PRIVATE INFORMATION RETRIEVAL

#SecureCAT

#FOLLOW UP MEETING

secureCAT

Coding theory and Algebraic Trends for Cryptography, Distributed Data Storage, Machine Learning and Quantum Information

Supported by Grant TED2021-130358B-I00 funded by MCIN/AEI/10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR"

PIRs and codes

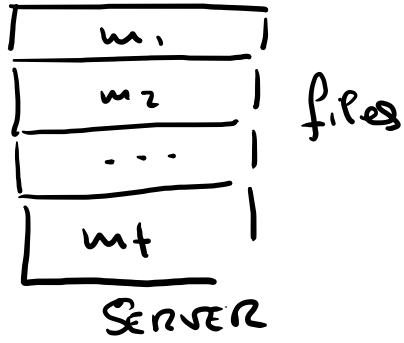
1) GENERAL IDEA OF HOLCANTI et al.
and THE RANK ATTACK.

2) OUR SETTING.

3) Future works.

HOLLANTI ET AL. PROTOCOL (2020)

- C an $[n,k]_{q^s}$ code, G its generator matrix. I an information set of the code. $I \subset [n] = \{1,2,\dots,n\}$

1) 

2) $S \gg 0$ $\{b_1, b_2, \dots, b_S\}$ a basis of $\mathbb{F}_{q^s}$

3) $V = \{b_1 \dots b_v\}$ $v < S$

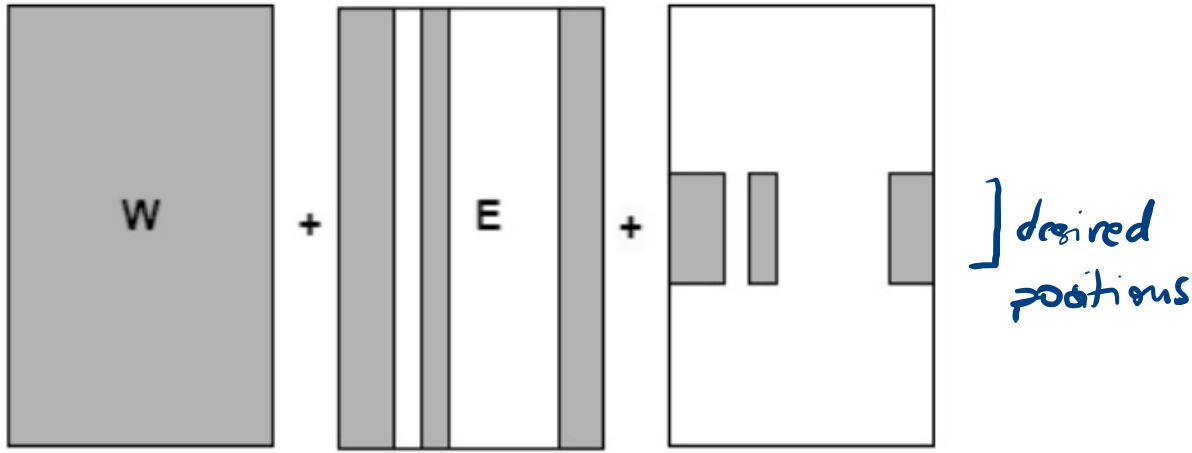
$V^c = \{b_{v+1}, \dots, b_S\}$

SET UP

User.

- Selects a random code $\mathcal{C}$ and W a matrix of t codewords also random.
- For I, the user selects also a matrix E with rows in $\langle V \rangle$ and entries $=0$ in the positions given by I.
- Finally a matrix U with entries in V^c pointing the desired file. [see the graphic]

STRUCTURE OF THE QUERY IN HOLLANTI PROTOCOL



$$Q = W + E + U$$

SERVER : Computes $r = \sum_i^t m_i q_i$

RECOVERING STAGE

$$r - r_I G_I^{-1} G = \sum_{j \in I} m_j i_j$$

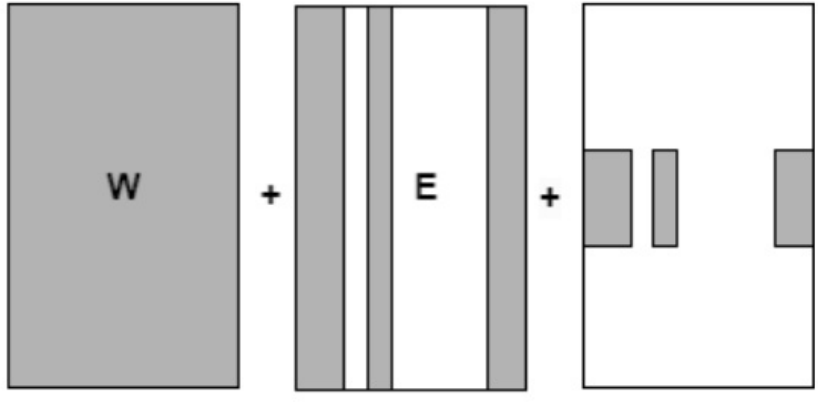
and let $proj_{V^c} : \mathbb{F}_q^s \to V^c$

$$proj_{V^c}\left(r - r_I G_I^{-1} G\right) = \underline{m_d e_j^d}$$

SOLVE.

ATTACK [Bourdage, Lavauzelle, 2020]

Rank difference within the matrix Q.



Our Proposal

arXiv:2311.04688

Mainly use as alphabet $\mathcal{R} = \frac{\mathbb{Z}_m[x]}{\langle x^n - 1 \rangle}$!

With some technical conditions.

Purpose $\longrightarrow$ Avoid S.F. Linear algebra.

THE CODE [over $\mathcal{R}^n$]

Generated by G_{sr} with entries in $\mathcal{R}^n$.

NOTE THAT WHEN WE SEE THOSE ENTRIES IN $\mathbb{Z}_m$ IT IS JUST A **QC-code** OVER $\mathbb{Z}_m$.

Thus, in order to choose the code we will build it as

$$[\tilde{C}_1, \tilde{C}_2, \ldots, \tilde{C}_s]\, M$$

MATRIX PRODUCT CODE

We will also take a $\mathbb{Z}_{2^m}$-code in $\mathcal{R}$ as inner code C_{IN}. [Plays the same role as $\sqrt{}$]

USER.

$$\mathbf{a}^i = \begin{pmatrix} a^i_{11} & a^i_{12} & \dots & a^i_{1s} \\ a^i_{21} & a^i_{22} & \dots & a^i_{2s} \\ \vdots & \vdots & & \vdots \\ a^i_{r1} & a^i_{r2} & \dots & a^i_{rs} \end{pmatrix},$$

where $i \in \{1, \dots, t\}$, $k \in \{1, \dots, r\}$, and $j \in \{1, \dots, s\}$. Then the user encodes $\mathbf{a}^i$ as

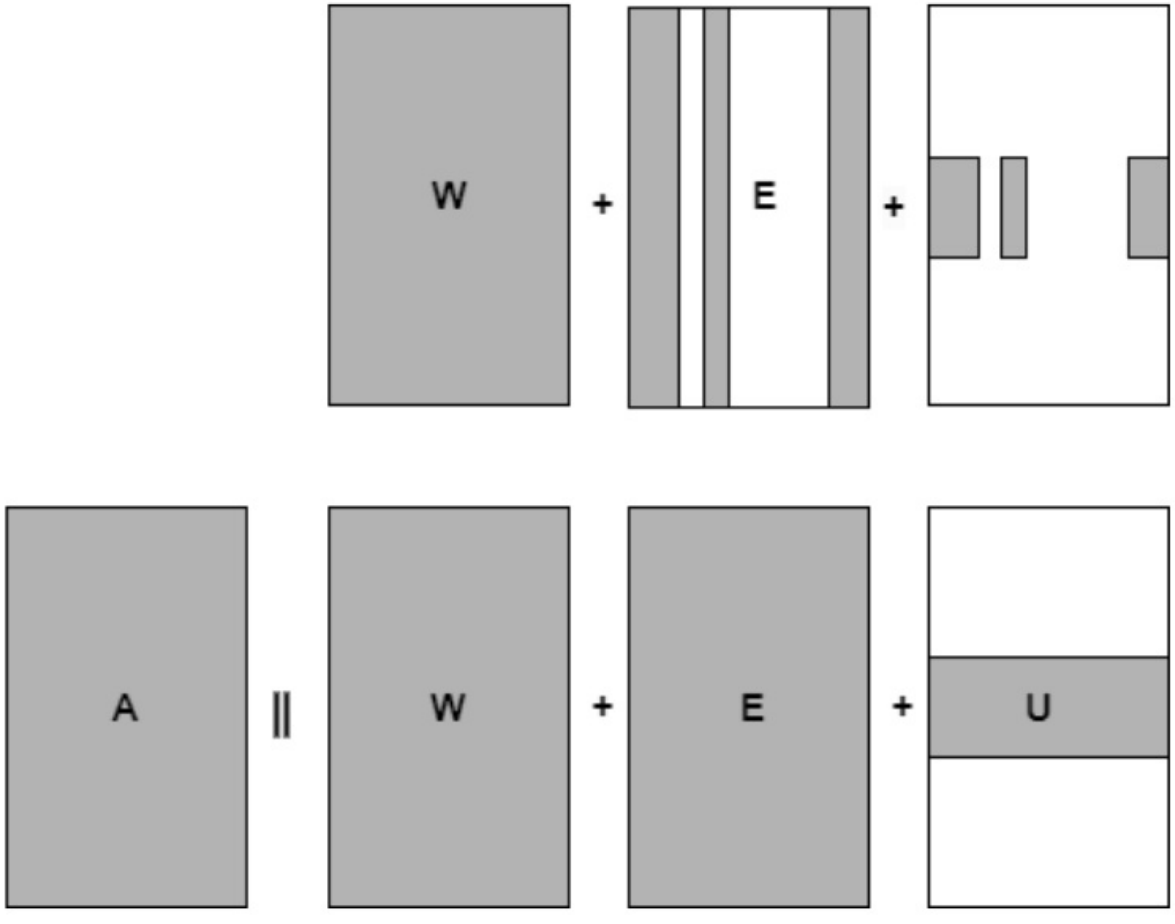
$$\mathbf{w}^i = \mathbf{a}^i \cdot G_{\mathtt{OUT}},$$

$$\boldsymbol{\Delta} = \begin{pmatrix} \boldsymbol{\delta}^1 \\ \boldsymbol{\delta}^2 \\ \vdots \\ \boldsymbol{\delta}^d \\ \vdots \\ \boldsymbol{\delta}^t \end{pmatrix} = \begin{pmatrix} \mathbf{w}^1 + \mathbf{e}^1 + \mathbf{u}^1 \\ \mathbf{w}^2 + \mathbf{e}^2 + \mathbf{u}^2 \\ \vdots \\ \mathbf{w}^d + \mathbf{e}^d + \mathbf{u}^d \\ \vdots \\ \mathbf{w}^t + \mathbf{e}^t + \mathbf{u}^t \end{pmatrix} = \begin{pmatrix} \mathbf{w}^1 + \mathbf{e}^1 \\ \mathbf{w}^2 + \mathbf{e}^2 \\ \vdots \\ \mathbf{w}^d + \mathbf{e}^d + \mathbf{u}^d \\ \vdots \\ \mathbf{w}^t + \mathbf{e}^t \end{pmatrix}, \qquad \mathbf{A} = \begin{pmatrix} \mathbf{a}^1 \\ \mathbf{a}^2 \\ \vdots \\ \mathbf{a}^d \\ \vdots \\ \mathbf{a}^t \end{pmatrix}.$$

entries in C_{IN}

entries in $\widetilde{C}_S \cap (C_{IN}^{\perp} \setminus C_{IN})$

1st condition: Easy if $\widetilde{C}_i$ are nested.


W
+
E
+
A
||
W
+
E
+
U

D. Recovering Stage

We will use a recovering method that resembles the technique used in [11] but without information sets. The user can get the matrix $(\mathbf{R_1} \parallel \mathbf{R_2})$ in Equation (4) from the matrix $\mathbf{R}$ in Equation (3) since the user knows the ring $\mathcal{R}$ and henceforth n. Thus, the user can compute

$$\mathbf{R_2} - (\mathbf{R_1} \cdot G_{\mathtt{OUT}}) = \sum_{i=1}^{t} (\mathbf{DB^i} \cdot \mathbf{w}^i + \mathbf{DB^i} \cdot (\mathbf{e}^i + \mathbf{u}^i) - \mathbf{DB^i} \cdot \mathbf{w}^i) = \sum_{i=1}^{t} (\mathbf{DB^i} \cdot \mathbf{e}^i) + (\mathbf{DB^i} \cdot \mathbf{u^i}). \quad (5)$$

Let us denote by $\Gamma_s(C) = [C, \ldots, C]\mathrm{Id}_s$ the matrix product code of a cyclic code $C \subset \mathcal{R}$, where Id_s is the $s \times s$ the identity matrix and let us denote by $H_{\Gamma_r(C_{\mathrm{IN}}^{\perp})}$ a parity check matrix of the code $\Gamma(C_{\mathrm{IN}})$ over $\mathbb{Z}_m$.

Then the user computes

$$\begin{aligned}
[\mathbf{R_2} - (\mathbf{R_1} \cdot G_{\mathtt{OUT}})] H^{\top}_{\Gamma_r(C_{\mathrm{IN}}^{\perp})} &= \sum_{i=1}^{t} [\mathbf{DB^i} \cdot \mathbf{e}^i + \mathbf{DB^i} \cdot \mathbf{u}^i] H^{\top}_{\Gamma_r(C_{\mathrm{IN}}^{\perp})} \\
&= \sum_{i=1}^{t} \underbrace{[\mathbf{DB^i} \cdot \mathbf{e}^i H^{\top}_{\Gamma_r(C_{\mathrm{IN}}^{\perp})}]}_{= \mathbf{0} \text{ since } \mathbf{e}^i_{kj} \in C_{\mathtt{IN}}} + \underbrace{[\mathbf{DB^i} \cdot \mathbf{u}^i H^{\top}_{\Gamma_r(C_{\mathrm{IN}}^{\perp})}]}_{= \mathbf{0}, \text{ when } t \neq d} \\
&= [\mathbf{DB^d} \cdot \mathbf{u}^d H^{\top}_{\Gamma_r(C_{\mathrm{IN}}^{\perp})}] = \mathbf{M}.
\end{aligned}$$

PROBLEM : INFORMATION SET AGAIN

We have to look the complete matrix.

SOLUTION : OUR INITIAL PURPOSE → NON FREE

ALWAYS WORK ON THE N-F part

AVOID HENSEL LIFTS OF CODES.

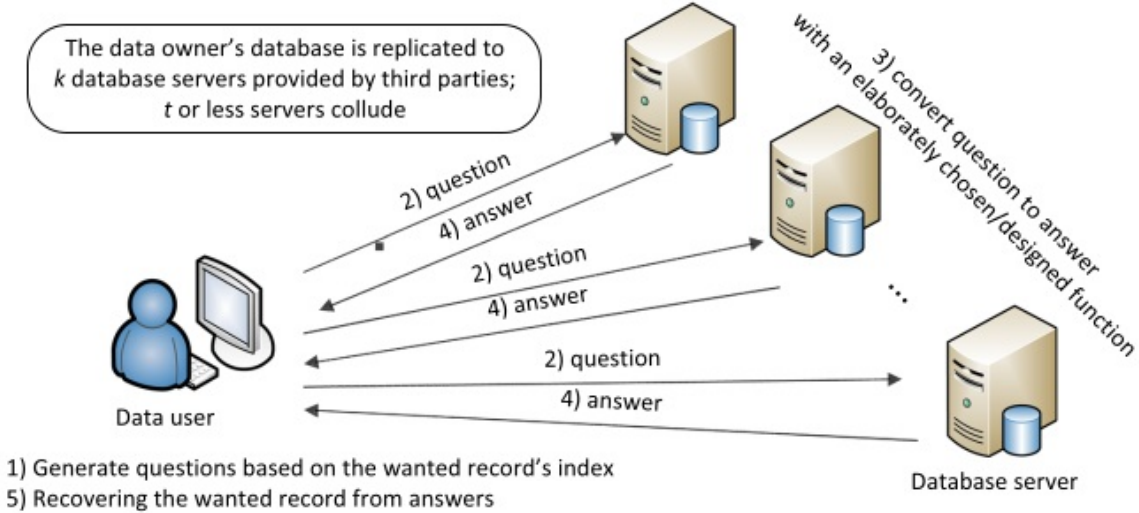
$$\langle f \rangle \subset \frac{\mathbb{Z}_m[x]}{\langle x^n - 1 \rangle}$$

$$\langle af_1 + a^2 f_2 + \dots \rangle \subset \frac{\mathbb{Z}_m[x]}{\langle x^n - 1 \rangle}$$

This provides technical conditions in

- n
- the cyclic codes to be considered.

rPIR: Ramp secret sharing-based PIR.



A secret is represented by a vector $S \in \mathbb{F}_2^h$.

The user breaks it in $t+1$ additive shares

$V[1], V[2], \dots, V[t]$ random vectors in $\mathbb{F}_2^h$

and $V[t+1] = \left(\sum_{i=1}^{t} V[i]\right) + S$

Clearly $\sum_{i=1}^{t+1} V[i] = S$ and any t shares reveal no info on S.

TRIVIAL ADDITIVE RAMP S.S.

Secret is divided in some blocks, each one a vector.

$$\{s_i : i = 1, 2, \ldots, u\} \quad s_i \in \mathbb{F}_2^h$$

$V[1]$ random

$V[2] = V[1] + s_1$

$\vdots$

$V[u+1] = V[1] + \sum_{i}^{u} s_i$

$s_1 = V[1] + V[2]$

$s_2 = V[2] + V[3]$

$\vdots$

$s_u = V[u] + V[u+1]$

t-private $(t+1)$ server single query PIR

There are $k = t+1$ servers each one with a replica of the DB.

$DB = x[1], \ldots, x[n]$. $\quad x[i] \in \mathbb{F}_2^l$

If $Y \in \mathbb{F}_2^n$ one defines

$$P(Y) = \sum_{j=1}^{n} x[j] \cdot Y_j$$

Using the trivial additive s.s. before we have the following protocol.

I) User breaks the input secret e_i, s.t $\sum_{j=1}^{k} V[j] = e_i$

II) Sends $V[j]$ to the j-th server. Question length $= n$

III) The j-th server computes $P(V[j])$

Answer length is ℓ

(IV) User reconstruction

$\longrightarrow$

$$X[i] = P(e_i)$$

$$= P\left(\sum_{V[j] \in S_i} V[j]\right) = \sum_{h=1}^{u} X[h] \times \left(\sum V[j]\right)_h$$

$$= \sum \left(\sum_{h=1}^{n} X[h] \times V[j]_h\right)$$

$$= \sum P(V_j)$$

.

S_i is the set of additive shares of e_i

OTHER CHOICES: Threshold

	Piece 1	Piece 2
Share 1	$r_1 + r_2 + s_1$	$r_3 + r_4 + s_2$
Share 2	$r_1 + r_4$	$r_3 + r_1$
Share 3	$r_1 + s_2$	$r_2 + r_3$

$s_1, s_2, \underbrace{r_1, r_2, r_3, r_4}_{\text{random}} \in \mathbb{F}_2^h$

$$s_1 = sp_{1,2} + sp_{2,1} + sp_{2,2} + sp_{3,1}$$

$$s_2 = sp_{1,1} + sp_{2,2} + sp_{3,1} + sp_{3,2}$$

7 Concluding remarks

In this paper, a new family of information-theoretic PIR based on ramp secret sharing, rPIR, has been studied. Four rPIR schemes have been proposed, and three ramp secret sharing schemes are adopted in these rPIR designs. rPIR's usages has been demonstrated in outsourced data sharing and P2P content delivery scenarios. The performance of rPIR schemes has been evaluated by theoretic analysis and experiments under different numbers of available servers and privacy requirements. The results have shown that rPIR schemes can achieve

Int. J. Inf. Secur. (2017) 16:603–625
DOI 10.1007/s10207-016-0347-8

REGULAR CONTRIBUTION

rPIR: ramp secret sharing-based communication-efficient private information retrieval

Batch Codes [Holmann' 21]

A *t-batch code* is a method to store a data record in encoded form on multiple servers in such a way that the bit-values in any batch of t positions from the record can be retrieved by decoding the bit-values in t disjoint groups of positions. Batch codes were initially introduced in [3] as a method to improve load-balancing in distributed data storage systems. Later, so-called *switch codes*, a special type of batch codes, were proposed in [8] as a method to increase the throughput rate in network switches.

Ex. 2-batch code for 3 servers

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

Smarter than replication on 4 servers.

PIR & BATCH codes are characterized by the property that given some encoded data, certain type of simultaneous request for specific data symbols can each been handled by reading and decoding data from a set of positions (Recovery set), where there is limited overlap of this recovery windows.

Research Questions

- Find lower bounds on $P_q(k,t)$ and $B_q(k,t)$, the shortest length of a q-ary t-PIR or t-Batch

  ?? V. Skachek, Batch and PIR codes and their relation to LRC. (2018)

- Find interesting non-linear PIR/BATCH codes

- The strength of a code as a PIR/BATCH code is largest t for which it is t-PIR or t-BATCH

  Can we compute or bound it?

-
-
-