

Private Information Retrieval and Connections to Coding Theory

Anna-Lena Horlemann-Trautmann

University of St. Gallen, Switzerland

April 24th, 2017

A little bit about myself:

- born 1983 in Hattingen, Germany
- studied mathematics and philosophy at Ruhr-Universität Bochum, Germany (2002 – 2007)



- Erasmus year in Bergen, Norway (2004 – 2005)



- PhD in mathematics (applied algebra) at University of Zurich (2009 – 2013), Switzerland



- Postdoc in mathematics/electrical engineering at University of Melbourne and Monash University, Australia (2013 – 2015)



- Postdoc in mathematics at University of Zurich (2015)
- Senior postdoc in mathematics/communications at EPF Lausanne, Switzerland (2015 – 2017)



- Assistant professor at University of St. Gallen (currently)



- 1 Privacy
- 2 Coding Theory
- 3 Coding for Private Information Theory
- 4 Computational Privacy
- 5 Conclusions

What is privacy?¹

- *the state of being **free from unwanted** or undue **intrusion** or disturbance in one's private life or affairs; freedom to be let alone*
- *freedom from damaging publicity, public scrutiny, secret surveillance, or unauthorized disclosure of one's personal data or information, as by a government, corporation, or individual*

¹www.dictionary.com/browse/privacy

What is privacy?¹

- *the state of being **free from unwanted** or undue **intrusion** or disturbance in one's private life or affairs; freedom to be let alone*
- *freedom from damaging publicity, public scrutiny, secret surveillance, or unauthorized disclosure of one's personal data or information, as by a government, corporation, or individual*

Privacy aspects

- communicate without being listened to
- store data without anyone reading it
- use search engine, online shopping etc. without anyone knowing what I am interested in

¹www.dictionary.com/browse/privacy

Private information retrieval (PIR)

ask for specific information from a database without letting the server know which information you want

Private information retrieval (PIR)

ask for specific information from a database without letting the server know which information you want

Applications:

- don't want Google (or any other company) to create complete profile of me

Private information retrieval (PIR)

ask for specific information from a database without letting the server know which information you want

Applications:

- don't want Google (or any other company) to create complete profile of me
- be able to watch/read prohibited videos/websites in censorship environment

Private information retrieval (PIR)

ask for specific information from a database without letting the server know which information you want

Applications:

- don't want Google (or any other company) to create complete profile of me
- be able to watch/read prohibited videos/websites in censorship environment
- stock exchange: ask for information about stocks without letting the competitors know what you are interested in

Private information retrieval (PIR)

ask for specific information from a database without letting the server know which information you want

Applications:

- don't want Google (or any other company) to create complete profile of me
- be able to watch/read prohibited videos/websites in censorship environment
- stock exchange: ask for information about stocks without letting the competitors know what you are interested in
- and many more...

Alternative search engines:

- meta search engines that promise not to store your data connected to your IP address (e.g. Tor, ixquick)
- Problem: you have to trust that they do that
- Better solution: have a protocol such that the server never even sees what you searched for
 - ⇒ private information retrieval protocol

First thoughts:

- Assume we have one server that stores the whole database with files $x_1, \dots, x_n$.
- If we want to retrieve file x_1 , but want to hide our wish, we can download several files, e.g. x_1, x_2 and x_n .
- The server then knows that the file we wanted was one of the three downloaded files.
 $\implies$ partial, but not complete privacy!

First thoughts:

- Assume we have one server that stores the whole database with files $x_1, \dots, x_n$.
- If we want to retrieve file x_1 , but want to hide our wish, we can download several files, e.g. x_1, x_2 and x_n .
- The server then knows that the file we wanted was one of the three downloaded files.
 $\implies$ partial, but not complete privacy!

Fact: If we consider only one server, we need to download the complete database for complete privacy.



First thoughts:

- Assume we have one server that stores the whole database with files $x_1, \dots, x_n$.
- If we want to retrieve file x_1 , but want to hide our wish, we can download several files, e.g. x_1, x_2 and x_n .
- The server then knows that the file we wanted was one of the three downloaded files.
 $\implies$ partial, but not complete privacy!

Fact: If we consider only one server, we need to download the complete database for complete privacy.



But luckily: Nowadays almost everything is stored distributed over many servers (clouds, big data)!



Mathematical definition PIR:

- Consider a database with the files $x_1, \dots, x_n$.
- The database is stored over k servers $S_1, \dots, S_k$.
- There are k query functions $Q_1, \dots, Q_k$, one for each server, depending on some random element r .
- There are k answer functions $A_1, \dots, A_k$, one from each server.
- There is one reconstruction function R .

These functions should satisfy

- Correctness: $R(i, r, A_1(x, Q_1(i, r)), \dots, A_k(x, Q_k(i, r))) = x_i$
- Privacy: $Pr(Q_s(i, r) = q) = Pr(Q_s(j, r) = q)$

First simple scheme:

- Assume that $x_1, \dots, x_n \in \mathbb{Z}$, we want to get x_i .
- Replicate the whole database $x_1, \dots, x_n$ and store on two servers S_1, S_2 .
- Queries: $Q_1 \in \{0, 1\}^n$ is random bit string of length n ; Q_2 is equal to Q_1 with the bit flipped at position i .
- Answers: $A_1 = \langle Q_1, (x_1, \dots, x_n) \rangle$; $A_2 = \langle Q_2, (x_1, \dots, x_n) \rangle$
- Reconstruction: if bit i of Q_1 was 0, then

$$R = A_2 - A_1 = x_i$$

otherwise

$$R = A_1 - A_2 = x_i$$

Example:

$$(x_1, x_2, x_3, x_4, x_5) = (3, -1, 12, 4, 5)$$

We want to know the 3rd information, i.e., x_3 . Choose random Q_1 :

$$Q_1 = (1, 0, 0, 1, 0) \implies Q_2 = (1, 0, 1, 1, 0)$$

Example:

$$(x_1, x_2, x_3, x_4, x_5) = (3, -1, 12, 4, 5)$$

We want to know the 3rd information, i.e., x_3 . Choose random Q_1 :

$$Q_1 = (1, 0, 0, 1, 0) \implies Q_2 = (1, 0, 1, 1, 0)$$

$$\begin{aligned} A_1 &= \langle Q_1, (x_1, \dots, x_n) \rangle = \langle (1, 0, 0, 1, 0), (3, -1, 12, 4, 5) \rangle \\ &= 3 + 4 = 7 \end{aligned}$$

$$\begin{aligned} A_2 &= \langle Q_2, (x_1, \dots, x_n) \rangle = \langle (1, 0, 1, 1, 0), (3, -1, 12, 4, 5) \rangle \\ &= 3 + 12 + 4 = 19 \end{aligned}$$

Example:

$$(x_1, x_2, x_3, x_4, x_5) = (3, -1, 12, 4, 5)$$

We want to know the 3rd information, i.e., x_3 . Choose random Q_1 :

$$Q_1 = (1, 0, 0, 1, 0) \implies Q_2 = (1, 0, 1, 1, 0)$$

$$\begin{aligned} A_1 &= \langle Q_1, (x_1, \dots, x_n) \rangle = \langle (1, 0, 0, 1, 0), (3, -1, 12, 4, 5) \rangle \\ &= 3 + 4 = 7 \end{aligned}$$

$$\begin{aligned} A_2 &= \langle Q_2, (x_1, \dots, x_n) \rangle = \langle (1, 0, 1, 1, 0), (3, -1, 12, 4, 5) \rangle \\ &= 3 + 12 + 4 = 19 \end{aligned}$$

Reconstruction:

$$A_2 - A_1 = 19 - 7 = 12 = x_3$$



Quality measures of PIR schemes:

- storage overhead (replication is pretty bad)
- communication rate (how much data is the query for / answer from each server)
- computational complexity of the functions Q_i, A_i, R

Quality measures of PIR schemes:

- storage overhead (replication is pretty bad)
- communication rate (how much data is the query for / answer from each server)
- computational complexity of the functions Q_i, A_i, R

We can improve the previous scheme by

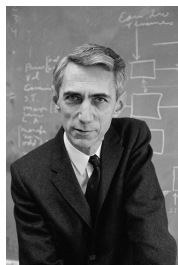
- storing the database as an error correcting code,
- choosing the queries from an error correcting code and
- choosing the reconstruction function as a decoding algorithm.

We keep the idea that the answer functions are scalar products.

- 1 Privacy
- 2 Coding Theory
- 3 Coding for Private Information Theory
- 4 Computational Privacy
- 5 Conclusions

A little bit of history...

Claude Shannon (1916 - 2001),
the *(God-)Father of Information Theory*



- Channel coding (1948):
 - Noisy-channel coding theorem/Shannon capacity (maximum information transfer rate for a given channel and noise level)
- Compression (1948):
 - Source coding theorem (limits to possible data compression)
- Cryptography (1949):
 - One-time pad is the only theoretically unbreakable cipher

Channel Coding

... deals with noisy transmission of information

- over space (communication)
- over time (storage)

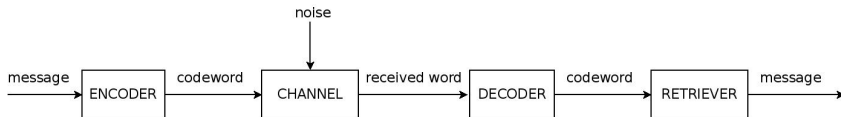
Channel Coding

... deals with noisy transmission of information

- over space (communication)
- over time (storage)

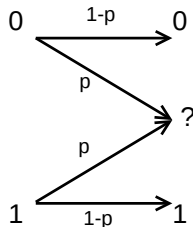
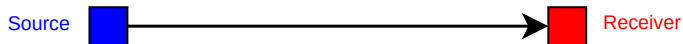
To deal with the noise

- the data is *encoded* with added redundancy,
- the receiver can “filter out” the noise (*decoding*)
- and then *recover* the sent data.



Example (binary erasure channel)

- send 0 or 1
- with probability $p < 1/2$ we receive no symbol



Example (repetition code)

We encode 0 as (000) and 1 as (111). If one or two erasures happen we can uniquely decide which was the sent codeword. If three erasures happen we have no information.

$\implies$ *erasure correction capability* of the code is 2 erasures

Example (repetition code)

We encode 0 as (000) and 1 as (111). If one or two erasures happen we can uniquely decide which was the sent codeword. If three erasures happen we have no information.

$\implies$ *erasure correction capability* of the code is 2 erasures

General repetition code of length n can correct $n - 1$ erasures.

General classical setup (erasure channel):

- codewords are vectors of length n over a finite field $\mathbb{F}_q$
- received word = codeword with erasures
- there should be a unique codeword that can be reconstructed from the received word (otherwise failure)

General definitions:

Definition

A *block code* is a subset $C \subseteq \mathbb{F}_q^n$. The *Hamming distance* of $u, v \in \mathbb{F}_q^n$ is defined as

$$d_H((u_1, \dots, u_n), (v_1, \dots, v_n)) := |\{i \mid u_i \neq v_i\}|.$$

The *minimum (Hamming) distance* of the code is defined as

$$d_H(C) := \min\{d_H(u, v) \mid u, v \in C, u \neq v\}.$$

General definitions:

Definition

A *block code* is a subset $C \subseteq \mathbb{F}_q^n$. The *Hamming distance* of $u, v \in \mathbb{F}_q^n$ is defined as

$$d_H((u_1, \dots, u_n), (v_1, \dots, v_n)) := |\{i \mid u_i \neq v_i\}|.$$

The *minimum (Hamming) distance* of the code is defined as

$$d_H(C) := \min\{d_H(u, v) \mid u, v \in C, u \neq v\}.$$

Theorem

Let C be a code with minimum Hamming distance $d_H(C) = d$. Then for any codeword $c \in C$ and any erasure pattern with at most $d - 1$ erasures, only c can be reconstructed from the received word.

$\implies$ the *erasure correction capability* of C is $d_H(C) - 1$

Typical questions in channel coding theory:

- For a given erasure correction capability, what is the best transmission rate?
 $\implies$ packing problem in $(\mathbb{F}_q^n, d_H)$
- How can one efficiently encode, decode, recover the messages?
 $\implies$ algebraic structure in the code
- What is the trade-off between the two above?

Typical questions in channel coding theory:

- For a given erasure correction capability, what is the best transmission rate?
 $\implies$ packing problem in $(\mathbb{F}_q^n, d_H)$
- How can one efficiently encode, decode, recover the messages?
 $\implies$ algebraic structure in the code
- What is the trade-off between the two above?

Typical tools used in classical setup:

- linear subspaces of $\mathbb{F}_q^n$
- polynomials (and their roots) in $\mathbb{F}_q[x]$
- finite projective geometry

Definition (Reed-Solomon codes)

Let $a_1, \dots, a_n \in \mathbb{F}_q$ be distinct. The code

$$C = \{(f(a_1), f(a_2), \dots, f(a_n)) \mid f \in \mathbb{F}_q[x], \deg f < k\}$$

is called a *Reed-Solomon code* of length n and dimension k . It has minimum distance $n - k + 1$ (optimal).

Definition (Reed-Solomon codes)

Let $a_1, \dots, a_n \in \mathbb{F}_q$ be distinct. The code

$$C = \{(f(a_1), f(a_2), \dots, f(a_n)) \mid f \in \mathbb{F}_q[x], \deg f < k\}$$

is called a *Reed-Solomon code* of length n and dimension k . It has minimum distance $n - k + 1$ (optimal).

A Reed-Solomon code is a linear subspace of $\mathbb{F}_q^n$ of dimension k , it can be represented by a generator matrix

$$G = \begin{pmatrix} 1 & 1 & \dots & 1 \\ a_1 & a_2 & \dots & a_n \\ a_1^2 & a_2^2 & \dots & a_n^2 \\ \vdots & \vdots & & \vdots \\ a_1^{k-1} & a_2^{k-1} & \dots & a_n^{k-1} \end{pmatrix}.$$

Example:

- Consider $\mathbb{F}_3 = \{0, 1, 2\}$, $n = 3, k = 2$ and the evaluation points $a_1 = 0, a_2 = 1, a_3 = 2$.
- Polynomials of degree 0: $0, 1, 2$
- Polynomials of degree 1: $x, x + 1, x + 2, 2x, 2x + 1, 2x + 2$
- Codewords:

$$\begin{array}{rcl}
 (f(0), f(1), f(2)) = & (000) & f(x) = 0 \\
 & (111) & 1 \\
 & (222) & 2 \\
 & (012) & x \\
 & (120) & x + 1 \\
 & (201) & x + 2 \\
 & (021) & 2x \\
 & (102) & 2x + 1 \\
 & (210) & 2x + 2
 \end{array}$$

$$\begin{array}{rcl}
 (f(0), f(1), f(2)) = & (000) & | \quad f(x) = 0 \\
 & (111) & | \quad 1 \\
 & (222) & | \quad 2 \\
 & (012) & | \quad x \\
 & (120) & | \quad x + 1 \\
 & (201) & | \quad x + 2 \\
 & (021) & | \quad 2x \\
 & (102) & | \quad 2x + 1 \\
 & (210) & | \quad 2x + 2
 \end{array}$$

→ any two words differ in $\geq n - k + 1 = 3 - 2 + 1 = 2$ positions

$$\begin{array}{r|l}
 (f(0), f(1), f(2)) = & (000) & f(x) = & 0 \\
 & (111) & & 1 \\
 & (222) & & 2 \\
 & (012) & & x \\
 & (120) & & x + 1 \\
 & (201) & & x + 2 \\
 & (021) & & 2x \\
 & (102) & & 2x + 1 \\
 & (210) & & 2x + 2
 \end{array}$$

→ any two words differ in $\geq n - k + 1 = 3 - 2 + 1 = 2$ positions

we can correct any one erasure, e.g.,

- (2?1) must have been (201)
- (22?) must have been (222)

⇒ decoding = polynomial interpolation

$$\begin{array}{rcl}
 (f(0), f(1), f(2)) = & (000) & \left| \begin{array}{l} f(x) = 0 \\ 1 \\ 2 \\ x \\ x+1 \\ x+2 \\ 2x \\ 2x+1 \\ 2x+2 \end{array} \right. \\
 & (111) & \\
 & (222) & \\
 & (012) & \\
 & (120) & \\
 & (201) & \\
 & (021) & \\
 & (102) & \\
 & (210) &
 \end{array}$$

→ any two words differ in $\geq n - k + 1 = 3 - 2 + 1 = 2$ positions

Generator matrix:

$$G = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

We define the dual code of C as

$$C^\perp = \{v \in \mathbb{F}_q^n \mid \langle v, c \rangle = 0 \ \forall c \in C\}.$$

Lemma

The dual of a Reed-Solomon code

$$C = \{(f(a_1), f(a_2), \dots, f(a_n)) \mid f \in \mathbb{F}_q[x], \deg f < k\}$$

is the Reed-Solomon code

$$C^\perp = \{(f(a_1), f(a_2), \dots, f(a_n)) \mid f \in \mathbb{F}_q[x], \deg f < n - k\}.$$

- 1 Privacy
- 2 Coding Theory
- 3 Coding for Private Information Theory**
- 4 Computational Privacy
- 5 Conclusions

Coded PIR setup:

- the m files $x_1, \dots, x_m$ of the database are stored over servers $S_1, \dots, S_n$ as codewords of k -dimensional Reed-Solomon code $C \subseteq \mathbb{F}_q^n$

- choose m random numbers $d_1, \dots, d_m \in \mathbb{F}_q$

- the query to the first k servers is (if we want x_i)

$$Q_j = (d_1, \dots, d_m) + \underbrace{(0 \dots 0 1 0 \dots 0)}_i ;$$

the queries to the other servers are $Q_j = (d_1, \dots, d_m)$

- the answer of server S_j is $A_j = \langle S_j, Q_j \rangle$
- the vector $(\underbrace{?, \dots, ?}_k, A_{k+1}, \dots, A_n)$ is a codeword of $C^\perp$

with k erasures; we can correct these erasures to $B_1, \dots, B_k$

- $(B_1 - A_1, \dots, B_k - A_k)$ has only one unique preimage with the code C , which corresponds to x_i

Example

Consider $\mathbb{F}_5 = \{0, 1, 2, 3, 4\}$ and the 5 files

$$x_1 = (0, 0), x_2 = (0, 1), x_3 = (1, 0), x_4 = (1, 1), x_5 = (0, 2)$$

which we encode with the Reed-Solomon code

$$G_C = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \end{pmatrix}$$

$$c_1 = x_1 G = (0, 0, 0, 0), c_2 = x_2 G = (0, 1, 2, 3), \dots$$

We store them over $n = 4$ servers

0	0	0	0	c_1
0	1	2	3	c_2
1	1	1	1	c_3
1	2	3	4	c_4
0	2	4	1	c_5
S_1	S_2	S_3	S_4	

0	0	0	0	c_1
0	1	2	3	c_2
1	1	1	1	c_3
1	2	3	4	c_4
0	2	4	1	c_5
S_1	S_2	S_3	S_4	

We choose $m = 5$ random numbers in $\mathbb{F}_5$, e.g. $(1, 2, 3, 4, 0)$. For querying file x_3 :

$$Q_1 = Q_2 = (1, 2, 3, 4, 0) + (0, 0, 1, 0, 0) = (1, 2, 4, 4, 0)$$

$$Q_3 = Q_4 = (1, 2, 3, 4, 0)$$

0	0	0	0	c_1
0	1	2	3	c_2
1	1	1	1	c_3
1	2	3	4	c_4
0	2	4	1	c_5
S_1	S_2	S_3	S_4	

We choose $m = 5$ random numbers in $\mathbb{F}_5$, e.g. $(1, 2, 3, 4, 0)$. For querying file x_3 :

$$Q_1 = Q_2 = (1, 2, 3, 4, 0) + (0, 0, 1, 0, 0) = (1, 2, 4, 4, 0)$$

$$Q_3 = Q_4 = (1, 2, 3, 4, 0)$$

Answers:

$$A_1 = \langle Q_1, S_1 \rangle = \langle (1, 2, 4, 4, 0), (0, 0, 1, 1, 0) \rangle = 8 \equiv 3 \pmod{5}$$

$$A_2 = \langle Q_2, S_2 \rangle = \langle (1, 2, 4, 4, 0), (0, 1, 1, 2, 2) \rangle = 14 \equiv 4 \pmod{5}$$

$$A_3 = \langle Q_3, S_3 \rangle = \langle (1, 2, 3, 4, 0), (0, 2, 1, 3, 4) \rangle = 19 \equiv 4 \pmod{5}$$

$$A_4 = \langle Q_4, S_4 \rangle = \langle (1, 2, 3, 4, 0), (0, 3, 1, 4, 1) \rangle = 25 \equiv 0 \pmod{5}$$

$$\begin{array}{c|c|c|c||c}
 0 & 0 & 0 & 0 & c_1 \\
 0 & 1 & 2 & 3 & c_2 \\
 1 & 1 & 1 & 1 & c_3 \\
 1 & 2 & 3 & 4 & c_4 \\
 0 & 2 & 4 & 1 & c_5 \\
 \hline
 S_1 & S_2 & S_3 & S_4 &
 \end{array} , \quad (A_1, A_2, A_3, A_4) = (3, 4, 4, 0)$$

We decode $(?, ?, 4, 0)$ in the code $C^\perp = C$ by polynomial interpolation ($f(x) = ax + b$):

$$\begin{aligned}
 f(2) = 4 &\iff 2a + b = 4, & f(3) = 0 &\iff 3a + b = 0 \\
 &\implies f(x) = x + 2 \\
 &\implies (?, ?, 4, 0) = (2, 3, 4, 0) =: (B_1, B_2, 4, 0)
 \end{aligned}$$

$$\begin{array}{c|c|c|c||c}
 0 & 0 & 0 & 0 & c_1 \\
 0 & 1 & 2 & 3 & c_2 \\
 1 & 1 & 1 & 1 & c_3 \\
 1 & 2 & 3 & 4 & c_4 \\
 0 & 2 & 4 & 1 & c_5 \\
 \hline
 S_1 & S_2 & S_3 & S_4 &
 \end{array} , \quad (A_1, A_2, A_3, A_4) = (3, 4, 4, 0)$$

We decode $(?, ?, 4, 0)$ in the code $C^\perp = C$ by polynomial interpolation ($f(x) = ax + b$):

$$f(2) = 4 \iff 2a + b = 4, \quad f(3) = 0 \iff 3a + b = 0$$

$$\implies f(x) = x + 2$$

$$\implies (?, ?, 4, 0) = (2, 3, 4, 0) =: (B_1, B_2, 4, 0)$$

There is only one codeword in the database with $(A_1 - B_1, A_2 - B_2) = (1, 1)$ as first two entries, namely c_3 , which corresponds to the file x_3 .



Advantages of coded PIR:

- less storage overhead than with replication on the servers
- or more resilient against server failures than replication
- less symbols need to be downloaded (in the answers)
- can also tolerate some communication between the servers and still keep privacy
- can be generalized such that the query vectors are taken from another RS-code, then even better performance

- 1 Privacy
- 2 Coding Theory
- 3 Coding for Private Information Theory
- 4 Computational Privacy**
- 5 Conclusions

Two options to look at privacy:

- *information theoretic privacy* – queries are uniformly distributed, the server cannot get any information about the user's intent
- *computational privacy* – server cannot get any information about user intent unless it solves a certain computationally (too) hard problem

Two options to look at privacy:

- *information theoretic privacy* – queries are uniformly distributed, the server cannot get any information about the user's intent
- *computational privacy* – server cannot get any information about user intent unless it solves a certain computationally (too) hard problem

Everything presented before assures information theoretic privacy.

We now present an example for computational privacy, based on the *quadratic residuosity assumption*.

Example on one server

- Let m be a positive integer. A number a is called a quadratic residue modulo m if there exists an integer x such that

$$x^2 \equiv a \pmod{m}.$$

- Suppose we have a database with $n = s^2$ bits and store it in a square matrix:

$$\begin{pmatrix} x_{11} & \dots & x_{1s} \\ \vdots & & \vdots \\ x_{s1} & \dots & x_{ss} \end{pmatrix}$$

- Select some large random integers $m_1, \dots, m_r$, compute $m = m_1 \cdot \dots \cdot m_r$ and generate $s - 1$ random quadratic residues $a_1, \dots, a_{s-1}$, as well as one quadratic non-residue b .
- To retrieve x_{ij} send $Q = (a_1, \dots, a_{i-1}, b, a_i, \dots, a_{s-1})$ as query to the server.

- To retrieve x_{ij} send $Q = (a_1, \dots, a_{i-1}, b, a_i, \dots, a_{s-1})$ as query to the server.
- The server answers $A = (\prod_{i:x_{i1}=1} Q_i, \dots, \prod_{i:x_{is}=1} Q_i)$.
- Reconstruction: Check if A_j is a quadratic residue; if yes, then $x_{ij} = 0$ – if no, then $x_{ij} = 1$.
(Follows from the fact that product of residues is a residue and product with a non-residue is a non-residue.)

- To retrieve x_{ij} send $Q = (a_1, \dots, a_{i-1}, b, a_i, \dots, a_{s-1})$ as query to the server.
- The server answers $A = (\prod_{i:x_{i1}=1} Q_i, \dots, \prod_{i:x_{is}=1} Q_i)$.
- Reconstruction: Check if A_j is a quadratic residue; if yes, then $x_{ij} = 0$ – if no, then $x_{ij} = 1$.
(Follows from the fact that product of residues is a residue and product with a non-residue is a non-residue.)

Clue: Since the server does not know the factorization of m , it is very hard to find out which of the elements of Q was the quadratic non-residue. But since we know the factorization it is easy to check if A_j is a quadratic residue or not.

- To retrieve x_{ij} send $Q = (a_1, \dots, a_{i-1}, b, a_i, \dots, a_{s-1})$ as query to the server.
- The server answers $A = (\prod_{i: x_{i1}=1} Q_i, \dots, \prod_{i: x_{is}=1} Q_i)$.
- Reconstruction: Check if A_j is a quadratic residue; if yes, then $x_{ij} = 0$ – if no, then $x_{ij} = 1$.
(Follows from the fact that product of residues is a residue and product with a non-residue is a non-residue.)

Clue: Since the server does not know the factorization of m , it is very hard to find out which of the elements of Q was the quadratic non-residue. But since we know the factorization it is easy to check if A_j is a quadratic residue or not.

Advantage: Here we only need to communicate $\sqrt{n}$ bits, in contrast to information theoretic privacy where we need to communicate the whole database.

- 1 Privacy
- 2 Coding Theory
- 3 Coding for Private Information Theory
- 4 Computational Privacy
- 5 Conclusions

Conclusions

- We presented private information retrieval (PIR) as one of the aspects of privacy.
- We defined a PIR scheme and what privacy means in mathematical terms.
- Complete privacy is not achievable on one server (except for downloading everything).
- A first scheme for several servers is presented by simply replicating the database on each server.
- Performance can be improved by using coding theory, in particular Reed-Solomon codes.

Conclusions

- Performance can be improved by using coding theory, in particular Reed-Solomon codes.
- Therefore we gave an introduction to erasure correcting codes and RS-codes (polynomial evaluation and interpolation).
- Finally we present a coded PIR scheme where the database is stored in coded form with a RS-code.
- Scheme can be generalized by using two RS-codes, one for the database, one for the queries.

Related Work

- *Computational privacy* – server cannot get any information about user intent unless it solves a certain computationally (too) hard problem.
- *Multiround protocols* – one can improve the performance by considering how much information one can privately retrieve over several rounds, instead of just one round.
- *Private information storage* – scheme in which a user is able to read *and write* into the database in a private manner.

Related Work

- *Computational privacy* – server cannot get any information about user intent unless it solves a certain computationally (too) hard problem.
- *Multiround protocols* – one can improve the performance by considering how much information one can privately retrieve over several rounds, instead of just one round.
- *Private information storage* – scheme in which a user is able to read *and write* into the database in a private manner.

Thank you for your attention!
Questions? – Comments?

