A t-Private k-Database Private Information Retrieval Scheme

Carlo Blundo, Paolo D'Arco, and Alfredo De Santis

Dipartimento di Informatica ed Applicazioni
Università di Salerno, 84081 Baronissi (SA), Italy

e-mail: {carblu,paodar,ads}@dia.unisa.it

July 30, 2002

Abstract

A private information retrieval scheme enables a user to *privately* recover an item from a public accessible database.

In this paper we present a private information retrieval scheme for k replicated databases. The scheme is information-theoretically secure against coalitions of databases of size $t \leq k-1$. It improves the communication complexity of the scheme described in [12] for coalitions of size $\frac{k}{2} \leq t \leq k-1$.

1 Introduction

User's privacy issue has received a lot of attention in recent years. Many efforts have been produced to cover different aspects of user's interaction with private and public entities or organizations. An important aspect concerns with recovering information from public available databases. Indeed, a curious or dishonest database operator can follow the user's queries to infer which item of information he is interested in. A practical and significant scenario can be, for example, a medical information database. The concept of private information retrieval scheme, (PIR scheme, for short) was introduced in [7]. Such a scheme enables a user to retrieve an item of information from a database maintaining, at the same time, privacy with respect to the database operator. The authors of [7] showed that, if a single database is available in the system, unconditional privacy can be achieved only sending to the user the entire database. Of course, this solution is unacceptable from a communication complexity point of view. Surprisingly, they showed that allowing *copies* of the database, it is possible to do better. Therefore, they considered the following distributed environment: the network holds k copies of the database. A user, to keep secret the item he is interested in, sends different queries to the k databases. Then, he retrieves his item of information by computing a simple function of the received answers. Each query assures that the database operator does not gain any information on the item recovered by the user. Assuming that the database stores n bits, and the user needs a single bit, they presented a construction for 2 databases with communication complexity $\mathcal{O}(n^{\frac{1}{3}})$, a construction for a constant number k of databases with communication complexity $\mathcal{O}(n^{\frac{1}{k}})$, and a construction for $\frac{1}{3}\log n$ databases with communication complexity polylogarithmic in n. The first construction uses concepts of Coding Theory, while the second and the third ones use polynomial interpolation techniques.

Ambainis [1], proposed a PIR scheme for k copies of the database with communication complexity $\mathcal{O}(n^{1/(2k-1)})$, improving the result of [7].

Two interesting extensions of the PIR model have been presented in [16] and [11]. The first addresses the problem of privately read and *write* into the database, while the second one considers a symmetric privacy condition, i.e., the database's privacy must be protected, too.

Chor and Gilboa [5], began studying PIR schemes in the *computational setting*, requiring the privacy condition to be satisfied against a curious database operator polynomially bounded. They gave a PIR scheme for 2 databases with communication complexity $\mathcal{O}(n^{\epsilon})$, for any $\epsilon > 0$.

Kushilevitz and Ostrovsky [13], proved that communication complexity can be subpolynomial in n even if the databases is not replicated. More precisely, using the Quadratic Residuosity Assumption, they presented a PIR scheme computationally private, with a *single* database, achieving communication complexity $\mathcal{O}(n^{\epsilon})$, for any $\epsilon > 0$.

Recently, Cachin et al., in [4], introducing a new cryptographic assumption, have shown that in the computational setting, can be constructed a single database PIR scheme with communication complexity polylogarithmic in the length of the database.

An interesting approach to private information retrieval has been proposed by Di Crescenzo at al. [8]. They have shown how communication complexity of the PIR problem can be improved using some additional service-providers selling "commodities" to the users that will be used to privately retrieve information from the database. Their results hold for both the information-theoretic and the computational setting. Other papers focussing on private information retrieval schemes are [2, 3, 9, 10, 14, 15]. A major drawback of all above mentioned PIR schemes is the assumption that the user knows the physical address of the sought item. This problem has been efficiently solved in [6].

Information-theoretic privacy with a single database requires communicating the whole database. While, with database replication it is possible to reduce communication complexity. Notice that all the schemes information-theoretically private with replicated database proposed in the literature assume *absence* of communication between the databases. This assumption is unsuitable and difficult to achieve in the "real" world.

In [7], it has been proposed a scheme information-theoretic private *even if a coalition* of t databases can communicate and collude to break the scheme. Recently, Ishai and Kushilevitz [12], improved the known upper bounds on Information Theoretic PIR Schemes. More precisely, they showed a 1-private PIR scheme with the same asymptotic communication complexity of the scheme proposed in [1] but with a smaller constant factor. Then, for

t-private schemes, they proposed a generalization of the 1-private construction which improves the communication complexity of the schemes presented in [7].

Our result: In this paper we assume that coalitions of databases can collude to infer what item of information the user is interested in. We present a PIR scheme for k database copies unconditionally private against any coalition of size up to $k-1$ with communication complexity $2 \cdot k \cdot n^{\frac{1}{2}}$. Since the best scheme that can be constructed using the technique presented in [12] with k database and coalitions of size $\frac{k}{2} \leq t \leq k-1$ has communication complexity equal to $4 \cdot k \cdot n^{\frac{1}{2}}$, our scheme achieves an enhancement of a factor 2. The scheme we propose is elegant and simple. Notice that, communication complexity is the main performance evaluation parameter for private information retrieval schemes, and even slight improvements can be of some interest from a practical point of view.

2 Notation and Model

In this section we define private information retrieval schemes.

Let $\{0,1\}^n$ be the set of binary strings of length n. Denote with $[n] \stackrel{\triangle}{=} \{1, \ldots, n\}$. The database is a string $X = x_1, \ldots, x_n \in \{0,1\}^n$, replicated k times across the network. Each string is hold by a server; let $\mathcal{DB}_1, \ldots, \mathcal{DB}_k$ be the k servers storing the database. The user holds a *private* input $i \in [n]$, which represents the index of the bit the user is interested in, and a private source of randomness. In [6] the authors present a simple and modular way to privately access data by keywords. The solution they propose combine any conventional search structure with any underlying PIR scheme. Moreover, in [6] it has been presented a general transformation from PIR scheme to retrieval by keywords schemes.

A private information retrieval scheme can be seen as a game: The user, to recover bit x_i from the databases in a secure way, generates k queries for servers $\mathcal{DB}_1, \ldots, \mathcal{DB}_k$ using his private source r of random bits and the index i. The servers compute answers, based on the received queries and the content of the database, and send these answers to the user. Finally, the user, by means of some computation involving the servers' answers, the index i, and the random bits, is able to recover the bit x_i he is interested in. Moreover, any subset of the servers $\mathcal{DB}_1, \ldots, \mathcal{DB}_k$, of size $t \leq k-1$, putting together the user's queries, does not gain any information on the index i.

The queries, the answers, and the bit recovering operation can be represented by mathematical functions; while, the privacy condition can be modelled with a probabilistic requirement.

Let ℓ_r, ℓ_q, and ℓ_a be integers representing the lengths of, respectively, random strings generated by the private source of the user, query strings sent to the database servers, and answer strings received by the users from the servers.

Definition 2.1 *Let $X = x_1 \ldots x_n \in \{0,1\}^n$ be a database of n bits, and let t and k be integers such that $1 \leq t \leq k-1$. A t-private k-database Private Information Retrieval Scheme consists of the following $2k+1$ functions*

- k *query functions,* $Q_1, \ldots, Q_k : [n] \times \{0,1\}^{l_r} \mapsto \{0,1\}^{l_q}$*;*

- k *answer functions,* $A_1, \ldots, A_k : \{0,1\}^n \times \{0,1\}^{l_q} \mapsto \{0,1\}^{l_a}$*;*

- *a reconstruction function,* $R : [n] \times \{0,1\}^{l_r} \times (\{0,1\}^{l_a})^k \mapsto \{0,1\}$*;*

satisfying the two following conditions

`Correctness:` *For every* $x \in \{0,1\}^n$*,* $i \in [n]$*, and* $r \in \{0,1\}^{l_r}$

$$R(i, r, A_1(x, Q_1(i,r)), \ldots, A_k(x, Q_k(i,r))) = x_i.$$

`Privacy:` *For every* $i, j \in [n]$*,* $s_1, \ldots, s_t \in [k]$*, with* $1 \le t \le k-1$ *and* $(q_1, \ldots, q_t) \in (\{0,1\}^{\ell_q})^t$*,*

$$Pr[(Q_{s_1}(i,r), \ldots, Q_{s_t}(i,r)) = (q_1, \ldots, q_t)] = Pr[(Q_{s_1}(j,r), \ldots, Q_{s_t}(j,r)) = (q_1, \ldots, q_t)],$$

where the probability is taken over uniformly chosen $r \in \{0,1\}^{\ell_r}$*.*

The correctness condition of the above definition requires that each user can recover the bit in which he is interested. The (unconditional) privacy condition means that the joint distribution of t queries must be independent from the index i. This is enough to guarantee that a coalition of t servers does not gain any information on the index i from the received queries.

3 Our Protocol

In this section we describe a PIR scheme which is private against coalition of at most $k-1$ servers.

The main idea underlying the protocol is the following: viewing the database as a sequence of consecutive blocks, the user asks each server to xor-ing bit a bit, a certain number of them, and to receive back the resulting block. From the server point of view, this request looks like a totally random request, independent from any index of the database. On the other hand, the queries are constructed in such a way that, by xor-ing bit a bit all the blocks the servers send to the user, it comes up a single block of the database, which is exactly the block containing the bit the user is interested in. Recovering the bit from this block becomes straightforward.

Let us set up our notation. For any $S, S' \subseteq [\ell]$, we denote with $S \otimes S'$ the set

$$S \otimes S' = \{e \in [\ell] : e \in (S \cup S') \setminus (S \cap S')\}.$$

When $S' = \{e\}$ for any $e \in [\ell]$, to simplify the notation we denote with $S \otimes e$ the set $S \otimes \{e\}$, that is

$$S \otimes e = \begin{cases} S \cup \{e\} \text{ if } e \notin S \\ S \setminus \{e\} \text{ if } e \in S. \end{cases}$$

As we have pointed out, the database $X = x_1 \ldots x_n \in \{0,1\}^n$ is replicated k times across the network, and it is stored by the servers $\mathcal{DB}_1, \ldots, \mathcal{DB}_k$. Assume that X is divided into ℓ blocks of length $b = \lceil \frac{n}{\ell} \rceil$, i.e., $X = B_1 B_2 \ldots B_\ell$, where $B_j = x_{(j-1)b+1} \ldots x_{jb}$ for each $j = 1, \ldots, \ell$. If necessary, the last block can be padded with zeroes to have length b.

The protocol works as follows:

To recover bit x_i, the user $\mathcal{U}$ executes the following steps:

1. Compute $m = \lceil \frac{i}{b} \rceil$
2. For $j = 1, \ldots, k-1$, randomly choose $S_j \subseteq [\ell]$
3. Generate the sequence
   (a) $Q_1 = S_1$.
   (b) $Q_j = S_j \otimes S_{j-1}$, for $j = 2, \ldots, k-1$.
   (c) $Q_k = S_{k-1} \otimes m$.
4. For $j = 1, \ldots, k$, compute the characteristic sequence I_{Q_j} of ℓ bits, i.e., for each $e \in [\ell]$, $I_{Q_j}[e] = 1$ iff $e \in Q_j$.
5. For $j = 1, \ldots, k$, send the query I_{Q_j} to the server $\mathcal{DB}_j$.

The steps executed by server $\mathcal{DB}_j$, for $j = 1, \ldots, k$, are the following:

1. Compute the answer
$$R_j = \bigodot_{e : I_{Q_j}[e]=1} B_e,$$
   where the symbol $\bigodot$ denotes the bitwise xor operation.
2. Send the block R_j to the user $\mathcal{U}$.

Finally, the user $\mathcal{U}$ performs the following operations to recover the desired bit x_i:

1. Compute the block

$$B = x'_1 \dots x'_b = \bigodot_{j=1}^{k} R_j$$

2. Compute $s \leftarrow i - (m-1)b$

3. Recover $x_i \leftarrow x'_s$

CORRECTNESS. In the following we will prove that the above protocol satisfies the Correctness property of Definition 2.1. The reason is that when the user computes the xor of the k server replies R_j, he gets the block $B = B_m$, which contains the bit x_i he is interested in. To see that this condition is guaranteed we prove the following lemma.

Lemma 3.1 *Let $[\ell] = \{1, \dots, \ell\}$ be a set of ℓ elements, and let $m \in [\ell]$. For $j = 1, \dots, k$, let $S_j \subseteq [\ell]$ be a subset of elements randomly chosen in $[\ell]$. The sequence of subsets $Q_1, \dots, Q_k$, where $Q_1 = S_1$, $Q_j = S_j \otimes S_{j-1}$, for $j = 2, \dots, k-1$, and $Q_k = S_{k-1} \otimes m$, is such that m is the only element which appears an odd number of times in the sequence $Q_1, \dots, Q_k$.*

Proof. Let U be a *multiset* [1] defined as follows

$$U = \bigcup_{j=1}^{k} Q_j = S_1 \cup \left\{ \bigcup_{j=2}^{k-1} [(S_j \cup S_{j-1}) \setminus (S_j \cap S_{j-1})] \right\} \cup (S_{k-1} \otimes m)\,.$$

Since $(S_j \cup S_{j-1}) \setminus (S_j \cap S_{j-1})$ can be written as $(S_j \setminus (S_j \cap S_{j-1})) \cup (S_{j-1} \setminus (S_j \cap S_{j-1}))$, the above equality becomes

$$\begin{aligned}
\bigcup_{j=1}^{k} Q_j &= S_1 \cup \bigcup_{j=2}^{k-1} [S_j \setminus (S_j \cap S_{j-1})] \cup \bigcup_{j=2}^{k-1} [S_{j-1} \setminus (S_j \cap S_{j-1})] \bigcup (S_{k-1} \otimes m) \\
&= S_1 \cup \left\{ \left[\bigcup_{j=2}^{k-1} S_j \cup \bigcup_{j=2}^{k-1} S_{j-1} \right] \setminus \left[\bigcup_{j=2}^{k-1} (S_j \cap S_{j-1}) \cup \bigcup_{j=2}^{k-1} (S_j \cap S_{j-1}) \right] \right\} \cup \\
&\quad (S_{k-1} \otimes m)\,. \\
&= S_{k-1} \cup (S_{k-1} \otimes m) \cup \left\{ \left[\bigcup_{j=1}^{k-2} S_j \cup \bigcup_{j=1}^{k-2} S_j \right] \setminus \left[\bigcup_{j=2}^{k-2} (S_j \cap S_{j-1}) \cup \bigcup_{j=2}^{k-2} (S_j \cap S_{j-1}) \right] \right\}.
\end{aligned}$$

[1] In a multiset the same element can appear more than once.

It is easy to see that each element $\neq m$ appears in the sequence $Q_1, \ldots, Q_k$ an even number of times, while m appears an odd number since the $\otimes$ operator "flips" its membership in $Q_k = S_{k-1} \otimes m$. ∎

As we said before, the correctness of our protocol directly follows from Lemma 3.1. Indeed, the structure of the queries assures that, when the user $\mathcal{U}$ computes the xor of the replies R_j, he obtains exactly the block $B = B_m$ which contains the bit x_i. The last step of the protocol enables him to recover x_i from B by a simple computation.

Privacy. Our protocol is secure against coalitions of databases of size $t \leq k-1$. Privacy condition requires that each group of t queries is independent from the index i, which determines the bit in which the user is interested in. Indeed, let $Q_1, \ldots, Q_k$ be the set of queries generated by the user $\mathcal{U}$ to retrieve the bit x_i, and let $Q_{i_1}, \ldots, Q_{i_{k-1}}$ be a generic subset of these queries. For each $j \in [n]$, there exists a query Q^j such that the sequence $Q_{i_1}, \ldots, Q_{i_{k-1}}, Q^j$ enables the recovering of x_j. The query Q^j can be constructed as follows: Let U be a multiset defined as $U = \cup_{s=1}^{k-1} Q_{i_s}$. For all $e \in U$, let $m_e = |\{i_s : e \in Q_{i_s}\}|$ (m_e is the multiplicity of e in the multiset U). If $m_j \equiv 1 \bmod 2$ we set $Q^j = \{e \in [n] \setminus \{j\} : m_e \equiv 1 \bmod 2\}$ otherwise, we set $Q^j = \{e \in [n] \setminus \{j\} : m_e \equiv 1 \bmod 2\} \cup \{j\}$. By the structure of the queries, generated during the execution of the protocol, it results that the set Q^j is unique.
Hence, for any coalition of databases holding $k-1$ queries, the indices $j \in [n]$ are uniformly distributed.

Complexity. Notice that the user $\mathcal{U}$ sends k queries of ℓ bits to the servers $\mathcal{DB}_1, \ldots, \mathcal{DB}_k$; while, he receives k blocks each of $b = \lceil \frac{n}{\ell} \rceil$ bits. Hence, the communication complexity is equal to

$$(k \cdot \ell + k \cdot b) = k \cdot (\ell + \lceil \frac{n}{\ell} \rceil).$$

It is easy to see that, for a fixed n, the function $f(\ell) = \ell + \lceil \frac{n}{\ell} \rceil$, defined for $\ell \in [1, 2, \ldots, n]$, reaches its minimum for $\ell = \lceil \sqrt{n} \rceil$. Therefore, choosing $\ell = \lceil \sqrt{n} \rceil$, the communication complexity of our protocol becomes $2 \cdot k \cdot \lceil \sqrt{n} \rceil$, which is $\mathcal{O}(n^{1/2})$.

The above analysis can be synthesized in the following theorem

Theorem 3.2 *Let k and t be integers such that $1 \leq t \leq k-1$. There exist t-private k-database private information retrieval schemes with communication complexity equal to $2k\lceil \sqrt{n} \rceil$.*

Comparison. Let us compare the communication complexity of our scheme with the communication complexity of the scheme proposed in [12]. The protocol therein described works essentially as follows: the database is seen as a vector x in a *data space* X whose elements have size $n = \ell^d$. The user represents the index i as a d-tuple $(i_1, \ldots, i_d)$ and, for $h = 1, \ldots, d$, shares the *canonical* ℓ-vectors e_h among the k servers using a t-private linear secret sharing scheme. Each server performs a local computation on its shares, resulting in a collection of vectors belonging to the data space X, and returns to $\mathcal{U}$ the *inner product* of the database x with each of these vectors. Finally, the user reconstructs x_i by taking a fixed linear combination (depending on i) of the answers. Theorem 2 of [12] establishes that such a protocol, secure against coalitions of t databases, has communication complexity equal to

$$k \cdot \binom{k-1}{t} d \cdot n^{1/d} + k \cdot d \cdot n^{1/d},$$

where d is an integer such that the length of the database can be written as $n = \ell^d$, for some integer ℓ. On the other hand, Corollary 2 of [12] implies that, given k databases, d must be equal to 2 for a $(k-1)$-private scheme. In this case, the overall communication complexity of the scheme is

$$\mathcal{CC} = 4 \cdot k\sqrt{n}.$$

The authors of [12] showed that a better communication complexity can be obtained only increasing the number of available copies of the database. More precisely, Claim 5 of [12] states that to achieve communication complexity $\mathcal{O}(n^{1/3})$, it must be $k \geq 2 \cdot t$. Therefore, given k servers, for any security threshold t such that $\frac{k}{2} \leq t < k-1$, the best scheme that can be constructed using the technique described in [12] is the $(k-1)$-private scheme with communication complexity equal to $4 \cdot k\sqrt{n}$. Hence, for each $\frac{k}{2} \leq t < k-1$, our scheme is more efficient than the one in [12] for a factor 2.

For $t < \frac{k}{2}$, the technique described in [12] yields schemes with communication complexity asymptotically better than $2 \cdot k \cdot \sqrt{n}$.

References

[1] A. Ambainis, *Upper Bound on the Communication Complexity of Private Information Retrieval Scheme*, Proc. of the 24th ICALP, Lecture Notes in Computer Science, vol. 1256, Springer-Verlag, pp. 401-407, 1997.

[2] A. Beimel, Y. Ishai, E. Kushilevitz, and T. Malkin, *One-Way Functions are essential for Single-Server Private Information Retrieval*, Proc. of the 31st Annual ACM Symposium on Theory of Computing (STOC), pp. 89-98, 1999.

[3] A. Beimel, Y. Ishai, and T. Malkin, *Reducing the Servers' Computation in Private Information Retrieval: PIR with Preprocessing*, Proc. of Crypto 2000, Lecture Notes in Computer Science, vol. 1880, Springer-Verlag, pp. 56-74, 2000.

[4] C. Cashin, S. Micali, and M. Stadler, *Computationally Private Information Retrieval with Polylogarithmic Communication*, Proc. of Eurocrypt '99, Lecture Notes in Computer Science, vol. 1592, Springer-Verlag, pp. 402-414, 1999.

[5] B. Chor and N. Gilboa, *Computationally Private Information Retrieval*, Proc. of the 29th Annual ACM Symposium on Theory of Computing (STOC), pp. 26-38, 1997.

[6] B. Chor, N. Gilboa, and M. Naor, *Private Information Retrieval by Keywords*, Theory of Cryptography Library, 98-03, 1998. An on-line version of the paper can be found at `http://philby.ucsd.edu/cryptolib/psfiles/98-03.ps`.

[7] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, *Private Information Retrieval*, Journal of ACM, vol. 45, pp. 965-981, 1998. Preliminary version in Proc. 36th IEEE Symposium on Foundations of Computer Science (FOCS), 41-50, 1995.

[8] G. Di Crescenzo, Y. Ishai, and R. Ostrovsky, *Universal Service-Providers for Database private Information Retrieval*, Journal of Cryptology, vol. 14, n. 1, pp. 37-74, 2001. Preliminary version in Proc. of Seventeenth Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 91-100, 1998.

[9] G. Di Crescenzo, T. Malkin, and R. Ostrovsky, *Single Database Private Information Retrieval Implies Oblivious Transfer*, Proc. of Eurocrypt '00, Lecture Notes in Computer Science, vol. 1807, Springer-Verlag, pp. 122-138, 2000.

[10] Y. Gertner, S. Goldwasser, and T. Malkin, *A Random Server Model for Private Information Retrieval*, Proc. of the 2nd RANDOM, 1998.

[11] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, *Protecting Data Privacy in Private Information Retrieval Schemes*, Journal of Computer and Sciences, vol. 60, n. 3, pp. 592-629, 2000. Preliminary version in Proc. of the 30th Annual ACM Symposium on Theory of Computing (STOC), pp. 151-160, 1998.

[12] Y. Ishai and E. Kushilevitz, *Improved Upper Bound on Information-Theoretic Private Information Retrieval*, Proc. of the 31th Annual ACM Symposium on Theory of Computing (STOC), pp. 79-88, 1999.

[13] E. Kushilevitz and R. Ostrovsky, *Replication is not needed: Single Database, Computationally-Private Information Retrieval*, Proc. of the 38th IEEE Symposium on Foundations of Computer Science (FOCS), pp. 364-373, 1997.

[14] E. Kushilevitz and R. Ostrovsky, *One-Way Trapdoor Permutations are Sufficient for Single-database Computationally-Private Information Retrieval*, Proc. of Eurocrypt '00, Lecture Notes in Computer Science, vol. 1807, Springer-Verlag, pp. 104-122, 2000.

[15] M. Naor and B. Pinkas, *Oblivious Transfer and Polynomial Evaluation*, Proc. of the 31st Annual ACM Symposium on Theory of Computing (STOC), pp. 245-254, 1999.

[16] R. Ostrovsky and V. Shoup, *Private Information Storage*, Proc. of the 29th Annual ACM Symposium on Theory of Computing (STOC), pp. 294-303, 1997.