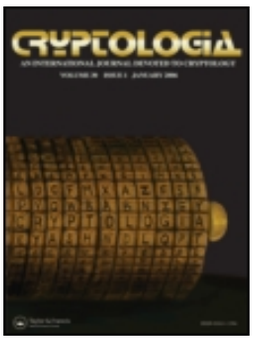

Cryptologia

ISSN: 0161-1194 (Print) 1558-1586 (Online) Journal homepage: https://www.tandfonline.com/loi/ucry20

Cryptanalysis and Improvement of a k-out-of-n Oblivious Transfer Protocol

Qinglong Wang & Jintai Ding

To cite this article: Qinglong Wang & Jintai Ding (2014) Cryptanalysis and Improvement of a k-out-of-n Oblivious Transfer Protocol, Cryptologia, 38:4, 370-376, DOI: 10.1080/01611194.2014.915261

To link to this article: https://doi.org/10.1080/01611194.2014.915261

Cryptologia, 38:370–376, 2014

ISSN: 0161-1194 print
DOI: 10.1080/01611194.2014.915261

Taylor & Francis
Taylor & Francis Group

Cryptanalysis and Improvement of a k-out-of-n Oblivious Transfer Protocol

QINGLONG WANG AND JINTAI DING

Abstract In this article, the authors cryptanalyze a k-out-of-n oblivious transfer protocol proposed in [12]. Their protocol is one of the most efficient k-out-of-n oblivious transfer protocols and is directly built from a 1-out-of-n oblivious transfer protocol. However, their analysis shows that the proposed k-out-of-n oblivious transfer protocol is insecure, though the primitive 1-out-of-n oblivious transfer protocol is secure. The weakness is that with high probability the receiver in their protocol can get all n secret messages encrypted by the sender. Finally, they fix the serious flaw and introduce an improved k-out-of-n oblivious transfer protocol without increasing any cost.

Keywords cryptanalysis, k-out-of-n oblivious transfer, oblivious transfer, security

1. Introduction

Oblivious transfer (OT) protocol is one of today's most important cryptographic technologies and is designed for various applications such as secret exchange, contract signing, private information retrieval, oblivious search, oblivious database queries, and secure function evaluation [1, 3, 9, 11]. Also, OT is an important foundation in cryptography used in many other cryptography protocols [10, 13, 17].

In 1981, Rabin first introduced an OT protocol [23] in which the sender, Alice, sends a message to the receiver, Bob, and would like Bob to acquire the message with probability $1/2$. 1-out-of-2 OT (OT_2^1) [14, 19, 22] is a protocol in which Alice sends two messages to Bob, and Alice wants Bob to acquire one of them, of his choice, while Alice remains oblivious to Bob's choice. In 1986, Brassard and colleagues proposed the 1-out-of-n OT (OT_n^1) [4, 5], which is an extension of OT_2^1. k-out-of-n OT (OT_n^k), in which Alice has n messages and Bob wants to get k of them, obliviously was first presented by Bellare and Micali in 1989 [2]. Since then, OT_n^k has become a hot research field, and many papers have been published [6, 7, 8, 15, 16, 18, 20, 24].

In 2010, Jain and Hari presented a OT_n^k and a OT_n^1 [12], which is a modification of [21]. The computational cost of their OT_n^k protocol is $n+k$ for Alice and $2k$ for Bob, and the transfer cost is $n+2k+2$. They introduced some advantages of their OT_n^k protocol in [12]. The security requirements of OT_n^k are

- Receiver's privacy: Alice should not be able to learn anything about the k messages that Bob selected.

Address correspondence to Jintai Ding, CPS Lab, Chongqing University, Chongqing, China; or Department of Mathematics, University of Cincinnati, Cincinnati, OH 45220, USA. E-mail: dingji@ucmail.uc.edu

- Sender's privacy: Bob should not be able to learn anything about the remaining $n-k$ messages that he did not select.

Jain and Hari's OT_n^k protocol was built by extending their OT_n^1 protocol. They claimed that their OT_n^k protocol was as secure as their OT_n^1 protocol. However, we found that it is not as secure. Cryptanalysis shows that their OT_n^k protocol is insecure, although their OT_n^1 protocol is secure. The flaw is that the receiver can learn all n secret messages with high probability, which violates the sender's privacy.

The rest of this article is organized as follows. In section 2, we give a brief introduction to Jain and Hari's OT_n^k protocol. In section 3, we show an attack against it. Section 4 introduces an improved OT_n^k protocol. Section 5 concludes the article.

2. Introduction of Jain and Hari's OT_n^k Protocol

2.1. System Parameters

Alice and Bob both agree upon a safe prime p[1] and a generator g of the group Z_p. Say that Alice has n secret messages $S_1, \ldots, S_n$ and that Bob wants to acquire k of them. Let there be a set of n distinct positive integers $\{x_1, x_2, \ldots, x_n\}$ known both to Alice and Bob. Each member x_i of the set $\{x_1, x_2, \ldots, x_n\}$ corresponds to the i^{th} secret message S_i.

2.2. Process of the Protocol

(1) Alice generates a random nonce N_{A_1} and sends the message $M_A = g^{N_{A_1}+\sum_{i=1}^{n} x_i} \bmod p$ to Bob.

(2) Bob selects $\{x_{i_1}, x_{i_2}, \ldots, x_{i_k}\} \subset \{x_1, \ldots, x_n\}$ that correspond to the secrets he wants to acquire and generates three nonces N_{B_1}, N_{B_2}, N_{B_3} such that $N_{B_3} = k \times N_{B_2}$, where k is a factor of N_{B_1}.

(3) Bob sends the messages $M_j = \left(\frac{M_A}{g^{x_{i_j}}}\right)^{\frac{N_{B_1} N_{B_2}}{N_{B_3}}} \bmod p$ to Alice for $j = 1, 2, \ldots, k$.

(4) Bob also sends $M_B = g^{N_{B_1}} \bmod p$ to Alice.

(5) Alice generates the nonce N_{A_2} and the set of keys $\{K_{A_1}, K_{A_2}, \ldots, K_{A_n}\}$ as $K_{A_j} = \left((M_B)^{N_{A_1}+\sum_{i=1}^{n} x_i - x_j}\right)^{N_{A_2}} \bmod p,\ j = 1, 2, \ldots, n.$

(6) Alice sends the messages $\left[M_j\right]^{N_{A_2}} \bmod p$ to Bob for $j = 1, 2, \ldots, k$.

(7) Bob calculates K_{B_j} as $\left[\left[M_j\right]^{N_{A_2}}\right]^{\frac{N_{B_3}}{N_{B_2}}} \bmod p,\ j = 1, 2, \ldots, k.$

(8) Alice sends all the secret messages encrypted under the respective keys. [S_i is encrypted under the generated key K_{A_j}. This is denoted by $\{S_1\}_{K_{A_1}}, \{S_2\}_{K_{A_2}}, \ldots, \{S_n\}_{K_{A_n}}$.]

(9) Bob can then decrypt the locked messages that he wished to learn using the keys $K_{B_j},\ j = 1, \ldots, k$ that he has generated.

3. Cryptanalysis of the OT_n^k Protocol

After analyzing the above protocol, we found that Bob can acquire all n secret messages with high probability. Therefore, Jain and Hari's OT_n^k protocol is not secure because it does not implement Alice's privacy. Details of our attack follow, but steps that are identical with those in 2.2 are omitted.

[1]A safe prime p is a prime number such that the discrete logarithm over $GF(p)$ is hard.

(2) Bob selects a $x_j \in \{x_1, \ldots, x_n\}$ satisfying

$$GCD\left(\sum_{i=1}^{n} x_i - x_j, p-1\right) = 1. \tag{1}$$

(When $\{x_1, \ldots, x_n\}$ is randomly selected, it is well known that (1) will be true with high probability.) Bob selects $x_j \in \{x_{i_1}, x_{i_2}, \ldots, x_{i_k}\}$ as the k messages he wants to acquire (without loss of generality, we assume $x_j = x_{i_2}$) and generates three nonces $N_{B_1}, N_{B_2}, N_{B_3}$ such that $N_{B_3} = k \times N_{B_2}$, where k is a factor of N_{B_1}.

(3) Bob sends the messages $M_1 = \left(\frac{M_A}{g^{\sum_{i=1}^{n} x_i}}\right)^{\frac{N_{B_1}N_{B_2}}{N_{B_3}}}$ and $M_j = \left(\frac{M_A}{g^{x_{i_j}}}\right)^{\frac{N_{B_1}N_{B_2}}{N_{B_3}}} \bmod p$, $j = 2, \ldots, k$ to Alice.

(5) Alice generates the nonce N_{A_2} and the set of keys $\{K_{A_1}, K_{A_2}, \ldots, K_{A_n}\}$ as

$$K_{A_j} = \left((M_B)^{N_{A_1} + \sum_{i=1}^{n} x_i - x_j}\right)^{N_{A_2}} \bmod p,\ j = 1, \ldots, n. \tag{2}$$

(7) Bob calculates

$$R_j = \left[\left[M_j\right]^{N_{A_2}}\right]^{\frac{N_{B_3}}{N_{B_2}}} \bmod p,\ j = 1, 2. \tag{3}$$

(9) Now, Bob can acquire all n messages $S_1, \ldots, S_n$ using the following steps:

Step 1. From (3), he has $R_1 = \left[\left[M_1\right]^{N_{A_2}}\right]^{\frac{N_{B_3}}{N_{B_2}}} = g^{N_{A_1}N_{A_2}N_{B_1}}$ and $R_2 = \left[\left[M_2\right]^{N_{A_2}}\right]^{\frac{N_{B_3}}{N_{B_2}}} =$

$$\left[\left[\left(\frac{M_A}{g^{x_{i_2}}}\right)^{\frac{N_{B_1}N_{B_2}}{N_{B_3}}}\right]^{N_{A_2}}\right]^{\frac{N_{B_3}}{N_{B_2}}} = \left(g^{N_{A_1} + \sum_{i=1}^{n} x_i - x_{i_2}}\right)^{N_{B_1}N_{A_2}} = g^{N_{B_1}N_{A_1}N_{A_2}} g^{N_{B_1}N_{A_2}\left(\sum_{i=1}^{n} x_i - x_{i_2}\right)}$$

Step 2. He computes $\alpha = \left(\sum_{l=1}^{n} x_l - x_{i_2}\right)^{-1} \bmod (p-1)$. [$\alpha$ exists because of (1)].

Step 3. He computes $R = \left((R_1)^{-1} R_2\right)^{\alpha} = g^{N_{B_1}N_{A_2}} \bmod p$.

Step 4. He computes the key K_{B_j} as

$$K_{B_j} = R_1 R^{\sum_{i=1}^{n} x_i - x_j} \bmod p \tag{4}$$

for $j = 1, \ldots, n$.

Step 5. He decrypts $\left\{S_j\right\}_{K_{A_j}}$ with the key K_{B_j} for $j = 1, \ldots, n$.

3.1. *Correctness of the Attack*

From (2), we have

$$\begin{aligned} K_{A_j} &= \left((M_B)^{N_{A_1} + \sum_{i=1}^{n} x_i - x_j}\right)^{N_{A_2}} = (M_B)^{N_{A_2}\left(N_{A_1} + \sum_{i=1}^{n} x_i - x_j\right)} \\ &= (M_B)^{N_{A_1}N_{A_2}} (M_B)^{N_{A_2}\left(\sum_{i=1}^{n} x_i - x_j\right)} = g^{N_{B_1}N_{A_1}N_{A_2}} g^{N_{B_1}N_{A_2}\left(\sum_{i=1}^{n} x_i - x_j\right)} \bmod p \end{aligned} \tag{5}$$

for $j = 1, \ldots, n$.

From (4), we have

$$\begin{aligned} K_{B_j} &= R_1(R)^{\sum_{i=1}^{n} x_i - x_j} = g^{N_{A_1} N_{A_2} N_{B_1}} \left(g^{N_{B_1} N_{A_2}}\right)^{\sum_{i=1}^{n} x_i - x_j} \\ &= g^{N_{B_1} N_{A_1} N_{A_2}} g^{N_{B_1} N_{A_2} \left(\sum_{i=1}^{n} x_i - x_j\right)} \bmod p \end{aligned} \tag{6}$$

for $j = 1, \ldots, n$.

From (5) and (6), we have $K_{A_j} = K_{B_j}$, $j = 1, \ldots, n$. Thus, Bob can decrypt all n encrypted messages $\{S_1\}_{K_{A_1}}, \{S_2\}_{K_{A_2}}, \ldots, \{S_n\}_{K_{A_n}}$.

The authors of [12] presented an example of their protocol. To illustrate how our attack works, we apply our attack method to their example.

Example: Alice is in possession of say five secret messages, S_1, S_2, S_3, S_4, S_5 (i.e., $n = 5$). They agree upon the safe prime $p = 23$, the generator $g = 5$ of the group Z_{23}, and the set $\{x_1, x_2, x_3, x_4, x_5\}$, which is selected as $\{1, 2, 3, 4, 5\}$. They also decide the number of secret messages to be transferred, $k = 2$.

(1) Alice generates the nonce $N_{A_1} = 4$ and sends $M_A = 5^{4+(1+2+3+4+5)} \bmod 23 \equiv 7$.
(2) Bob selects a $j \in \{1, 2, 3, 4, 5\}$ that satisfies Equation (1). Since $\gcd\left(\sum_{i=1}^{5} x_i - x_2, 23 - 1\right) = \gcd\left(\sum_{i=1}^{5} i - 2, 23 - 1\right) = \gcd(13, 22) = 1$, Bob chooses $j = 2$. Then Bob selects $\{1, 2\}$ as the two messages S_1, S_2 he wants to get. Bob also generates the nonces $N_{B_1} = 10$, $N_{B_2} = 6$ and $N_{B_3} = 12$. (Here, $N_{B_3} = k \times N_{B_2}$ where $k = 2$, which is a factor of N_{B_1}.)
(3) Bob calculates and sends $M_1 = \left(\frac{7}{19}\right)^5 \bmod 23 \equiv 12$ and $M_2 = \left(\frac{7}{2}\right)^5 \bmod 23 \equiv 7$.
(4) Bob also sends $M_B = 5^{10} \bmod 23 \equiv 9$.
(5) Alice generates the nonce $N_{A_2} = 8$ and calculates the following keys:
$K_{A_1} = \left(9^{19-1}\right)^8 \bmod 23 \equiv 9$, $K_{A_2} = \left(9^{19-2}\right)^8 \bmod 23 \equiv 6$, $K_{A_3} = \left(9^{19-3}\right)^8 \bmod 23 \equiv 4$
$K_{A_4} = \left(9^{19-4}\right)^8 \bmod 23 \equiv 18$ and $K_{A_5} = \left(9^{19-5}\right)^8 \bmod 23 \equiv 12$.
Alice encrypts S_1 with the key K_{A_1}, S_2 with the key K_{A_2} and so on.
(6) Alice calculates and sends $M_1^{N_{A_2}} \bmod p = 12^8 \bmod 23 \equiv 8$ and $M_2^{N_{A_2}} \bmod p = 7^8 \bmod 23 \equiv 12$ to Bob.
(7) Bob calculates $R_1 = 8^2 \bmod 23 \equiv 18$ and $R_2 = 12^2 \bmod 23 \equiv 6$.
(8) Alice sends all the encrypted messages to Bob, that is, $\{S_1\}_{K_{A_1}}$, $\{S_2\}_{K_{A_2}}$, $\{S_3\}_{K_{A_3}}$, $\{S_4\}_{K_{A_4}}$ and $\{S_5\}_{K_{A_5}}$.
(9) Bob does the following:
Step 1. He calculates $\alpha = \left(\sum_{i=1}^{5} i - 2\right)^{-1} \bmod (p-1) = (13)^{-1} \bmod 22 \equiv 17$.
Step 2. He computes $R = (18^{-1} \times 6)^{17} \bmod 23 \equiv 13$.
Step 3. He calculates
$K_{B_1} = 18 \times 13^{15-1} \bmod 23 \equiv 9$, $K_{B_2} = 18 \times 13^{15-2} \bmod 23 \equiv 6$, $K_{B_3} = 18 \times 13^{15-3} \bmod 23 \equiv 4$,
$K_{B_4} = 18 \times 13^{15-4} \bmod 23 \equiv 18$ and $K_{B_5} = 18 \times 13^{15-5} \bmod 23 \equiv 12$.

It can be found that $K_{B_j} = K_{A_j}$ is true for $j = 1, 2, 3, 4, 5$. Thus, Bob can get all messages by decrypting $\{S_j\}_{K_{A_j}}$ with the key K_{B_j}.

4. Improved OT_n^k Protocol

In this section, we give an improved OT_n^k protocol based on Jain and Hari. The improved OT_n^k protocol also includes nine steps, and we omit those steps which are same as in section 2.2.

(1) Alice generates random nonces N_{A_1} and r and sends the message $M_A = g^{N_{A_1}+r\sum_{i=1}^{n} x_i} \bmod p$ to Bob.

(5) Alice generates nonce N_{A_2} and the set of keys $\{K_{A_1}, K_{A_2}, \ldots, K_{A_n}\}$ as $K_{A_j} = \left((M_B)^{N_{A_1}+r\sum_{i=1}^{n} x_i - x_j}\right)^{N_{A_2}} \bmod p,\ j = 1, \ldots, n.$

Since r is randomly selected by Alice, it is obvious that our improved protocol is secure against the above attack.

5. Conclusion

We have introduced a concrete attack method on the OT_n^k protocol of [12] and have shown that it does not satisfy the sender's privacy. We have presented an improved OT_n^k protocol for which the computational cost and the transfer cost of our OT_n^k protocol are the same as that of [12]. Thus, our proposed OT_n^k protocol is as efficient as [12].

About the Authors

Qinglong Wang received his MS degree in cryptography from Xi'dian University, Xi'an, Shaanxi, China, in 2002, and his PhD degree in information security from Beijing Jiaotong University, Beijing, China, in 2009. He is currently an associate professor at Chang'an University, Xi'an, China. His research interests include public key cryptography and information security.

Jintai Ding received his PhD in Mathematics from Yale in 1995. He was a lecturer at Kyoto University from 1995–1998. He has been a professor in the Department of Mathematical Sciences at University of Cincinnati since 1998. He was a Humboldt Fellow 2006–2007. His main research interests are in cryptography, computational algebra, and information security.

Acknowledgments

The authors would like to thank the reviewers for their detailed reviews and constructive comments, which have helped improve the quality of this article. The authors would also like to especially thank Chris Christensen for useful discussions and help with polishing the article.

Funding

This work was supported by the Fundamental Research Funds for the Central Universities under Grant No. CHD2009JC146; the Science & Technology Funds of Ministry of Transport of the P.R. China under Grant No. 2012-364-208-600.

References

1. Aiello, B., Y. Ishai, and O. Reingold. 2001. Priced Oblivious Transfer: How to Sell Digital Goods. In *Proceedings of Advances in Cryptology—EUROCRYPT 2001*, LNCS, vol. 2045, Springer-Verlag, pp. 119–135.
2. Bellare, M., and S. Micali. 1989. Non-Interactive Oblivious Transfer and Applications. In *Proceedings of Advances in Cryptology—CRYPTO'89*, LNCS, vol. 435, Springer-Verlag, pp. 547–557.
3. Ben-Or, M., S. Goldwasser, and A. Wigderson. 1988. Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing (STOC'88)*, ACM, pp. 1–10.
4. Brassard, G., C. Crepeau, and J.-M. Robert. 1986. Information Theoretic Reductions among Disclosure Problems. In *Proceedings of 27th Annual Symposium on Foundations of Computer Science (FOCS'86)*, IEEE, pp. 427–437.
5. Brassard, G., C. Crepeau, and M. Santha. 1996. Oblivious Transfers and Intersecting Codes, *IEEE Transactions on Information Theory*, 42(6):1769–1780.
6. Chang, C.-C., and J.-S. Lee. 2009. Robust t-out-of-n Oblivious Transfer Mechanism Based on CRT, *Journal of Network and Computer Applications*, 32(1):226–235.
7. Chou, J.-S. 2012. A Novel k-out-of-n Oblivious Transfer Protocol from Bilinear Pairing, *Advances in Multimedia*, 2012:1–9.
8. Chu, C.-K., and W.-G. Tzeng. 2008. Efficient k-out-of-n Oblivious Transfer Schemes, *Journal of Universal Computer Science*, 14(3): 397–415.
9. Even, S., O. Goldreich, and A. Lempel. 1985. A Randomized Protocol for Signing Contracts, *Communications of the ACM*, 28(6): 637–647.
10. Goldreich, O., and R. Vainish. 1987. How to Solve Any Protocol Problem—An Efficiency Improvement. In *Proceedings of Advances in Cryptology—CRYPTO'87*, LNCS, vol. 293, Springer-Verlag, pp. 73–86.
11. Goldreich, O., and R. Vainish. 1997. How to Solve Any Protocol Problem—An Efficiency Improvement (Extended Abstract). www.wisdom.weizmann.ac.il
12. Jain, A., and C. Hari. 2010. A New Efficient Protocol for k-out-of-n Oblivious Transfer, *Cryptologia*, 34(4): 282–290.
13. Killian, J. 1988. Founding Cryptography on Oblivious Transfer. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC'88)*, ACM, pp. 20–31.
14. Lindell, Y. 2008. Efficient Fully-Simulatable Oblivious Transfer. In *CT-RSA'08*, Springer-Verlag, pp. 52–70.
15. Ma, X., L. Xu, and F. Zhang. 2011. Oblivious Transfer with Timed Release Receiver's Privacy, *Journal of Systems and Software*, 84(3): 460–464.
16. Mu, Y., J. Zhang, and V. Varadharajan. 2002. m out of n Oblivious Transfer. In *Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP'02)*, LNCS, vol. 2384, Springer-Verlag, pp. 395–405.
17. Murugesan, M., W. Jiang, E. Nergiz, and S. Uzunbaz. 2009. Homomorphic Encryption Based k-out-of-n Oblivious Transfer Protocols. http://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=2721&context=cstech.
18. Murugesan, M., W. Jiang, E. Nergiz, and S. Uzunbaz. 2011. k-out-of-n Oblivious Transfer Based on Homomorphic Encryption and Solvability of Linear Equations. In *Proceedings of CODASPY'11*, ACM, pp. 169–178.
19. Naor, M., and B. Pinkas. 2001. Efficient Oblivious Transfer Protocols. In *Proceedings of the 12th Annual Symposium on Discrete Algorithms (SODA'01)*, ACM/SIAM, pp. 448–457.
20. Naor, M., and B. Pinkas. 2005. Computationally Secure Oblivious Transfer, *Journal of Cryptology*, 18(1): 1–35.
21. Parakh, A. 2008. Oblivious Transfer Based on Key Exchange, *Cryptologia*, 32(1): 37–44.

22. Peikert, C., V. Vaikuntanathan, and B. Waters. 2008. A Framework for Efficient and Composable Oblivious Transfer. In *Proceedings of Advances in Cryptology—CRYPTO'08*, LNCS, vol. 5157, Springer-Verlag, pp. 554–571.
23. Rabin, M. O. 1981. *How to Exchange Secrets by Oblivious Transfer*. Technical Report TR-81, Aiken Computation Laboratory, Harvard University.
24. Wu, Q.-H., J.-H. Zhang, and Y.-M. Wang. 2003. Practical t-out-n Oblivious Transfer and its Applications. In *Proceedings of 5th International Conference on Information and Communications Security (ICICS'03)*, LNCS, vol. 2836, Springer-Verlag, pp. 226–237.