

# Private Information Retrieval

Slides prepared by Panos Kalnis, panos.kalnis@kaust.edu.sa

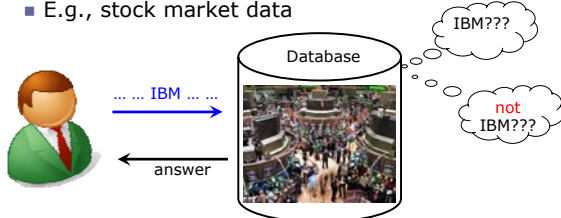
## Outline

- Theoretical PIR
- K-server model
- Covering codes scheme
- Computational PIR

2

## Private Information Retrieval

- The user sends a query to the database
- The database must not be able to infer what the user is after
  - Paradox(?): imagine buying in a store without the seller knowing what you buy.
  - E.g., stock market data



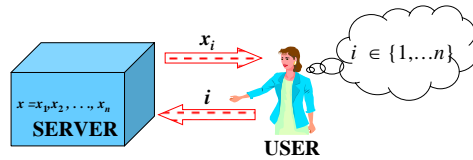
3

## Modeling

- **Server:** holds  $n$ -bit string  $x$   
 $n$  should be thought of as very large
- **User:** wishes
  - to retrieve  $x_i$
  - and
  - to keep  $i$  private

4

## Non-Private Protocol

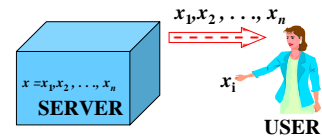


NO privacy!!!

Communication:  $\log n$

5

## Trivial Private Protocol



Server sends entire database  $x$  to User.

Information theoretic privacy.

Communication:  $n$

Is this optimal?

6

## Obstacle

- Theorem [CGKS]:

In any 1-server PIR with information theoretic privacy the communication is at least  $n$ .

[CGKS95] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. Journal of the ACM 45(6), 965-982, 1995.

7

## Naïve Approaches

- Ask for  $m$  different indices (including  $i$ )
  - Reveals a lot of information
- Anonymization techniques
  - Must trust another party (the anonymizer)
  - All-against-one attack

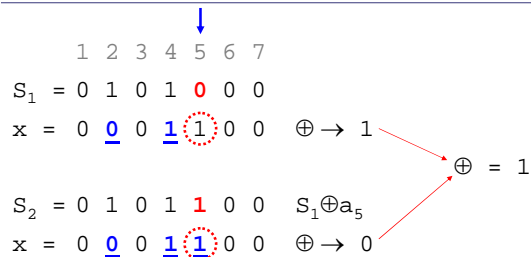
8

## Theoretical PIR

- The attacker has "infinite" computational power
- One server
  - Communication cost  $O(n)$
- $k$  servers
  - Database is replicated
  - Non-collusion
  - Communication cost  $O(k \log k n^{1/(\log k + \log \log k)})$

9

## Naive 2-Server Scheme

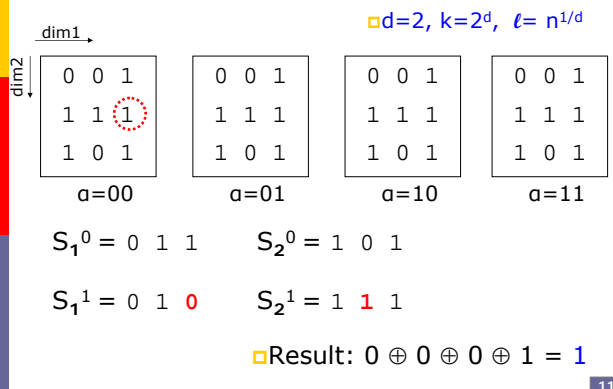


Communication cost:  $2 \cdot n + 2$

- How many bits are sent by each server?

10

## Basic k-Server Scheme (1)



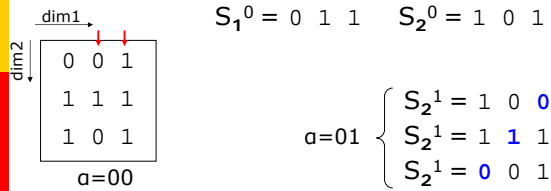
11

## Basic k-Server Scheme (2)

- Generalization to  $k=2^d$  servers:
- Communication cost:  $2^d \cdot (d \cdot n^{1/d} + 1)$
- Example
  - $n=2^{20}=1,048,576$ ,  $d=3$ ,  $k=8$  servers
  - Communication cost:  $8 \cdot (3 \cdot 102 + 1) = 2,456$
- Communication is **not** balanced
  - User to each server:  $d \cdot n^{1/d}$  bits
  - Each server to user: 1 bit

12

## Covering Codes Scheme



- Basic scheme
  - User would send  $2 \cdot (3+3)$  and receive 2 bits
- Covering codes scheme
  - User sends  $3+3$  bits to  $\alpha=00$
  - $\alpha=00$  computes 4 queries and sends 4 bits
  - Etc...

13

## Covering Hamming Codes

$d=3$ , 2 keywords

000 111  
001 011  
010 101  
100 110

No redundancy

$d=4$ , 4 keywords

0000 1111 1000 0111  
0001 0111 0000 1111  
0010 1011 1100 0011  
0100 1101 1010 0101  
1000 1110 1001 0110

4 redundant

14

## Covering Codes and PIR Complexity

| Dimension<br>(i.e., $d$ ) | $2^d$ | # Codewords<br>(# Servers)<br>(i.e., $k$ ) | Volume<br>(Lower)<br>Bound | Total Communication |              |              |              |
|---------------------------|-------|--|----------------------------|---------------------|--------------|--------------|--------------|
|                           |       |  |                            | Asymptotic          | $n = 2^{20}$ | $n = 2^{30}$ | $n = 2^{40}$ |
| 3                         | 8     | 2  | 2                          | $12n^{1/3}$         | 1,224        | 12,300       | 123,864      |
| 4                         | 16    | 4  | 4                          | $28n^{1/4}$         | 924          | 5,096        | 28,700       |
| 5                         | 32    | 7  | 6                          | $60n^{1/5}$         | 1,020        | 3,900        | 15,420       |
| 6                         | 64    | 12   | 10                         | $124n^{1/6}$        | 1,249        | 3,968        | 12,598       |
| 7                         | 128   | 16   | 16                         | $224n^{1/7}$        | 1,792        | 4,480        | 11,872       |
| 8                         | 256   | 32   | 29                         | $480n^{1/8}$        | 2,715        | 6,432        | 15,360       |

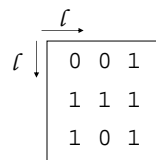
1,048,576

2,456

15

## Computational PIR

- Server computationally **bounded**
- Assume 1-way functions
- Communication complexity  $O(n^\epsilon)$ ,  $\epsilon > 0$



- string  $x$ : a 2-D array
- $l = n^{1/2}$

[KO97] E. Kushilevitz and R. Ostrovsky. Replication is NOT needed: Single database, computationally-private information retrieval. In IEEE Symposium on Foundations of Computer Science, pages 364-373, 1997.

16

## $\mathbb{Z}_N^*$ set

- The user selects two large **prime** numbers:  $q_1$  and  $q_2$
- $N = q_1 \cdot q_2$

$$\mathbb{Z}_N^* = \{x \in \mathbb{Z}_N \mid \text{gcd}(N, x) = 1\}$$

Set of numbers that are prime to N

- E.g.:  $q_1=2, q_2=5, N=10$
- $\mathbb{Z}_{10}^* = \{1, 3, 7, 9, 11, 13, 17, 19, 21, \dots\}$

17

## QR and QNR

Quadratic Residue

$$QR = \{y \in \mathbb{Z}_N^* \mid \exists x \in \mathbb{Z}_N^* : y = x^2 \pmod{N}\}$$

- E.g.:  $x=7 \in \mathbb{Z}_{10}^*$  and  $y=9 \in \mathbb{Z}_{10}^*$   
 $7^2 \pmod{10} = 9$   
 Therefore,  $9 \in QR_{10}$
- **QNR**: Quadratic Non Residue
  - All numbers in  $\mathbb{Z}_N^*$  that are **not** QR

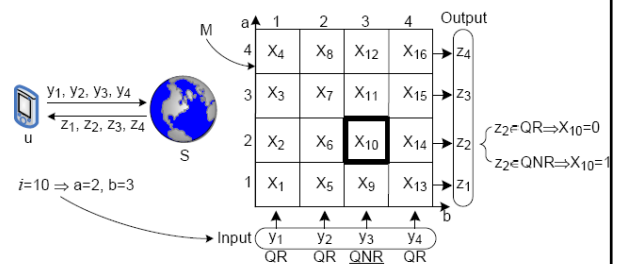
18

## Operations

- $QR \times QR \rightarrow QR$
- $QR \times QNR \rightarrow QNR$
- If  $q_1$  and  $q_2$  are **not** known, it is computationally infeasible to distinguish if a number is QR or QNR

19

## Architecture



20

## PIR Calculations

The **server** calculates for each row:

$$z_r = \prod_{j=1}^t w_{r,j} \quad \begin{array}{l} w_{r,j} = y_j \text{ if } M_{r,j} = 1 \\ \text{else } w_{r,j} = 1 \end{array}$$

The **user** calculates only for row **a**:

$$\left( z_a^{\frac{q_1-1}{2}} = 1 \pmod{q_1} \right) \wedge \left( z_a^{\frac{q_2-1}{2}} = 1 \pmod{q_2} \right)$$

TRUE  
 $z_a$  is QR

FALSE  
 $z_a$  is QNR