Explicit Constructions of Weakly Secure Regenerating Codes for Distributed Storage

Swanand Kadhe
Advisor: Prof. Alex Sprintson

Texas A&M University

SPCodingSchool
January 19th to 30th 2015 - Campinas, Brazil
SP Coding and Information School

Weakly Secure Distributed Storage

- Eve should not learn about *any* small group of files
  - No loss in storage capacity ! ☺
- Example: Suppose there are 4 files $\{S_1, S_2, S_3, S_4\} \in \mathbb{F}_{13}^4$

$$S_1 + S_2 + S_3$$
$$S_1 + S_2$$

  - Eve can decode third file

$$S_1 + S_2 + S_3 + S_4$$
$$S_1 + 5S_2 + 12S_3 + 8S_4$$

  - Eve cannot infer anything about any group of two files

$$I\left(S_{\mathcal{G}'}; E\right) = 0 \quad \forall \mathcal{G}' \ : \ |\mathcal{G}'| \leq g + 1$$

- Equivalent to maximizing the minimum Hamming weight of any vector in the span of eavesdropped codewords

[1] Bhattad and Narayanan [2005]

Weakly Secure Regenerating Codes

- Regenerating Codes: Minimize the repair bandwidth by trading off storage space – focus on exact codes

- Generator matrix of a regenerating code is typically sparse
- *e.g.*, generator matrix of a Product-Matrix framework based regenerating code (MBR point)

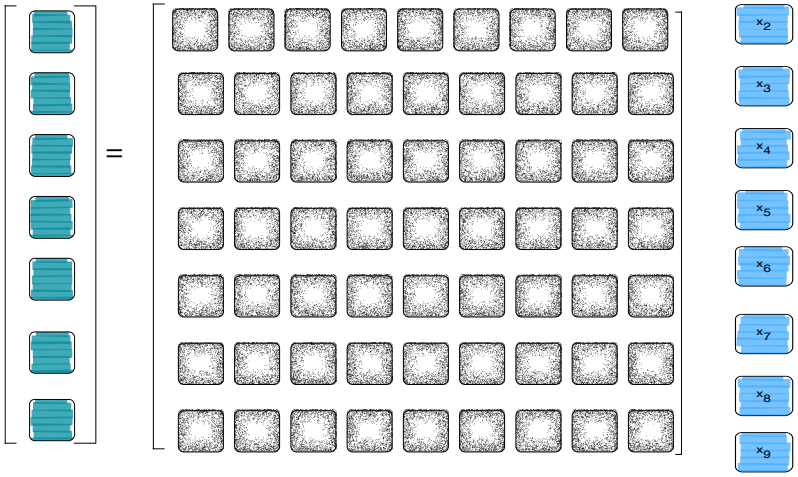
$$G_e = \begin{bmatrix} \Psi(e,1) & \Psi(e,2) & \Psi(e,3) & \Psi(e,4) & 0 & 0 & 0 & 0 & 0 \\ 0 & \Psi(e,1) & 0 & 0 & \Psi(e,2) & \Psi(e,3) & \Psi(e,4) & 0 & 0 \\ 0 & 0 & \Psi(e,1) & 0 & 0 & \Psi(e,2) & 0 & \Psi(e,3) & \Psi(e,4) \\ 0 & 0 & 0 & \Psi(e,1) & 0 & 0 & \Psi(e,2) & 0 & \Psi(e,3) \end{bmatrix}$$

$(n = 5, k = 3, d = 4, \alpha = 4, \beta = 1)$-Regenerating Code

- Not weakly secure against 2 guesses – outer code can help !

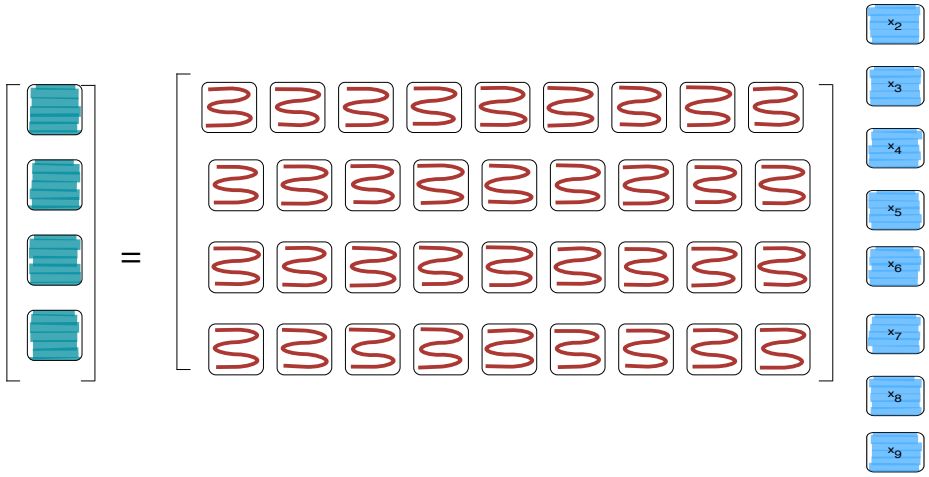
[2] Rashmi, Shah, Kumar, and Ramachandran [2013]

Outer Coset Code for Weak Security



S H X

Coset Code



E G X

Regenerating Code

Can we use elegant structure of Product-Matrix codes to explicitly design H such that small field size is sufficient ?

Explicit Outer Code Design

- Design *H* such that it has the same *structure* as *G*

$$H = \begin{bmatrix} 0 & \hat{\Psi}(1,1) & 0 & 0 & \hat{\Psi}(1,2) & \hat{\Psi}(1,3) & \hat{\Psi}(1,4) & 0 & 0 \\ 0 & \hat{\Psi}(2,1) & 0 & 0 & \hat{\Psi}(2,2) & \hat{\Psi}(2,3) & \hat{\Psi}(2,4) & 0 & 0 \\ 0 & \hat{\Psi}(3,1) & 0 & 0 & \hat{\Psi}(3,2) & \hat{\Psi}(3,3) & \hat{\Psi}(3,4) & 0 & 0 \\ 0 & 0 & \hat{\Psi}(1,1) & 0 & 0 & \hat{\Psi}(1,2) & 0 & \hat{\Psi}(1,3) & \hat{\Psi}(1,4) \\ 0 & 0 & \hat{\Psi}(2,1) & 0 & 0 & \hat{\Psi}(2,2) & 0 & \hat{\Psi}(2,3) & \hat{\Psi}(2,4) \\ 0 & 0 & \hat{\Psi}(3,1) & 0 & 0 & \hat{\Psi}(3,2) & 0 & \hat{\Psi}(3,3) & \hat{\Psi}(3,4) \\ 0 & 0 & 0 & \hat{\Psi}(1,1) & 0 & 0 & \hat{\Psi}(1,2) & 0 & \hat{\Psi}(1,3) \end{bmatrix}$$

$(n = 5, k = 3, d = 4, \alpha = 4, \beta = 1)$-Regenerating Code

- Features:
  - Fixed one or two units loss in capacity
    - Almost achieves the insecure storage capacity
  - Twofold enhancement over uncoded security level
    - ✓ $g_{max} \approx 2g_{uncoded}$

[NetCod 14: MBR case, Allerton 14: MSR case]
Longer version: arXiv:1405:2894