Oblivious Keyword Search

Wakaha Ogata[1] Kaoru Kurosawa[2]

[1]Tokyo Institute of Technology,
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan
wakaha@ss.titech.ac.jp

[2] Ibaraki University,
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan
kurosawa@cis.ibaraki.ac.jp

Abstract. In this paper, we introduce a notion of Oblivious Keyword Search (OKS). Let W be the set of possible keywords. In the commit phase, a database supplier $\mathcal{T}$ commits n data. In each transfer subphase, a user $\mathcal{U}$ can choose a keyword $w \in W$ adaptively and find $Search(w)$ without revealing w to $\mathcal{T}$, where $Search(w)$ is the set of all data which includes w as a keyword.

We then show two efficient protocols such that the size of the commitments is only $O(nB)$ regardless of the size of W, where B is the size of each data. It is formally proved that $\mathcal{U}$ learns nothing more than $Search(w)$ and $\mathcal{T}$ gains no information on the keywords which $\mathcal{U}$ searched for. We further present a more efficient adaptive OT_k^n protocol than the previous one [19] as an application of our first OKS protocol.

Keywords: oblivious transfer, blind signature, oblivious polynomial evaluation.

1 Introduction

1.1 Background

The notion of oblivious transfer (OT) was introduced by Rabin [22]. It has many flavors such as 1-out-of-2 OT (OT_1^2) [13], 1-out-of-n OT (OT_1^n), under the name of ANDOS [5, 6], adaptive k-out-of-n OT (adaptive OT_k^n) [19], and oblivious polynomial evaluation (OPE) [18]. Each of them is a two party protocol between a sender $\mathcal{S}$ and a chooser $\mathcal{C}$.

In a OT_1^2 protocol, $\mathcal{S}$ has two secret strings M_0 and M_1, and $\mathcal{C}$ has a secret bit b. $\mathcal{C}$ learns M_b, but nothing more. $\mathcal{S}$ gains no information on b. Essentially every known suggestion of public-key cryptography allows to implement OT_1^2 protocols. Therefore, OT_1^2 protocols can be based on factoring, Diffie-Hellman and so on. See [1] for ElGamal based OT_1^2 protocol, for example. [20] proved its security formally in the random oracle model. OT, OT_1^2 and OT_1^n are all equivalent in the information theoretic sense [5, 12, 7].

On the other hand, an adaptive OT_k^n protocol consists of a commit phase and a transfer phase. In the commit phase, a sender $\mathcal{S}$ commits n secret strings $M_1, \cdots, M_n$. In each transfer subphase j $(1 \leq j \leq k)$, a chooser $\mathcal{C}$ chooses an index i_j adaptively and obtains M_{i_j}. (i_j may depend on all the previous information $\mathcal{C}$ learned.) However, $\mathcal{C}$ learns nothing more and $\mathcal{S}$ gains no information on $i_1, \cdots, i_k$. Naor and Pinkas [19] showed two adaptive OT_k^n protocols, a DDH based protocol and a random oracle based one, such that each transfer subphase involves $\log_2 n$ invocations of a OT_1^2 protocol.

In a more theoretical aspect, Kilian [16] and Goldreich and Vainish [15] showed that we can implement general oblivious function evaluation by using a OT_1^2 protocol, i.e. $\mathcal{S}$ can let $\mathcal{C}$ evaluate any function $f(X)$ for the input x^* without revealing $f(X)$ to $\mathcal{C}$ while $\mathcal{S}$ gains no information on x^*.

OPE is a special case such that $f(X)$ is a polynomial over a field F. Naor and Pinkas [18] showed an efficient OPE protocol whose complexity does not depend on F, except that it uses a OT_1^2 protocol over F. A modified version of this protocol works if the polynomial reconstruction problem is hard [4, page 64]. (Naor and Pinkas first assumed that the noisy polynomial problem is hard in [18]. However, this assumption was shown to be weaker than expected in [4].)

1.2 Our Contribution

In this paper, we introduce a notion of Oblivious Keyword Search (OKS). In an OKS protocol, there is a database supplier who possesses some secret data. It allows a user to search and retrieve the data containing some keywords chosen by the user in such a way that the chosen keywords are unknown to the data supplier.

That is, let W be the set of possible keywords. In the commit phase, a database supplier $\mathcal{T}$ commits n data (through a CD-ROM or DVD). In each transfer subphase, a user $\mathcal{U}$ can choose a keyword $w \in W$ adaptively and find $Search(w)$ on the pay-per-view basis without revealing w to $\mathcal{T}$, where $Search(w)$ is the set of all data which includes w as a keyword. This is a new and interesting cryptographic primitive that should have real-world applications.

We then show two efficient protocols, one is based on the one-more RSA inversion problem and the other is based on the polynomial reconstruction problem. In our protocols, the size of the commitments is only $O(n \times B)$ regardless of the size of W, where B is the length of each data. It is formally proved that $\mathcal{U}$ learns nothing more and $\mathcal{T}$ gains no information on the keywords which $\mathcal{U}$ searched for in the random oracle model. We further present a more efficient adaptive OT^n_k protocol than the previous one [19] as an application of our second OKS protocol.

More formally, a k-out-of-n OKS protocol (OKS^n_k protocol) is a two-party protocol between a database supplier $\mathcal{T}$ and a user $\mathcal{U}$. In the commit phase, $\mathcal{T}$ commits n data $B_1, \ldots, B_n$ such that

$$B_i = (w_i, c_i)$$

where $w_i \in W$ is a keyword and c_i is a content. The transfer phase consists of k subphases. In each subphase j $(1 \leq j \leq k)$, $\mathcal{U}$ chooses a keyword $w^*_j \in W$ adaptively and learns $Search(w^*_j)$. (Remember that $Search(w^*_j)$ is the set of all data which includes w^*_j as a keyword.) However, $\mathcal{U}$ learns nothing more about the data and $\mathcal{S}$ gains no information on the keywords $w^*_1, \cdots, w^*_k$ which $\mathcal{U}$ searched for.

Our first OKS^n_k protocol uses the RSA blind signature scheme [9] which is often used for e-cash systems. Bellare et.al proved that the RSA blind signature scheme is secure if the One-More RSA-inversion problem is hard [2, 3]. We prove that our first OKS^n_k protocol is secure under the same assumption.

Our first protocol is very efficient. However, the intractability assumption on the One-More RSA-inversion problem is new and very strong [3, page 4]. Therefore, we show our second OKS^n_k protocol which is based on a more widely accepted assumption.

Our second OKS^n_k protocol uses an OPE protocol. It is known that there exists an OPE protocol if the polynomial reconstruction problem is hard [18, 4]. Our second protocol is secure under this assumption.

We further present a more efficient adaptive OT^n_k protocol than the previous one [19] as an application of our first OKS^n_k protocol. At each transfer subphase, it requires executing the RSA blind signature scheme once while the previous scheme [19] requires $\log_2 n$ invocations of a OT_1^2 protocol. The proposed adaptive OT_k^n protocol is secure if the One-More RSA-inversion problem is hard.

See the following table for comparison.

Protocol	Assumption	Each Transfer Phase
Our first OKS_k^n	One-More RSA	RSA-BS once
Our secondOKS_k^n	Polynomial reconst.	OPE once
Our adaptive OT_k^n	One-More RSA	RSA-BS once
Adaptive OT_k^n of [19]	for example, DDH	$\log_2 n$ of OT_1^2

(BS stands for "blind signature scheme".)

1.3 Comparison with Adaptive OT_k^n

Remember that $\{w_1, \cdots, w_n\} \subset W$ is the set of keywords which actually appear in the database, where W is the set of all possible keywords. We require that

- $\mathcal{U}$ does not know $\{w_1, \cdots, w_n\}$.
- Suppose that w_i appears L_i times in the data base. Then $\mathcal{U}$ should not be able to know even L_i.

Actually, we can construct an OKS_k^n protocol by using an adaptive OT_k^n protocol as follows. Consider a $|W| \times n$ matrix such that the ith row includes all the indices of data whose keyword is $\hat{w}_i$, where $\hat{w}_i$ is the ith element of W. Then the size of the sender's commitments will be $O(n|W| + nB)$ because some keyword $\hat{w} \in W$ may appear in all the n data.

In the proposed OKS_k^n protocols, on the contrary, the size of the commitments is only $O(nB)$ independently of W.

1.4 Other Related Work

In [10], the private retrieval by keywords problem is discussed in the context of private information retrieval (PIR) [11, 17, 14, 8]. In this problem, $\mathcal{T}$ has n strings $M_1, \cdots, M_n$ and $\mathcal{U}$ has a keyword w. A solution to this problem is a protocol which allows $\mathcal{U}$ to find out if there exists M_j such that $M_j = w$. That is, the output of $\mathcal{U}$ is *yes* or *no*. This problem is clearly different from ours.

In [23], Song, Wagner and Perrig considered the following scenario. Suppose that Alice does secret keyword searches playing with Bob. Then Alice first encrypts the data and gives the ciphertexts to Bob. On the other hand, in our paper, Bob encrypts the data and gives the ciphertexts to Alice. Hence our problem is different from [23] and it is impossible to apply their technique to our problem.

1.5 Organization of the Paper

In Sec. 2, we introduce a model and definitions of OKS_k^n protocols. In Sec. 3, we show our first OKS_k^n protocol which is based on the RSA blind signature scheme. In Sec. 4, we present a more efficient adaptive OT_k^n protocol than the previous one [19] as an application of our first OKS_k^n protocol. In Sec. 5, we show our second OKS_k^n protocol which is based on OPE.

Throughout the paper, all players are probabilistic polynomial-time interactive Turing machines. l denotes a security parameter.

2 Oblivious Keyword Search

In this section, we introduce a notion of Oblivious Keyword Search (OKS). A k-out-of-n OKS protocol (OKS_k^n protocol) is a two-party protocol between a database supplier $\mathcal{T}$ and a user $\mathcal{U}$ as follows. Let W be the set of keywords. In the commit phase, $\mathcal{T}$ commits n data $B_1, \ldots, B_n$ such that

$$B_i = (w_i, c_i)$$

where $w_i \in W$ and c_i is a content. Define

$$Search(w) = \{(i, c_i) \mid w_i = w\}.$$

The transfer phase consists of k subphases. At each subphase j ($1 \le j \le k$), $\mathcal{U}$ chooses a keyword $w_j^* \in W$ adaptively and learns $Search(w_j^*)$. However, $\mathcal{C}$ learns nothing more and $\mathcal{S}$ gains no information on $w_1^*, \cdots, w_k^*$. More formally,

(The User's Security) A protocol is secure for the user if for any malicious database supplier $\widetilde{\mathcal{T}}$, the view of $\widetilde{\mathcal{T}}$ for $w_1^*, \cdots, w_k^*$ and that for $\tilde{w}_1^*, \cdots, \tilde{w}_k^*$ are computationally indistinguishable for any $(w_1^*, \cdots, w_k^*) \neq (\tilde{w}_1^*, \cdots, \tilde{w}_k^*)$.

(The Database's Security) We make a comparison with an *ideal world*: A trusted third party (TTP) first receives $(B_1, \cdots, B_n)$ from a database supplier. The TTP next tells $Search(w_j^*)$ to the user on her request w_j^* for $1 \leq j \leq k$.

Definition 2.1 *We say that a protocol is $\epsilon(l)$-secure for the database if for any malicious user $\widetilde{\mathcal{U}}$, there exists a simulator $\mathcal{A}$ that plays the role of the user in the ideal world such that for any polynomial time distinguisher D,*

$$|\Pr(D(\textit{the output of}\, \widetilde{\mathcal{U}}) = 1) - \Pr(D(\textit{the output of}\, \mathcal{A}) = 1)| < \epsilon(l). \quad (1)$$

Definition 2.2 *We say that a protocol is secure for the database if the above $\epsilon(l)$ is negligible.*

Definition 2.3 *We say that a protocol is an OKS_k^n protocol if it is secure for the user and the database.*

3 OKS_k^n Protocol Based on RSA Blind-Signature

In this section, we show an efficient OKS_k^n protocol under the intractability assumption of the one-more RSA-inversion problem. In this OKS_k^n protocol, RSA blind signature scheme is executed once at each transfer subphase.

3.1 RSA Blind Signature Scheme

In a blind signature scheme, Bob can ask Alice to sign a message M without revealing M. Let (N, e) be an RSA public key of Alice and let d be the secret key. Let H be a random hash function. Then the RSA blind signature scheme is described as follows.

(Step 1) Suppose that Bob wishes to get Alice's signature of a message M. Then he first chooses a random number r and computes

$$Y = r^e H(M) \bmod N.$$

He sends Y to Alice.

(Step 2) Alice computes $S' = Y^d \bmod N$ and sends it back to Bob.

(Step 3) Bob obtains a signature S of M as

$$S = S'/r = H(M)^d \bmod N.$$

To define the security, we consider a *forger* who is allowed to play the role of Bob. His task is to compute $m+1$ message-sgnature pairs while he is allowed to make at most $m \in \mathsf{N}$ queries to Alice for some m. We say that the RSA blind signature scheme is secure if the success probability of any polynomial time bounded *forger* is negligible.

Formally [21, 2], let $\mathcal{F}$ be a forger who has access to RSA-inversion oracle (Alice) and hash oracle H. Consider the following experiment, where l is a security parameter.

Experiment $\mathbf{EXP}_{\mathcal{F}}^{forge}(l)$: Run $\mathcal{F}$. Suppose that

$$((M_1, x_1), \cdots, (M_{m+1}, x_{m+1})) \leftarrow \mathcal{F}(N, e, l),$$

where $m \in \mathsf{N}$. If the following are all true, then return 1 else return 0:

- $\forall i \in \{1, \cdots, m+1\} : H(M_i) = x_i^e \bmod N$.
- Messages $M_1, \cdots, M_{m+1}$ are all distinct.
- $\mathcal{F}$ made at most m queries to its RSA-inversion oracle.

Definition 3.1 The RSA blind signature scheme is *polynomially-secure against one-more forgery* if the probability $\Pr(\text{EXP}_{\mathcal{F}}^{forge}(l) = 1)$ is negligible for any forger $\mathcal{F}$ whose time-complexity is polynomial in the security parameter l.

Bellare et al. proved that the RSA blind signature scheme is secure if the RSA known target inversion problem (RSA-KTI) is hard [2, 3]. In RSA-KIT problem, an adversary is given $m+1$ random targets $y_1, \cdots, y_{m+1} \in Z_N$. His task is to compute $y_1^d, \cdots, y_{m+1}^d \bmod N$ while he is allowed to make at most m queries to RSA-inversion oracle.

Formally, let $\mathcal{A}$ be an adversary who has access to RSA-inversion oracle. Consider the following experiment, where l is a security parameter and let $m : \mathsf{N} \rightarrow \mathsf{N}$ be a function of l.

Experiment $\mathbf{EXP}_{\mathcal{A},m}^{inv}(l)$: Choose $y_i \in Z_N^*$ randomly for $i = 1$ to $m(l)+1$. Run $\mathcal{A}$. Suppose that

$$(x_1, \cdots, x_{m(l)+1}) \leftarrow \mathcal{A}(N, e, l, y_1, \cdots, y_{m(l)+1}).$$

If the following are both true, then return 1 else return 0:

- $\forall i \in \{1, \cdots, m(l)+1\} : y_i = x_i^e \bmod N$.

- $\mathcal{A}$ made at most $m(l)$ oracle queries.

Definition 3.2 The RSA known target inversion problem (RSA-KTI) is *hard* if the probability $\Pr(\mathrm{EXP}^{inv}_{\mathcal{A},m}(l) = 1)$ is negligible for all polynomially bounded $m(\cdot)$ and for any adversary $\mathcal{A}$ whose time-complexity is polynomial in the security parameter l.

Proposition 3.1 *If RSA-KTI is hard, then the RSA blind signature scheme is polynomially-secure against one-more forgery.*

RSA-KTI is also called the one-more RSA inversion problem.

3.2 OKS^n_k Protocol Based on RSA Blind Signature

Let G be a pseudo-random generator.

Commit phase $\mathcal{T}$ generates a public key (N, e) and a secret key d of RSA. $\mathcal{T}$ publishes (N, e). Next for $i = 1, \ldots, n$, $\mathcal{T}$ computes

$$\begin{aligned} K_i &= (H(w_i))^d \bmod N, \\ E_i &= G(w_i \| K_i \| i) \oplus (0^l \| c_i), \end{aligned}$$

where $||$ denotes concatination. $\mathcal{T}$ sends $E_1, \ldots, E_n$ to $\mathcal{U}$.

Transfer phase At each transfer phase j,

(Step 1) $\mathcal{U}$ chooses a keyword w_j^*.

(Step 2) $\mathcal{U}$ chooses a random element r and computes

$$Y = r^e H(w_j^*) \bmod N.$$

$\mathcal{U}$ sends Y to $\mathcal{T}$.

(Step 3) $\mathcal{T}$ computes $K' = Y^d \bmod N$ and sends it to $\mathcal{U}$.

(Step 4) $\mathcal{U}$ computes

$$K = K'/r = H(w_j^*)^d \bmod N.$$

Then let $J = \emptyset$. For $i = 1, \ldots, n$, $\mathcal{U}$ computes

$$(a_i \| b_i) = E_i \oplus G(w_j^* \| K \| i).$$

If $a_i = 0^l$, then $\mathcal{U}$ adds (i, b_i) to J.

3.3 Security

Correctness: At each transfer phase j, the final J is equal to $Search(w_j^*)$ with probability at least $1 - n/2^l$.

User's security: $\mathcal{T}$ has no information on $w_1^*, \cdots, w_k^*$ because they are blinded in the RSA blind signature scheme.

We next prove the database's security by assuming that RSA-KTI is hard. For any malicious $\widetilde{\mathcal{U}}$ who queries to $\mathcal{T}$ (for RSA blind signatures) k times, we will show a simulator $\mathcal{A}$ in the ideal world. In the commit phase, $\mathcal{A}$ generates (N, e, d) and sends (N, e) to $\widetilde{\mathcal{U}}$. $\mathcal{A}$ also chooses $E_1, \cdots, E_n$ randomly and sends them to $\widetilde{\mathcal{U}}$. In the transfer phase, $\mathcal{A}$ behaves in the same way as $\mathcal{T}$. $\mathcal{A}$ can do this because $\mathcal{A}$ chooses (N, e, d) by itself. Finally $\mathcal{A}$ outputs the output of $\widetilde{\mathcal{U}}$.

$\mathcal{A}$ simulates H as follows. If $\widetilde{\mathcal{U}}$ queries w to H for the first time, then $\mathcal{A}$ chooses a random string y_w and sets $H(w) = y_w$. It is clear that $\mathcal{A}$ simulates H perfectly.

$\mathcal{A}$ simulates G as follows. Wlog, we can assume that $\widetilde{\mathcal{U}}$ queries w to H before $\widetilde{\mathcal{U}}$ queries $w\|K\|i$ to G. Let $cnt = 0$. Let QA-list be empty. Suppose that $\widetilde{\mathcal{U}}$ queries $w\|K\|i$ to G for the first time.

1. If $K \neq H(w)^d \bmod N$, then $\mathcal{A}$ sets $G(w\|K\|i)$ at random.

2. Suppose that $K = H(w)^d \bmod N$.
   If w is included in QA-list, then goto 4.
   Else, let $cnt := cnt + 1$.

3. If $cnt \geq k + 1$, then $\mathcal{A}$ sets $G(w\|K\|i)$ at random.
   Else $\mathcal{A}$ queries w to the TTP and receives $Search(w)$. $\mathcal{A}$ adds $(w, Search(w))$ to QA-list.

4. Suppose that w is included in QA-list, i.e., $(w, Search(w)) \in$ QA-list.
   If i is included in $Search(w)$, i.e. $(i, c_i) \in Search(w)$ for some c_i, then $\mathcal{A}$ sets
   $$G(w\|K\|i) = E_i \oplus (0^l\|c_i).$$
   Otherwise, $\mathcal{A}$ sets $G(w\|K\|i)$ at random.

Let **BAD** be the event that $cnt \geq k + 1$. If **BAD** does not occur, then $\mathcal{A}$ simulates G perfectly. Note that Pr(**BAD**) is the probability that $\widetilde{\mathcal{U}}$ succeeds in the one-more forgery attack on the RSA blind signature scheme.

From Proposition 3.1, it is negligible if the RSA known target inversion problem (RSA-KTI) is hard. Consequently, the outputs of $\mathcal{A}$ and $\widetilde{\mathcal{U}}$ are indistinguishable if the RSA known target inversion problem (RSA-KTI) is hard.

Therefore, we obtain the following theorem.

Theorem 3.1 *The above OKS_k^n protocol is secure if the RSA known target inversion problem (RSA-KTI) is hard.*

4 Application to an Adaptive OT_k^n Protocol

In this section, we first show that if there exists an OKS_k^n protocol, then there exists an adaptive OT_k^n protocol. We next present an adaptive OT_k^n protocol which is obtained from this implication and our OKS_k^n protocol of Sec. 3.2. The security is proved similarly to Sec. 3.2 under the intractability assumption of RSA-KTI.

This protocol executes the RSA blind signature scheme once at each transfer subphase. Therefore, it is more efficient than the previous adaptive OT_k^n protocol [19] which requires $\log_2 n$ invocations of a OT_1^2 protocol at each transfer subphase.

4.1 Adaptive Oblivious Transfer

An adaptive k-out-of-n Oblivious Transfer OT_k^n protocol consists of a commit phase and a transfer phase. In the commit phase, the sender $\mathcal{S}$ commits n secret strings $M_1, \cdots, M_n$. In each transfer subphase j $(1 \leq j \leq k)$, a chooser $\mathcal{C}$ chooses an index i_j adaptively and obtains M_{i_j}. (i_j may depend on all the previous information $\mathcal{C}$ learned.) However, $\mathcal{C}$ learns nothing more than $M_{i_1}, \cdots, M_{i_k}$ and $\mathcal{S}$ gains no information on $i_1, \cdots, i_k$. More formally,

(The Chooser's Security) For any malicious sender $\widetilde{\mathcal{S}}$, the view of $\widetilde{\mathcal{S}}$ for $i_1, \cdots, i_k$ and that for $i'_1, \cdots, i'_k$ are computationally indistinguishable for any $(i_1, \cdots, i_k) \neq (i'_1, \cdots, i'_k)$.

(The Sender's Security) We make a comparison with an *ideal world*: A trusted third party (TTP) first receives $(M_1, \cdots, M_n)$ from the sender. The TTP next tells M_{i_j} to the chooser on her request i_j for $1 \leq j \leq k$.

The requirement is that for any malicious chooser $\widetilde{\mathcal{C}}$, there exists a simulator $\mathcal{A}$ that plays the role of the chooser in the ideal world such that the output of $\mathcal{A}$ is computationally indistinguishable from the output of $\widetilde{\mathcal{C}}$.

4.2 New Adaptive OT_k^n Protocol

Theorem 4.1 *Suppose that there exists an OKS_k^n protocol. Then there exists an adaptive OT_k^n protocol which executes the OKS_k^n protocol once.*

(Proof) In the OKS_k^n protocol, let $w_i = i$ for $1 \leq i \leq n$. Then it is easy to see that we obtain an adaptive OT_k^n protocol.

Q.E.D

Our adaptive OT_k^n protocol is described as follows.

Commit phase $\mathcal{S}$ generates a public key (N, e) and a secret key d of RSA. $\mathcal{S}$ publishes (N, e). Next for $i = 1, \ldots, n$, $\mathcal{S}$ computes

$$\begin{aligned} K_i &= (H(i))^d, \\ E_i &= G(K_i \| i) \oplus M_i, \end{aligned}$$

where $M_1, \ldots, M_n$ are the secret messages of $\mathcal{S}$. $\mathcal{S}$ sends $E_1, \ldots, E_n$ to $\mathcal{C}$.

Transfer phase At each transfer phase j,

(Step 1) $\mathcal{C}$ chooses i_j.

(Step 2) $\mathcal{C}$ chooses a random element r and computes

$$Y = r^e H(i_j) \bmod N.$$

$\mathcal{C}$ sends Y to $\mathcal{S}$.

(Step 3) $\mathcal{S}$ computes $K' = Y^d \bmod N$ and sends it to $\mathcal{C}$.

(Step 4) $\mathcal{C}$ computes

$$K = K'/r = H(i_j)^d \bmod N$$

and obtains M_{i_j} as $M_{i_j} = E_{i_j} \oplus G(K \| i_j)$.

5 OKS_k^n Protocol Based on OPE

In this section, we show an OKS_k^n protocol such that an oblivious polynomial evaluation (OPE) protocol is executed at each transfer subphase. It is known that there exists an OPE protocol if the polynomial reconstruction problem is hard [4, page 64]. Our protocol is secure under the same assumption.

Let F be a finite field.

Problem 5.1 *(Polynomial reconstruction) Given as input integers k, t, and n points $(x_1, y_1), \cdots, (x_n, y_n) \in \mathsf{F}^2$, outputs all univariate polynomila P of degree at most k such that $y_i = P(x_i)$ for at least t values of i.*

5.1 Oblivious Polymonial Evaluation (OPE)

In an Oblivious Polynomial Evaluation $OPE(\tau, \mathsf{F})$ protocol, $\mathcal{S}$ has a secret polynomial $f(X)$ of degree at most τ over a finite field F, and $\mathcal{C}$ has a secret field element x^*. In the protocol, $\mathcal{C}$ learns $f(x^*)$, but nothing more. $\mathcal{S}$ gains no information about x^*. The security is formally defined as follows.

(The Chooser's Security) A protocol is secure for the chooser if for any $x_0, x_1 \in \mathsf{F}$ and for any malicious sender $\widetilde{\mathcal{S}}$, the view of $\widetilde{\mathcal{S}}$ for $x = x_0$ and that for x_1 are computationally indistinguishable given $f(X)$.

(The Sender's Security) We make a comparison with an *ideal world* in which a trusted third party (TTP) receives $f(X)$ from a sender and x^* from a chooser. He then tells $f(x^*)$ to the chooser.

Definition 5.1 *We say that a protocol is $\epsilon(l)$-secure for the sender if for any malicious chooser $\widetilde{\mathcal{C}}$, there exists a simulator $\mathcal{A}$ that plays the role of the chooser in the ideal world such that for any polynomial time distinguisher D,*

$$|\Pr(D(\textit{the output of } \widetilde{\mathcal{C}}) = 1) - \Pr(D(\textit{the output of } \mathcal{A}) = 1)| < \epsilon(l).$$

Definition 5.2 *We say that a protocol is secure for the sender if the above $\epsilon(l)$ is negligible.*

Definition 5.3 *We say that a protocol is an $OPE(\tau, \mathsf{F})$ protocol if it is secure for the chooser and the sender.*

5.2 Protocol for $k = 1$

In what follows, suppose that $|\mathsf{F}| \geq |W|$ and let G be a pseudo-random generator.

For simplicity, we first show a OKS_1^n protocol.

Commit phase $\mathcal{T}$ chooses a random polynomial $f(X) = aX + b$ over F such that $a \neq 0$. $\mathcal{T}$ then computes

$$\begin{aligned} K_i &= f(w_i) \\ E_i &= G(K_i \| i) \oplus (0^l \| c_i) \end{aligned}$$

for $i = 1, \ldots, n$, where $\|$ denotes concatenation. $\mathcal{T}$ sends $E_1, \ldots, E_n$ to $\mathcal{U}$. (Note that $f(w_i) \neq f(w_j)$ if $w_i \neq w_j$ because $a \neq 0$.)

Transfer phase

(Step 1) $\mathcal{U}$ chooses a keyword w^*.

(Step 2) $\mathcal{T}$ and $\mathcal{U}$ run an $OPE(1, \mathsf{F})$ protocol so that $\mathcal{U}$ learns $K_0 = f(w^*)$.

(Step 3) Let $J = \phi$. For $i = 1, \ldots, n$, $\mathcal{U}$ computes

$$(a_i, b_i) = E_i \oplus G(K_0 \| i).$$

If $a_i = 0^l$, then $\mathcal{U}$ adds (i, b_i) to J.

Correctness: It is easy to see that the final J is equal to $Search(w^*)$ with probability at least $1 - n/2^l$.

User's security: $\mathcal{T}$ has no information on w^* from the chooser's security of the $OPE(1, \mathsf{F})$ protocol.

We next prove the database's security in the random oracle model. In the random oracle model, we say that a protocol is $(q, \epsilon(l))$-secure for the database if eq.(1) holds for any malicious user $\widetilde{\mathcal{U}}$ who makes at most q queries to the random oracle G. Then the following theorem holds.

Theorem 5.1 *Suppose that the underlying $OPE(1, \mathsf{F})$ protocol is $\epsilon'(l)$-secure for the sender. Then our protocol is $(q, \epsilon(l))$-secure for the database, where*

$$\epsilon(l) = \epsilon'(l) + \frac{q}{|\mathsf{F}| - 1}.$$

A proof is given in the next subsection.

5.3 Proof of Theorem 5.1

Wlog, we can consider that any malicious user $\widetilde{\mathcal{U}}$ consists of $(\widetilde{\mathcal{U}}_0, \widetilde{\mathcal{U}}_1, \widetilde{\mathcal{U}}_2)$ as follows.

- $\widetilde{\mathcal{U}}_0$ receives $E_1, \cdots, E_n$ from $\mathcal{T}$ and outputs $view^* = (\alpha, E_1, \cdots, E_n)$, where α is the random input to $\widetilde{\mathcal{U}}$.
- $\widetilde{\mathcal{U}}_1$ has an auxiliarly input $view^*$. It executes the $OPE(1, \mathsf{F})$ protocol with $\mathcal{T}$ and outputs $view_1$, where $view_1$ is the view that $\widetilde{\mathcal{U}}_1$ saw and $view_1$ includes $view^*$.
- $\widetilde{\mathcal{U}}_2$ has an auxiliarly input $view_1$ and outputs the output of $\widetilde{\mathcal{U}}$.

There exists a simulator $\mathcal{B}$ which simulates $\widetilde{\mathcal{U}}_1$ from the sender's security of the OPE protocol. That is, the output of $\mathcal{B}$ is indistinguishable from that of $\widetilde{\mathcal{U}}_1$ if they have the same auxiliarly input $view^*$. By using this $\mathcal{B}$, we consider an imaginal protocol $(\mathcal{T}',\widetilde{\mathcal{U}}')$ as follows, where $\mathcal{T}'$ plays the role of the TTP in the ideal world of the OPE protocol.

Step 1'. $\mathcal{T}'$ chooses $f(X)$ and computes $E_1, \cdots, E_n$ in the same way as in the original protocol. $\mathcal{T}'$ then sends $E_1, \cdots, E_n$ to $\widetilde{\mathcal{U}}'$.

Step 2'. $\widetilde{\mathcal{U}}'$ runs $\mathcal{B}$ with the auxiliarly input $view^* = (\alpha, E_1, \cdots, E_n)$, where α is a random string. In this process, suppose that $\mathcal{B}$ queries w^* to the TTP in the ideal world of OPE. Then

Step 2'-1. $\widetilde{\mathcal{U}}'$ queries w^* to $\mathcal{T}'$.

Step 2'-2. $\mathcal{T}'$ gives $K_0 = f(w^*)$ to $\widetilde{\mathcal{U}}'$. $\widetilde{\mathcal{U}}'$ then gives K_0 to $\mathcal{B}$.

$\mathcal{B}$ finally outputs $view_1'$

Step 3'. $\widetilde{\mathcal{U}}'$ runs $\widetilde{\mathcal{U}}_2$ with the auxiliarly input $view_1'$. $\widetilde{\mathcal{U}}'$ then outputs the output of $\widetilde{\mathcal{U}}_2$.

Now we show a simulator $\mathcal{A}$ for $\widetilde{\mathcal{U}}$ of our OKS protocol. $\mathcal{A}$ behaves similarly to $\mathcal{T}'$ by using $\mathcal{B}$ and $\widetilde{\mathcal{U}}_2$ as subroutines as follows.

Step a. $\mathcal{A}$ chooses $E_1, \cdots, E_n$ randomly and runs $\mathcal{B}$ with the auxiliarly input $view^* = (\alpha, E_1, \cdots, E_n)$, where α is a random string.

Step b. If $\mathcal{B}$ queries w^*, then $\mathcal{A}$ chooses a random string K_0 which has never been queried to the random oracle G and gives K_0 to $\mathcal{B}$.

$\mathcal{A}$ also queries w^* to the TTP in the ideal world of OKS and receives $Search(w^*)$.

Step c. $\mathcal{B}$ finally outputs $view_1''$.

Step d. $\mathcal{A}$ runs $\widetilde{\mathcal{U}}_2$ with the auxiliarly input $view_1''$ and outputs the output of $\widetilde{\mathcal{U}}_2$.

$\mathcal{A}$ simulates the random oracle G as follows. Suppose that $(K\|i)$ is queried to G. If it happened before K_0 is chosen (at Step b), then $\mathcal{A}$ sets $G(K\|i)$ at random. Otherwise, if $K = K_0$ and i is included in $Search(w^*)$, then $\mathcal{A}$ sets

$$G(K\|i) = E_i \oplus (0^l\|c_i),$$

where $(i, c_i) \in \mathit{Search}(w^*)$. Else, $\mathcal{A}$ sets $G(K\|i)$ at random.

Let β denote the output of $\widetilde{\mathcal{U}}$, β' denote the output of $\widetilde{\mathcal{U}}'$ and β'' denote the output of $\mathcal{A}$. Then we can show the following lemmas for any fixed distinguisher D.

Lemma 5.1

$$|\Pr(D(\beta) = 1) - \Pr(D(\beta') = 1)| < \epsilon'(l).$$

(Proof) Suppose that this lemma does not hold. Then consider a distinguisher D' between $\{\mathit{view}_1\}$ and $\{\mathit{view}_1\}$ as follows. On input $\mathit{view}_1/\mathit{view}'_1$, D' runs $\widetilde{\mathcal{U}}_2$ with the auxiliarly input $\mathit{view}_1/\mathit{view}'_1$. D' then gives the output β/β' of $\widetilde{\mathcal{U}}_2$ to D. Then we have

$$|\Pr(D'(\mathit{view}_1) = 1) - \Pr(D'(\mathit{view}'_1) = 1)| \geq \epsilon'(l)$$

from our assumption.

On the other hand, $\mathcal{B}$ is a simulator of $\widetilde{\mathcal{U}}_1$ and they have the same auxiliarly input view^*. Therefore, their outputs must be indistinguishable. Hence we must have

$$|\Pr(D'(\mathit{view}_1) = 1) - \Pr(D'(\mathit{view}'_1) = 1)| < \epsilon'(l).$$

However, this is a contradiction. Q.E.D.

Lemma 5.2

$$|\Pr(D(\beta') = 1) - \Pr(D(\beta'') = 1)| \leq \frac{q}{|\mathsf{F}| - 1}.$$

(Proof) In the imaginal protocol $(\mathcal{T}', \widetilde{\mathcal{U}}')$:

- Let $\mathbf{BAD}_1$ be the event that $\widetilde{\mathcal{U}}'$ queries some $(K\|i)$ to G such that $K = f(w_i)$ before K_0 is given to $\widetilde{\mathcal{U}}'$.
- Let $\mathbf{BAD}_2$ be the event that $\widetilde{\mathcal{U}}'$ queries some $(K\|i)$ to G such that $K = f(w_i)$ and i is not included in $\mathit{Search}(w^*)$ after K_0 is given to $\widetilde{\mathcal{U}}'$.

Let **BAD** be the event that $\mathbf{BAD}_1$ or $\mathbf{BAD}_2$ occurs. If **BAD** does not occur in the imaginal protocol $(\mathcal{T}', \widetilde{\mathcal{U}}')$, then $\mathcal{A}$ simulates G perfectly. Therefore,

$$\Pr(D(\beta') = 1 \mid \neg\mathbf{BAD}) = \Pr(D(\beta'') = 1).$$

It is clear that

$\Pr(D(\beta')=1) = \Pr(D(\beta')=1, \mathbf{BAD}) + \Pr(D(\beta')=1 \mid \neg\mathbf{BAD})\Pr(\neg\mathbf{BAD})$.

Therefore,

$$\begin{array}{rcl}
\Pr(D(\beta')=1) & \leq & \Pr(\mathbf{BAD}) + \Pr(D(\beta'')=1) \\
\Pr(D(\beta')=1) & \geq & \Pr(D(\beta')=1 \mid \neg\mathbf{BAD})\Pr(\neg\mathbf{BAD}) \\
 & = & \Pr(D(\beta'')=1)(1-\Pr(\mathbf{BAD})) \\
 & = & \Pr(D(\beta'')=1) - \Pr(D(\beta'')=1)\Pr(\mathbf{BAD})) \\
 & \geq & \Pr(D(\beta'')=1) - \Pr(\mathbf{BAD}))
\end{array}$$

Hence

$$|\Pr(D(\beta')=1) - \Pr(D(\beta'')=1)| \leq \Pr(\mathbf{BAD}).$$

We next estimate $\Pr(\mathbf{BAD})$. Suppose that $\widetilde{\mathcal{U}}'$ queries to G at most q_1 times before K_0 is given and at most q_2 times after K_0 is given, where $q_1 + q_2 \leq q$. It is easy to see that for any w_i and for any K,

$$\Pr(f(w_i) = K) = \frac{1}{|\mathsf{F}|},$$

where the probability is taken over $f(X) = aX + b$ such that $a \neq 0$. Therefore,

$$\Pr(\mathbf{BAD}_1 \text{ occurs}) \leq \frac{q_1}{|\mathsf{F}|}.$$

Similarly,

$$\Pr(f(w_i) = K \mid f(w^*) = K_0) \leq \frac{1}{|\mathsf{F}|-1}$$

for any $w_i \neq w^*$ and any K. Therefore,

$$\Pr(\mathbf{BAD}_2 \text{ occurs}) \leq \frac{q_2}{|\mathsf{F}|-1}.$$

Hence

$$\Pr(\mathbf{BAD}) \leq \Pr(\mathbf{BAD}_1) + \Pr(\mathbf{BAD}_2) \leq \frac{q}{|\mathsf{F}|-1}. \tag{2}$$

Q.E.D.

Therefore we obtain that

$$\begin{array}{rl}
 & |\Pr(D(\beta)=1) - \Pr(D(\beta'')=1| \\
= & |\Pr(D(\beta)=1) - \Pr(D(\beta')=1) + \Pr(D(\beta')=1) - \Pr(D(\beta'')=1| \\
\leq & |\Pr(D(\beta)=1) - \Pr(D(\beta')=1)| + |\Pr(D(\beta')=1) - \Pr(D(\beta'')=1| \\
< & \epsilon'(l) + \dfrac{q}{|\mathsf{F}|-1}.
\end{array}$$

5.4 Protocol for $k \geq 2$

We next show a OKS_k^n protocol for $k \geq 2$. Let $P(k, \mathsf{F})$ be the set of polynomials of degree at most k over F. We assume that

$$\gcd(k, |\mathsf{F}| - 1) = 1. \tag{3}$$

Definition 5.4 *For a given database $B_1, \cdots, B_n$, we say that $f(X) \in P(k, \mathsf{F})$ is good if*

$$f(w_i) \neq f(w_j)$$

for any $w_i \neq w_j$.

Lemma 5.3 *If $f(X) \in P(k, \mathsf{F})$ is chosen randomly, then*

$$\Pr(f(X) \textit{ is good}) \geq 1 - \binom{n}{2} \frac{1}{|\mathsf{F}|}.$$

(Proof) Let

$$f(X) = a_0 + a_1 X + \cdots + a_k X^k.$$

Then for $x_1 \neq x_2$, it holds $f(x_1) = f(x_2)$ if and only if

$$a_1(x_1 - x_2) + \cdots + a_k(x_1^k - x_2^k) = 0.$$

For the above equation, a_k is uniquely determined from given $a_0, \cdots, a_{k-1}$ because $x_1^k \neq x_2^k$ from eq.(3). Therefore,

$$\begin{aligned} \Pr(f(x_1) = f(x_2)) &= \frac{|\{a_0, \cdots, a_{k-1}\}|}{|\{a_0, \cdots, a_k\}|} \\ &= \frac{|\mathsf{F}|^k}{|\mathsf{F}|^{k+1}} \\ &= \frac{1}{|\mathsf{F}|} \end{aligned}$$

Now the probability that $f(w_i) = f(w_j)$ for some $w_i \neq w_j$ is at most $\binom{n}{2} \frac{1}{|\mathsf{F}|}$. Therefore, we obtain Lemma 5.3.

Q.E.D.

Suppose that $|\mathsf{F}| \gg \binom{n}{2}$. Then a randomly chosen $f(X) \in P(k, \mathsf{F})$ is good with high probability from lemma 5.3.

Now our OKS_k^n protocol is obtained by slightly modifying the OKS_1^n protocol of Sec. 5.2 as follows.

- In the commit phase, $\mathcal{T}$ chooses a good $f(X) \in P(k, \mathsf{F})$ randomly by trial and error.
- At Step 2 of the transfer phase, $\mathcal{T}$ and $\mathcal{U}$ run an $OPE(k, \mathsf{F})$ protocol.

The rest of the protocol is the same as Sec. 5.2. The correctness and the user's security are clear. We can prove the database's security similarly to Theorem 5.1.

6 Conclusion

In this paper, we introduced a notion of Oblivious Keyword Search (OKS). We then showed two efficient protocols such that the size of the commitments is only $O(nB)$ regardless of the size of W, where nB is the size of the database and W is the set of all possible keywords.

- The first scheme assumes the intractability of the one-more RSA-inversion problem and the second one assumes the intractability of the polynomial reconstruction problem.
- The first scheme is more efficient. The second scheme is based on a more widely accepted assumption.

We further presented a more efficient adaptive OT_k^n protocol than the previous one [19] as an application of our first OKS protocol.

Usually, each content would have more than one keywords. Therefore, it will be a further work to construct an efficient protocol which can handle more than one keywords. It will also be a further work to derive a lower bound on the size of commitments.

Acknowledgement

We would like to thank Dr. Yasuhiro Ohtaki and Dr. Masaru Kamada of Ibaraki University for useful discussion.

References

[1] M. Bellare and S. Micali, "Non-interactive oblivious transfer," Proc. of Crypto '89, LNCS Vol. 435, pp. 547–557 (1990).

[2] M. Bellare, C. Namprempre, D. Pioncheval, M. Semanko, "The Power of RSA Inversion Oracles and the Security of Chaum's RSA-Based Blind Signature Scheme," Proc. of Financial Cryptography 2001, LNCS vol. 2339, pp. 319–338 (2001)

[3] M. Bellare, C. Namprempre, D. Pioncheval, M. Semanko, "The One-More-RSA-Inversion Problems and the security of Chaum's Blind Signature Scheme," An extended version of [2]. http://www-cse.ucsd.edu/users/mihir/crypto-research-papers.html

[4] D. Bleichenbacher and P. Nguyen, "Noisy polynomial interpolation and noisy Chinese remaindering," Proc. of Eurocrypt '2000, LNCS Vol. 1807, pp. 53–69 (2000).

[5] G. Brassard, C. Crépeau and J. M. Robert, "Information theoretic reduction among disclosure problems," 27th IEEE Symposium on Foundations of Computer Science, pp. 168–173 (1986).

[6] G. Brassard, C. Crépeau and J. M. Robert, "All-or-nothing disclosure of secrets," Proc. of Crypto '86, LNCS Vol. 263, pp. 234–238 (1987).

[7] C. Cachin, "On the foundations of oblivious transfer," Proc. of Eurocrypt '98, LNCS Vol. 1403, pp. 361–374 (1998).

[8] C. Cachin, S. Micali and M. Stadler, "Computationally private informational retrieval with polylogarithmic communication," Proc. of Eurocrypt '99, LNCS Vol. 1592, pp. 402–414 (1999).

[9] D. Chaum, "Blind signatures for untraceable payments," Proc. of Crypto '82, Plenum Press, pp. 199-203 (1982).

[10] B. Chor, N.Gilboa and M.Naor, "Private information retrieval by keywords," manuscript (1998)

[11] B. Chor, O. Goldreich, E. Kushilevitz and M. Sudan, "Private information retrieval," Journal of the ACM 45(6), pp. 965–982 (1998).

[12] C. Crépeau, "Equivalence between two flavors of oblivious transfers," Proc. of Crypto '87, LNCS Vol. 293, pp. 350–354 (1988).

[13] S. Even, O. Goldreich and A. Lempel, "A randomized protocol for signing contracts," Communications of the ACM 28, pp. 637–647 (1985).

[14] Y. Gertner, Y. Ishai, E. Kushilevitz and T. Malkin, “Protecting data privacy in private data retrieval schemes,” 30th ACM Symposium on Theory of Computing, pp. 151–160 (1998).

[15] O. Goldreich and R. Vainish, “How to solve any protocol problem: an efficient improvement,” Proc. of Crypto ’87, LNCS Vol. 293, pp. 73–86 (1988).

[16] J. Kilian, “Founding cryptography on oblivious transfer,” 20th ACM Symposium on Theory of Computing, pp. 20–31 (1988).

[17] E. Kushilevitz, R. Ostrovsky, “Replication is not needed: single database, computationally-private informational retrieval” 38th IEEE Symposium on Foundations of Computer Science, pp. 364–373 (1997).

[18] M. Naor and B. Pinkas, “Oblivious transfer and polynomial evaluation,” 31st ACM Symposium on Theory of Computing, pp. 145–254 (1999).

[19] M. Naor and B. Pinkas, “Oblivious transfer with adaptive queries,” Proc. of Crypto ’99, LNCS Vol. 1666, pp. 573–590 (1999).

[20] M. Naor and B. Pinkas, “Efficient oblivious transfer protocols,” 12th Annual Symposium on Discrete Algorithms (SODA), pp. 448–457 (2001).

[21] D.Pointcheval and J. P. Stern, “Provably secure blind signature schemes,” Proc. of Asiacrypt ’96, LNCS Vol. 1163, pp. 252–265 (1996).

[22] M. Rabin, “How to exchange secrets by oblivious transfer,” Technical Report TR 81, Aiken Computation Lab, Harvard University (1981).

[23] D. Song, D. Wagner and A. Perrig “Practical Techniques for Searches on Encrypted Data,” IEEE Symposium on Security and Privacy (2000)