while the sender wants to ensure that the receiver receive only one of the *n* messages.

1-out-of-*n* oblivious transfer is incomparable to private information retrieval (PIR). On the one hand, 1-out-of-*n* oblivious transfer imposes an additional privacy requirement for the database: namely, that the receiver learn at most one of the database entries. On the other hand, PIR requires communication sublinear in *n*, whereas 1-out-of-*n* oblivious transfer has no such requirement.

1-*n* oblivious transfer protocols were proposed, e.g., by Moni Naor and Benny Pinkas,[10] William Aiello, Yuval Ishai and Omer Reingold,[11] Sven Laur and Helger Lipmaa.[12]. In 2017, Kolesnikov et al.,[13] proposed an efficient 1-n oblivious transfer protocol which requires roughly 4x the cost of 1-2 oblivious transfer in amortized setting.

Brassard, Crépeau and Robert further generalized this notion to *k*-*n* oblivious transfer,[5] wherein the receiver obtains a set of *k* messages from the *n* message collection. The set of *k* messages may be received simultaneously ("non-adaptively"), or they may be requested consecutively, with each request based on previous messages received.[6]

Generalized oblivious transfer

k-*n* Oblivious transfer is a special case of generalized oblivious transfer, which was presented by Ishai and Kushilevitz.[7] In that setting, the sender has a set *U* of *n* messages, and the transfer constraints are specified by a collection *A* of permissible subsets of *U*. The receiver may obtain any subset of the messages in *U* that appears in the collection *A*. The sender should remain oblivious of the selection made by the receiver, while the receiver cannot learn the value of the messages outside the subset of messages that he chose to obtain. The collection *A* is monotone decreasing, in the sense that it is closed under containment (i.e., if a given subset *B* is in the collection *A*, so are all of the subsets of *B*). The solution proposed by Ishai and Kushilevitz uses the parallel invocations of 1-2 oblivious transfer while making use of a special model of private protocols. Later on, other solutions that are based on secret sharing were published – one by Bhavani Shankar, Kannan Srinathan, and C. Pandu Rangan,[8] and another by Tamir Tassa.[9]

Origins

In the early seventies Stephen Wiesner introduced a primitive called **multiplexing** in his seminal paper "Conjugate Coding", which was the starting point of quantum cryptography.[1] Unfortunately it took more than ten years to be published. Even though this primitive was equivalent to what was later called *1–2 oblivious transfer*, Wiesner did not see its application to cryptography.

Quantum oblivious transfer

Protocols for oblivious transfer can be implemented with quantum systems. In contrast to other tasks in quantum cryptography, like quantum key distribution, it has been shown that quantum oblivious transfer cannot be implemented with unconditional security, i.e. the security of quantum oblivious transfer protocols cannot be guaranteed only from the laws of quantum physics.[1]

See also

- k-anonymity
- Secure multi-party computation
- Zero-knowledge proof
- Private information retrieval

References

- **^0.** Stephen Wiesner, "Conjugate coding", Sigact News, vol. 15, no. 1, 1983, pp. 78–88; original manuscript written circa 1970.
- **^1.** Michael O. Rabin. "How to exchange secrets by oblivious transfer." Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981. Scanned handwriting + typed version on eprint.iacr.org archive (http://eprint.iacr.org/2005/187.pdf). Typed version available on Dousti's homepage (http://ce.sharif.edu/~dousti/home/papers/rabin_OT.pdf) (Alternate link on Google Docs (https://docs.google.com/viewer?a=v&pid=explorer&chrome=true&srcid=1NNsAGWFaxNp2O2h3-AAqP2uMfPXdliUJ2BIsdXScbK3ZYx7GJMXvE1hS7uKI&hl=en)).
- **^2.** S. Even, O. Goldreich, and A. Lempel, "A Randomized Protocol for Signing Contracts", Communications of the ACM, Volume 28, Issue 6, pg. 637–647, 1985. Paper at Catuscia Palamidessi's page (http://www.lix.polytechnique.fr/~catuscia/teaching/papers_and_books/SigningContracts.pdf)
- **^3.** Claude Crépeau. "Equivalence between two flavours of oblivious transfer". In Advances in Cryptology: CRYPTO '87, volume 293 of Lecture Notes in Computer Science, pages 350–354. Springer, 1988
- **^4.** Joe Kilian. "Founding Cryptography on Oblivious Transfer", Proceedings, 20th Annual ACM Symposium on the Theory of Computation (STOC), 1988. Paper at ACM portal (subscription required) (http://portal.acm.org/citation.cfm?id=62215)
- **^5.** Gilles Brassard, Claude Crépeau and Jean-Marc Robert. "All-or-nothing disclosure of secrets." In Advances in Cryptology: CRYPTO ' 86, volume 263 of LNCS, pages 234–238. Springer, 1986.
- **^6.** Moni Naor and Benny Pinkas. "Oblivious transfer with adaptive queries." In Advances in Cryptology: CRYPTO ' 99, volume 1666 of LNCS, pages 573–590. Springer, 1999.
- **^7.** Yuval Ishai and Eyal Kushilevitz. "Private simultaneous messages protocols with applications." In Proc. of ISTCS' 97, IEEE Computer Society, pages 174–184, 1997.
- **^8.** Bhavani Shankar, Kannan Srinathan and C. Pandu Rangan. "Alternative protocols for generalized oblivious transfer". In Proc. of ICDCN' 08, LNCS 4904, pages 304–309, 2008.

- **^9.** Tamir Tassa. "Generalized oblivious transfer by secret sharing". Designs, Codes and Cryptography, Volume 58:1, pages 11–21, January 2011. Paper at openu.ac.il (http://www.openu.ac.il/home/tamirtassa/Publications/got.pdf)
- **^10.** Moni Naor and Benny Pinkas (1990). Oblivious Polynomial Evaluation (http://www.wisdom.weizmann.ac.il/~naor/PAPERS/ope.pdf) 31st STOC
- **^11.** William Aiello, Yuval Ishai and Omer Reingold (2001) Priced Oblivious Transfer: How to Sell Digital Goods (https://www.iacr.org/archive/eurocrypt2001/20450118.pdf) EUROCRYPT '01 Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology, pages 119–135
- **^12.** Sven Laur and Helger Lipmaa (2007). "A New Protocol for Conditional Disclosure of Secrets And Its Applications" (http://www.cs.ut.ee/~lipmaa/papers/ll07). In Jonathan Katz and Moti Yung, editors, *ACNS*, *Lecture Notes in Computer Science* **4521**: 207–225. Springer, Heidelberg.
- **^13.** Vladimir Kolesnikov, Ranjit Kumaresan, Mike Rosulek, and Ni Trieu (2017). "Efficient batched oblivious prf with applications to private set intersection" (https://eprint.iacr.org/2016/799.pdf). In Edgar R.Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, ACM CCS 16, pages 818–829. ACM Press, October 2016.

External links

- Helger Lipmaa's collection of Web links on the topic

- Lo, H.-K. (1997). "Insecurity of quantum secure computations" (https://journals.aps.org/pra/abstract/10.1103/PhysRevA.56.1154). *Phys. Rev. A.* **56** (2): 1154. arXiv:quant-ph/9611031 (https://arxiv.org/abs/quant-ph/9611031). doi:10.1103/PhysRevA.56.1154 (https://doi.org/10.1103%2FPhysRevA.56.1154).