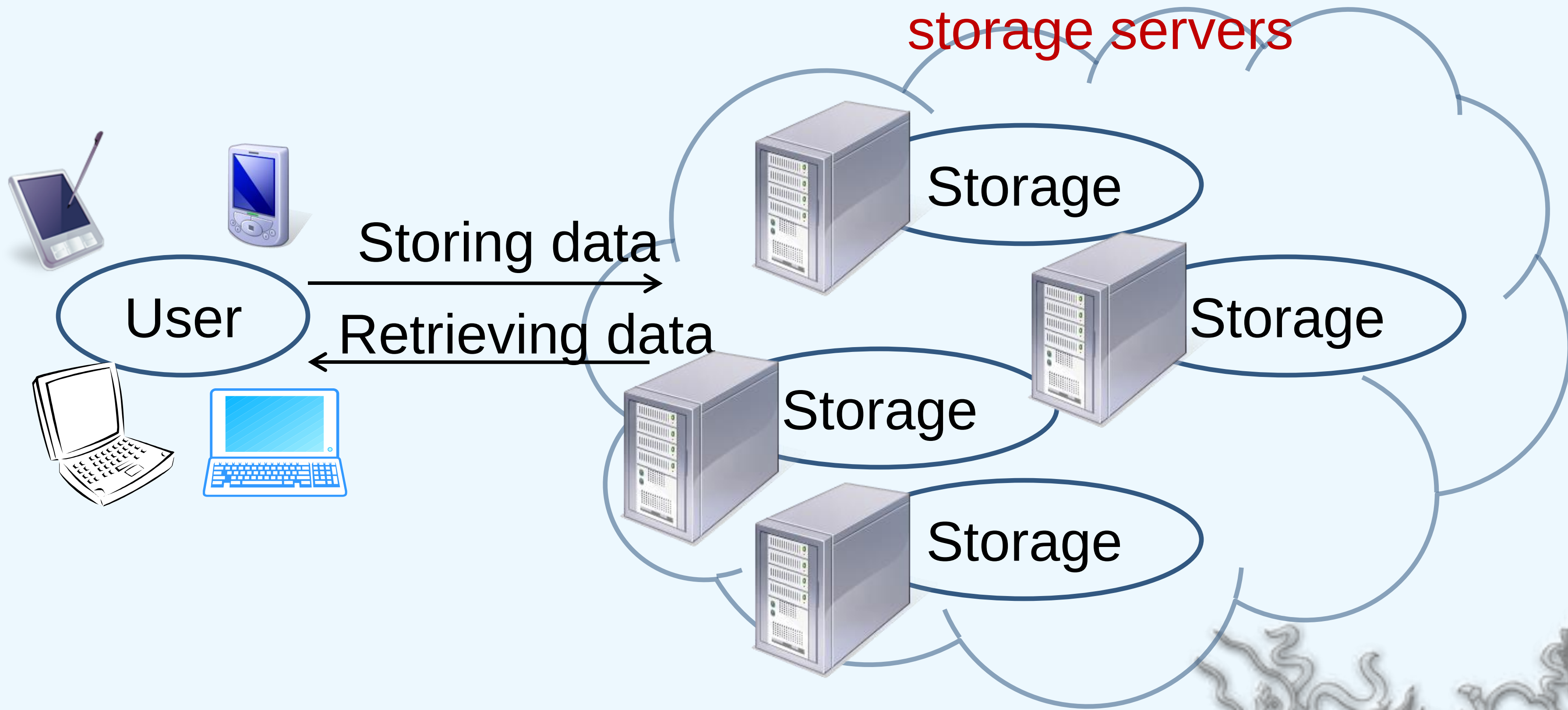
NSC-JST workshop

Secure Decentralized Erasure Code based Networked Storage Systems with Multiple Functionalities

Wen-Guey Tzeng 曾文貴
(joint work with Hsiao-Ying Lin)
National Chiao Tung University

Distributed Networked Storage



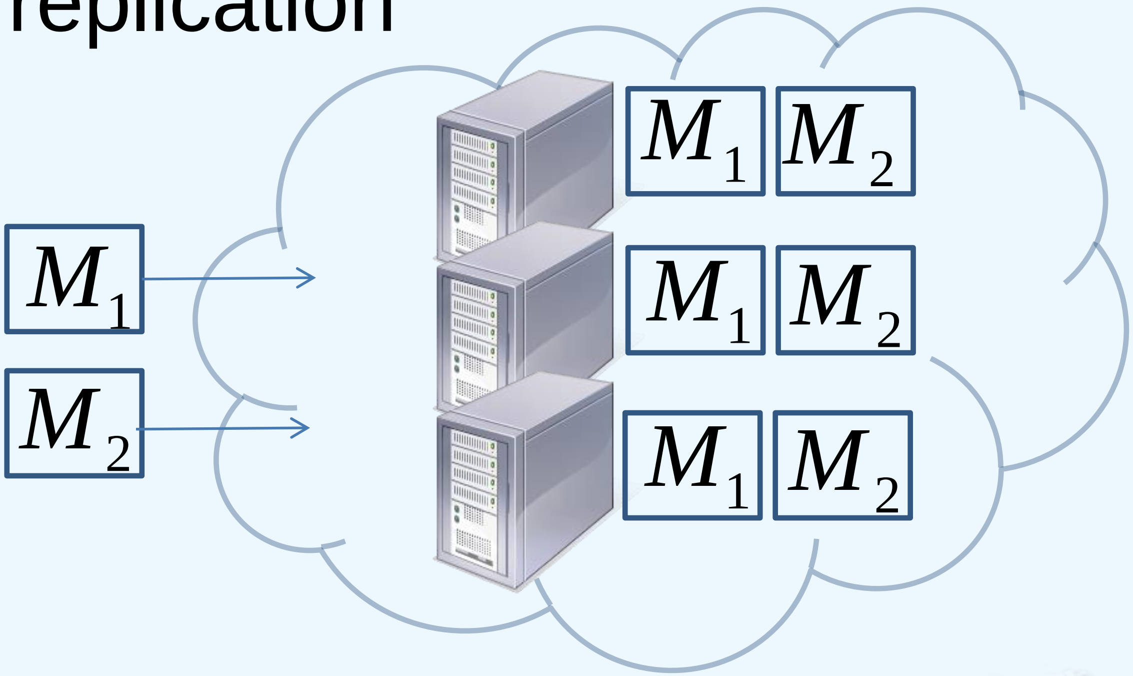
Servers are decentralized:
no single central authority

Objectives

- Data robustness
  - Storage servers may fail over time (erasure error)
- Data confidentiality
  - The mangers of storage servers may not be honest
- Functionalities
  - Data forwarding
  - System repairing
  - Integrity check
  - Keyword search
  - …

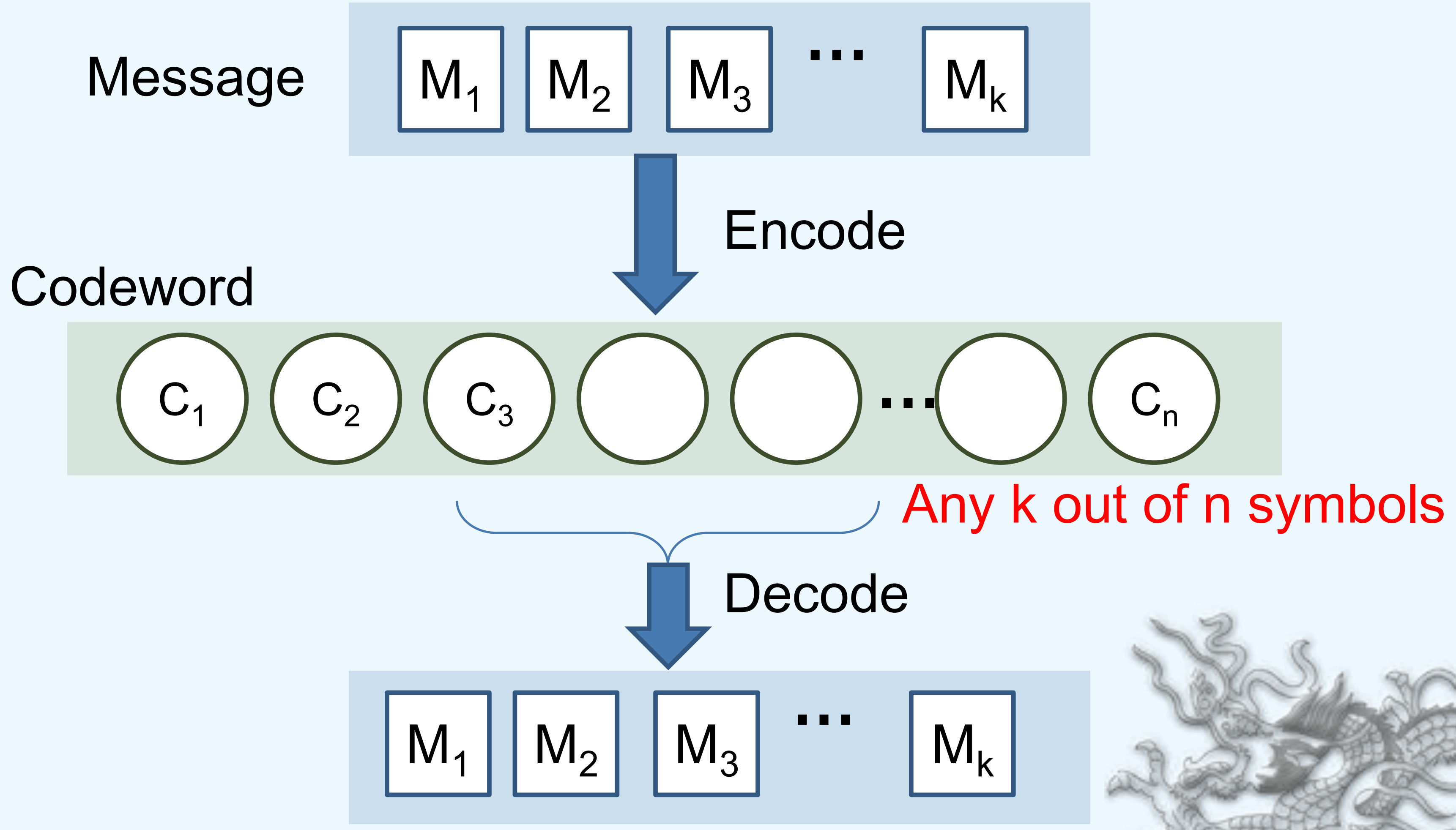
Data Robustness

- Simple replication



  - Expensive in storage space
  - Solution: <u>decentralized erasure code</u>

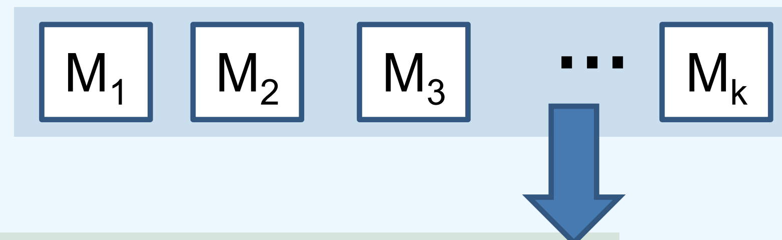
Erasure Code



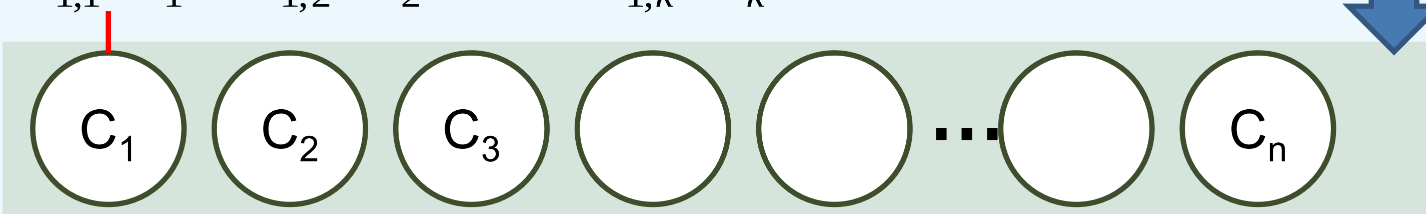
Decentralized Erasure Code

Decentralized encoding

- each codeword symbol is independently computed
- linear combination with random coefficients



$$C_1 = a_{1,1}M_1 + a_{1,2}M_2 + \ldots + a_{1,k}M_k$$



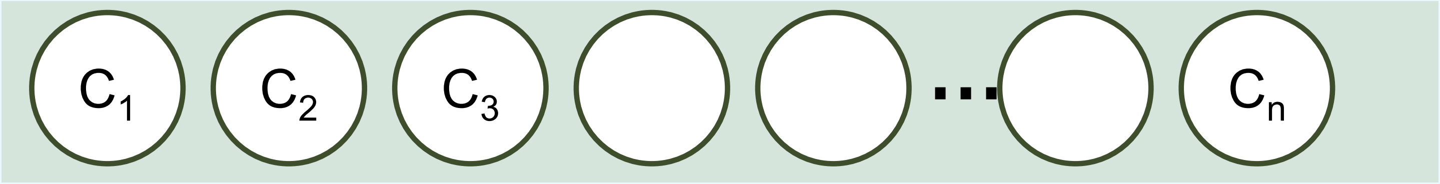
$$G = \begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,k} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,k} \\ a_{3,1} & a_{3,2} & \ldots & a_{3,k} \\ \ldots & \ldots & \ldots & \ldots \\ a_{n,1} & a_{n,2} & \ldots & a_{n,k} \end{bmatrix}$$

$$[M_1 \quad M_2 \quad \ldots \quad M_k] \bullet G^T = [C_1 \quad C_2 \quad \ldots \quad C_n]$$

Decentralized Erasure Code

Decode

- solve a linear system with k equations and k unknowns

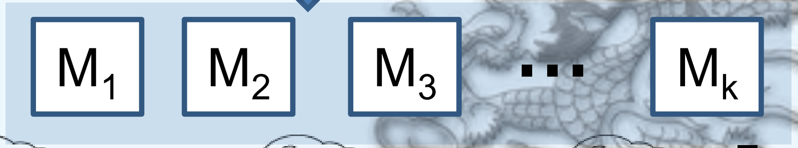


$$G = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,k} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,k} \\ a_{3,1} & a_{3,2} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,k} \end{bmatrix} \rightarrow K = \begin{bmatrix} & \\ & \end{bmatrix}$$

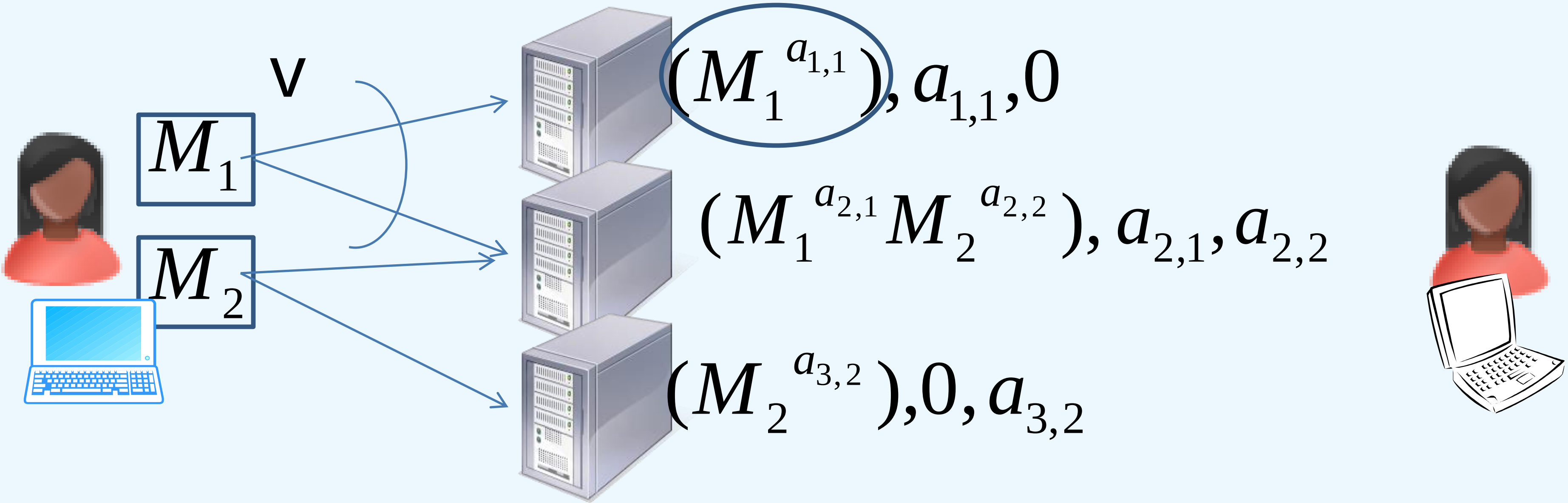
K: k columns, k rows

When K is invertible

$$\begin{bmatrix} M_1 \\ M_2 \\ \cdots \\ M_k \end{bmatrix}^T = \begin{bmatrix} C_1 \\ C_2 \\ \cdots \\ C_k \end{bmatrix}^T (K^T)^{-1}$$



Example



$$[M_1 \quad M_2] \circ \begin{bmatrix} a_{1,1} & a_{2,1} & 0 \\ 0 & a_{2,2} & a_{3,2} \end{bmatrix}$$

$$= [M_1^{a_{1,1}} \quad M_1^{a_{2,1}} M_2^{a_{2,2}} \quad M_2^{a_{3,2}}] = [c_1 \quad c_2 \quad c_3]$$

科学技術振興機構 Japan Science and Technology Agency

行政院國家科學委員會 National Science Council

Decentralized Erasure Code for Storage

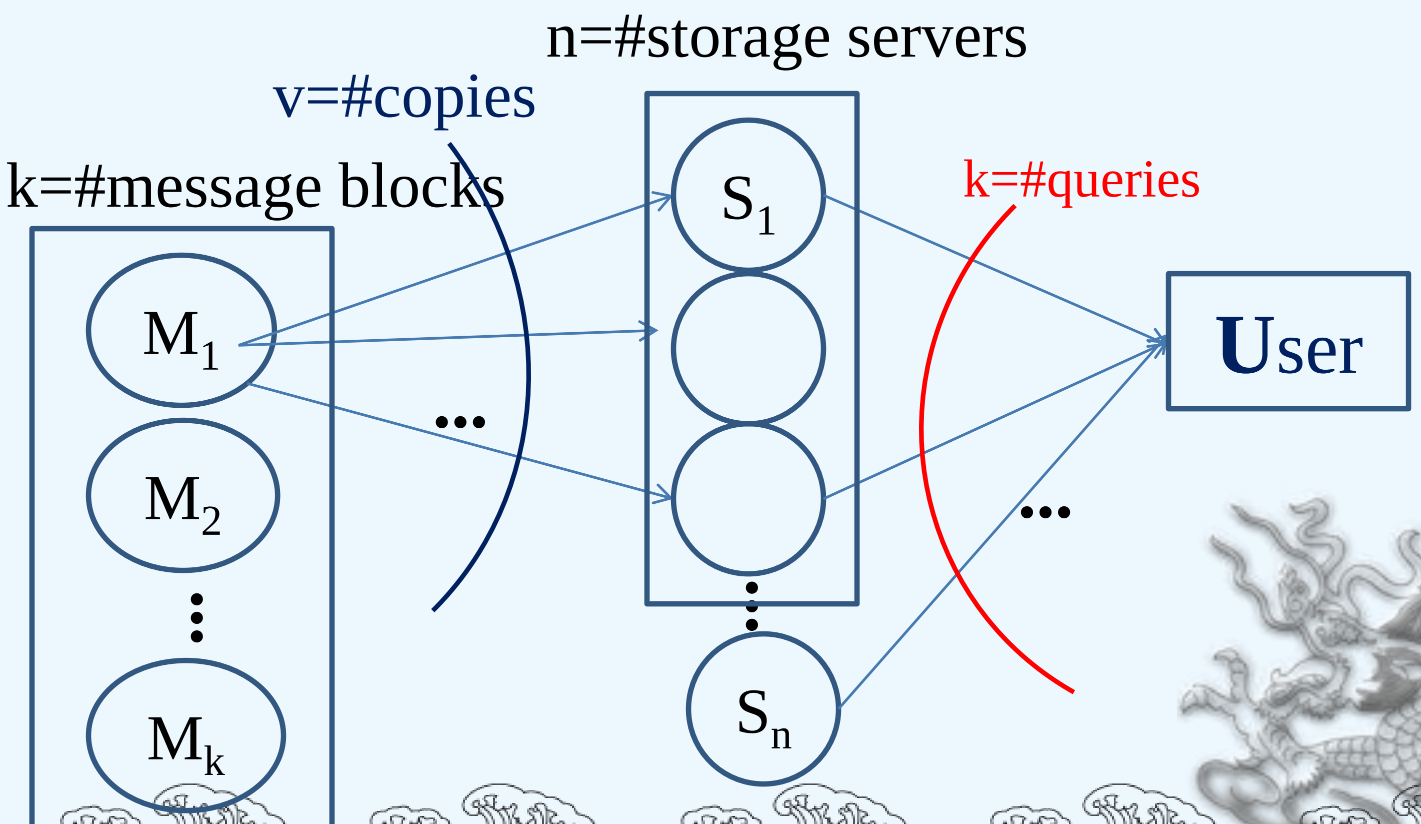
- Decentralized encoding (in servers)
- Robust against erasure errors
- Efficient in storage
- Light confidentiality

Previous Result

Assumption: random & independent distribution,

[2006] Result: **v=c ln(k), where c > 5 n/k**

$$\Pr[\text{success retrieval}] > 1 - k/p - o(1)$$



Decentralized Erasure Code for Storage

- Robust against erasure errors
- Decentralized encoding
- Efficient in storage
- Light confidentiality
- Stronger confidentiality is wanted
- More functionalities are desired
  - Data forwarding
  - System repairing
  - ...

Security Concerns

- Storage in public
  - Confidentiality of stored data
  - Solution: cryptographic encryption scheme
- Key management (key server)
  - Store secret key at single point is risky
  - Solution: key servers in private cloud
    - secret sharing
    - key share holder performs partial decryption
    - user recovers messages from partial decrypted data

Our work

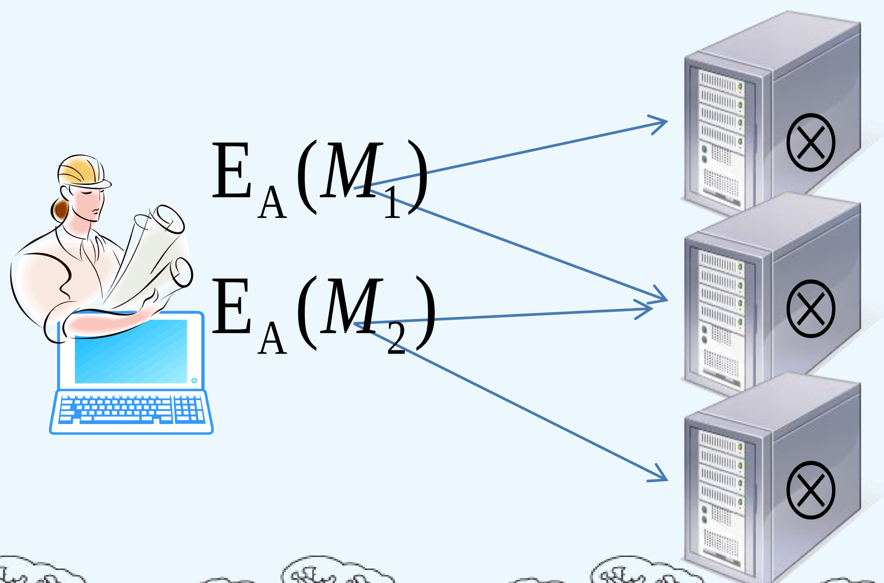
- No central authority (decentralized)
- Data robustness — Decentralized random linear code
- Strong confidentiality — Homomorphic public-key encryption
- Secure data forwarding — Proxy re-encryption
- Key management — Partial decryption
- Repair mechanism — Linear code repair

Erasure coding over ciphertext

- Homomorphic encryption (multiplicative)

$$\mathrm{E}_{\mathrm{A}}(M_1) \otimes \mathrm{E}_{\mathrm{A}}(M_2) = \mathrm{E}_{\mathrm{A}}(M_1 M_2)$$

$$\mathrm{E}_{\mathrm{A}}(M_1)^{\mathrm{a}} \otimes \mathrm{E}_{\mathrm{A}}(M_2)^{\mathrm{b}} = \mathrm{E}_{\mathrm{A}}(M_1{}^{a} M_2{}^{\mathrm{b}})$$



$$\mathrm{E}_{\mathrm{A}}(M_1{}^{a_{1,1}}), a_{1,1}, 0$$

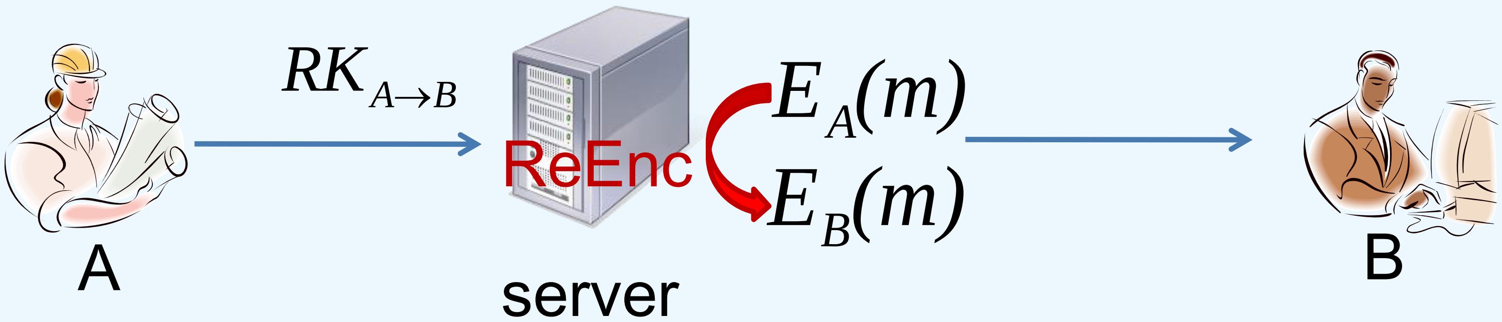
$$\mathrm{E}_{\mathrm{A}}(M_1{}^{a_{2,1}} M_2{}^{a_{2,2}}), a_{2,1}, a_{2,2}$$

$$\mathrm{E}_{\mathrm{A}}(M_2{}^{a_{3,2}}), 0, a_{3,2}$$

科学技術振興機構
Japan Science and Technology Agency

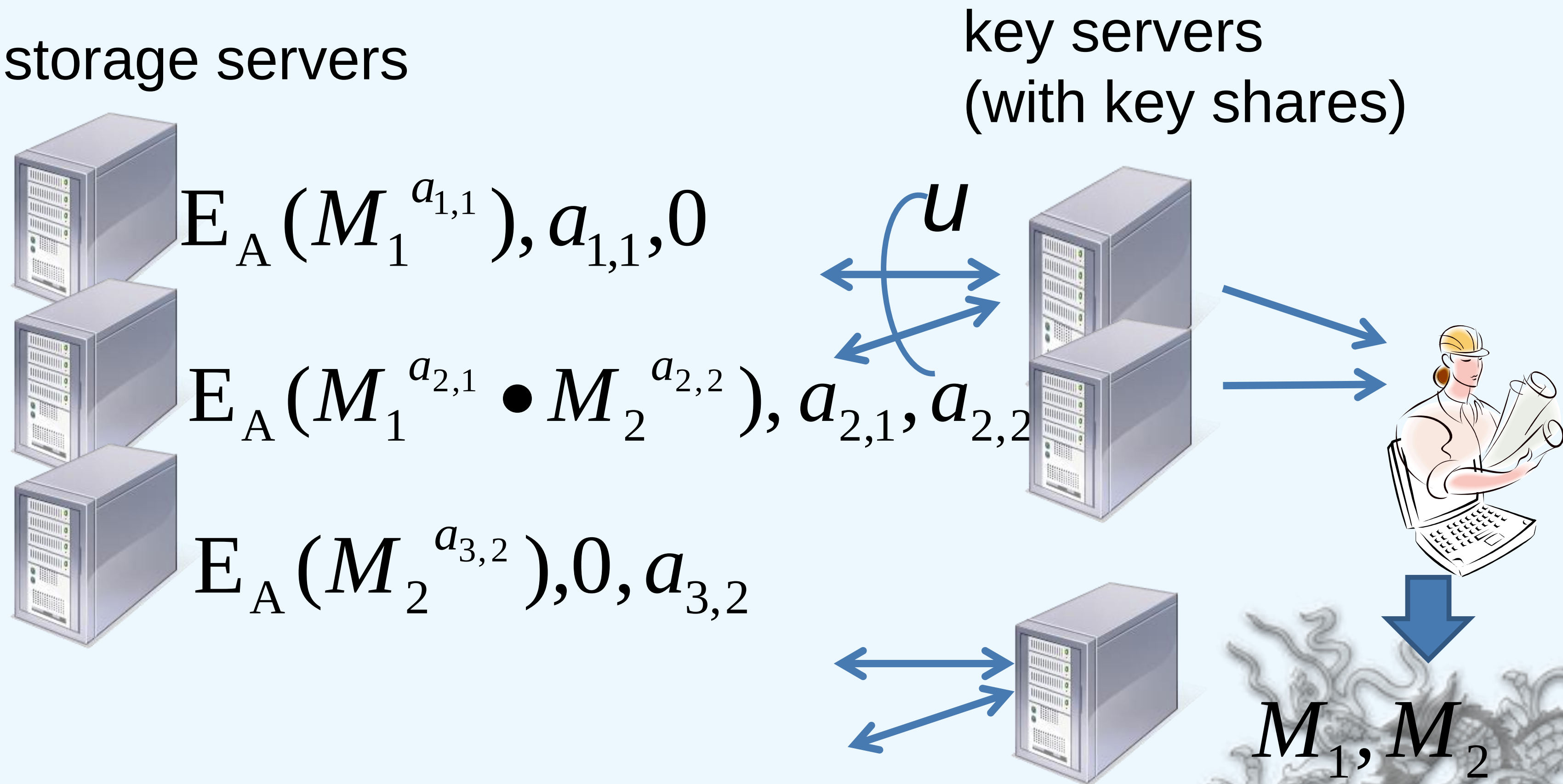
行政院國家科學委員會
National Science Council

Proxy Re-encryption



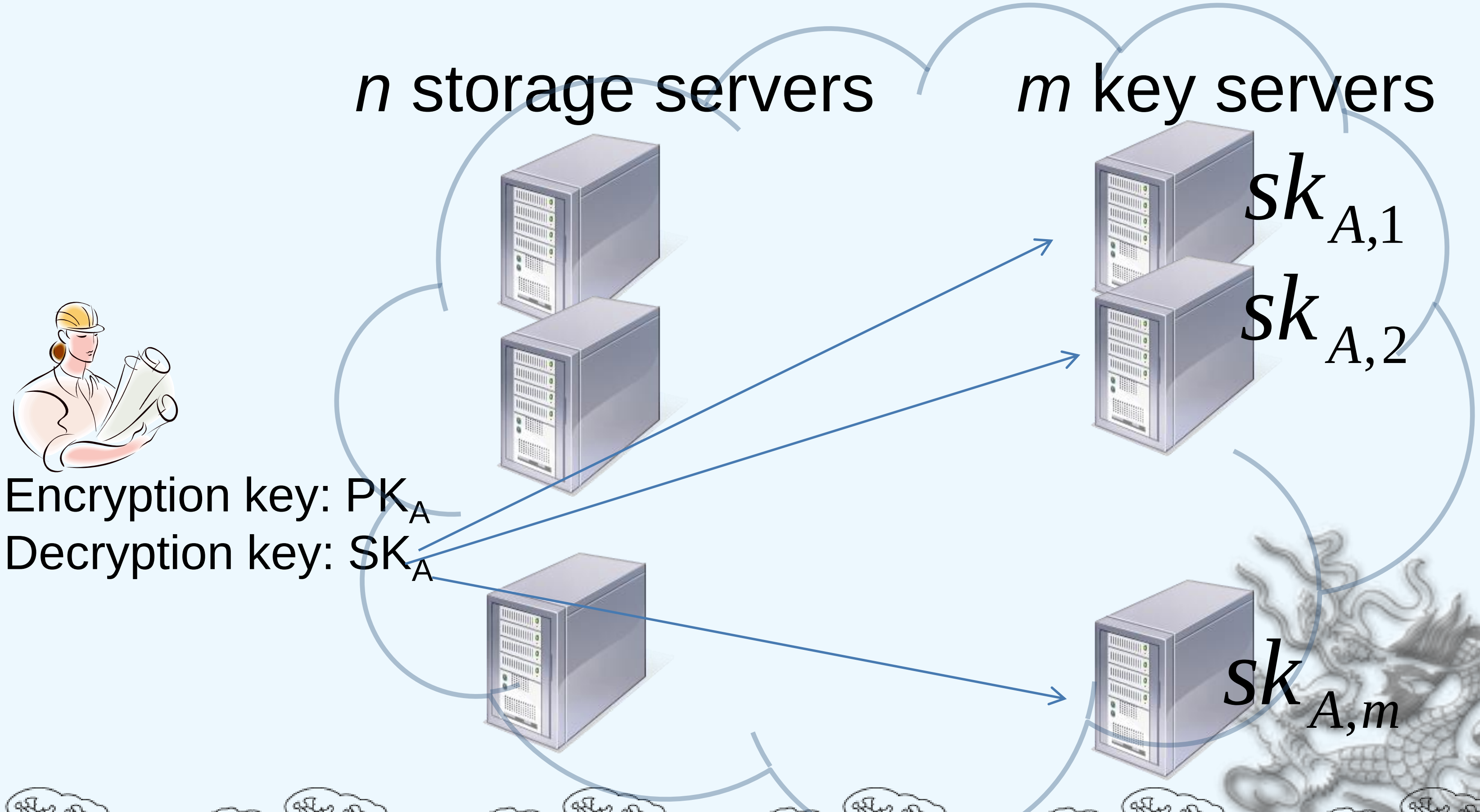
- Required properties
  - Support decentralized/secure erasure coding
  - Support decentralized partial decryption
- We construct one satisfying all. It is
  - Based on bilinear map
  - Multiplicatively homomorphic

Decentralized Partial Decryption



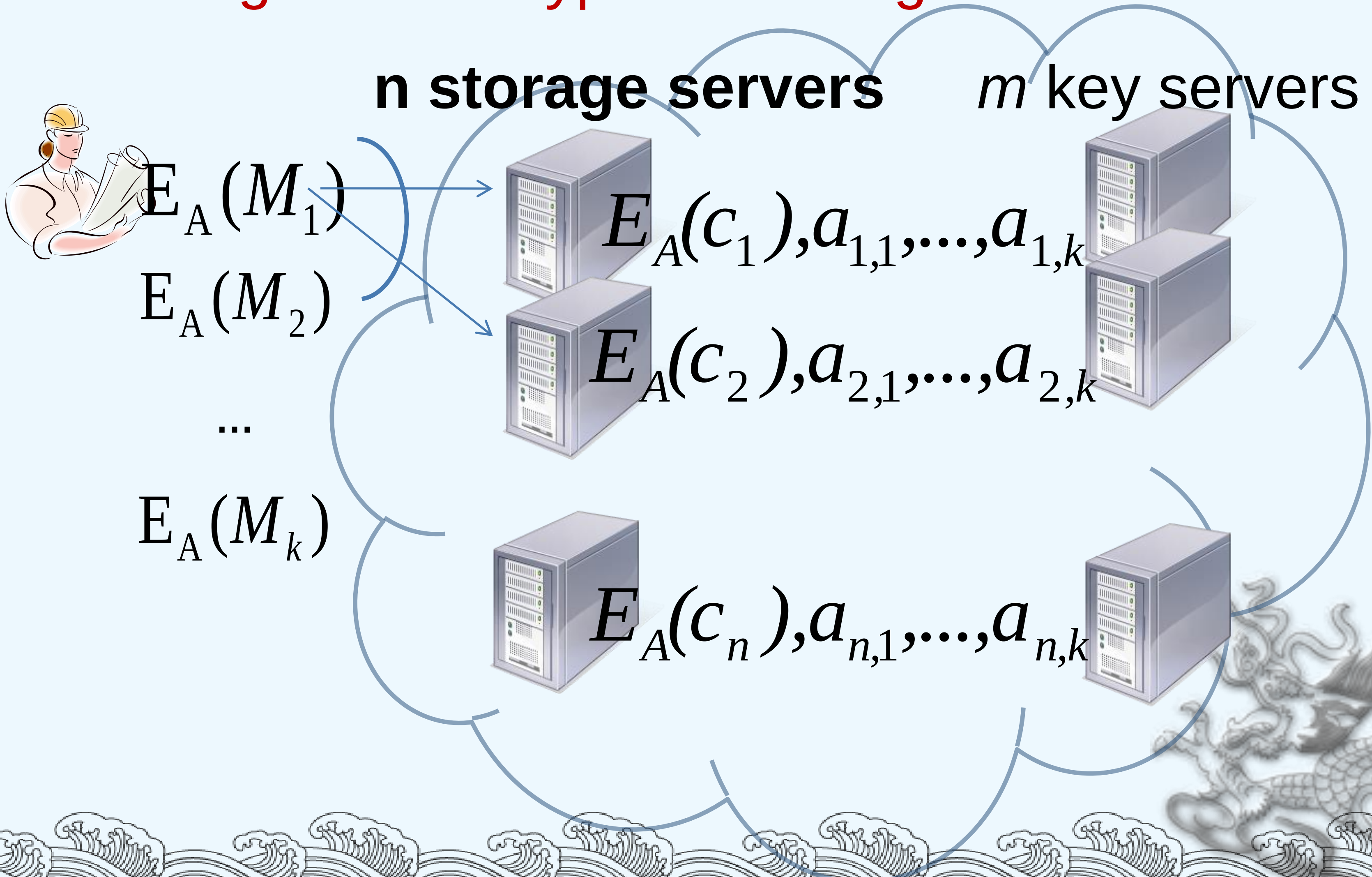
System Overview

t-out-of-m secret sharing



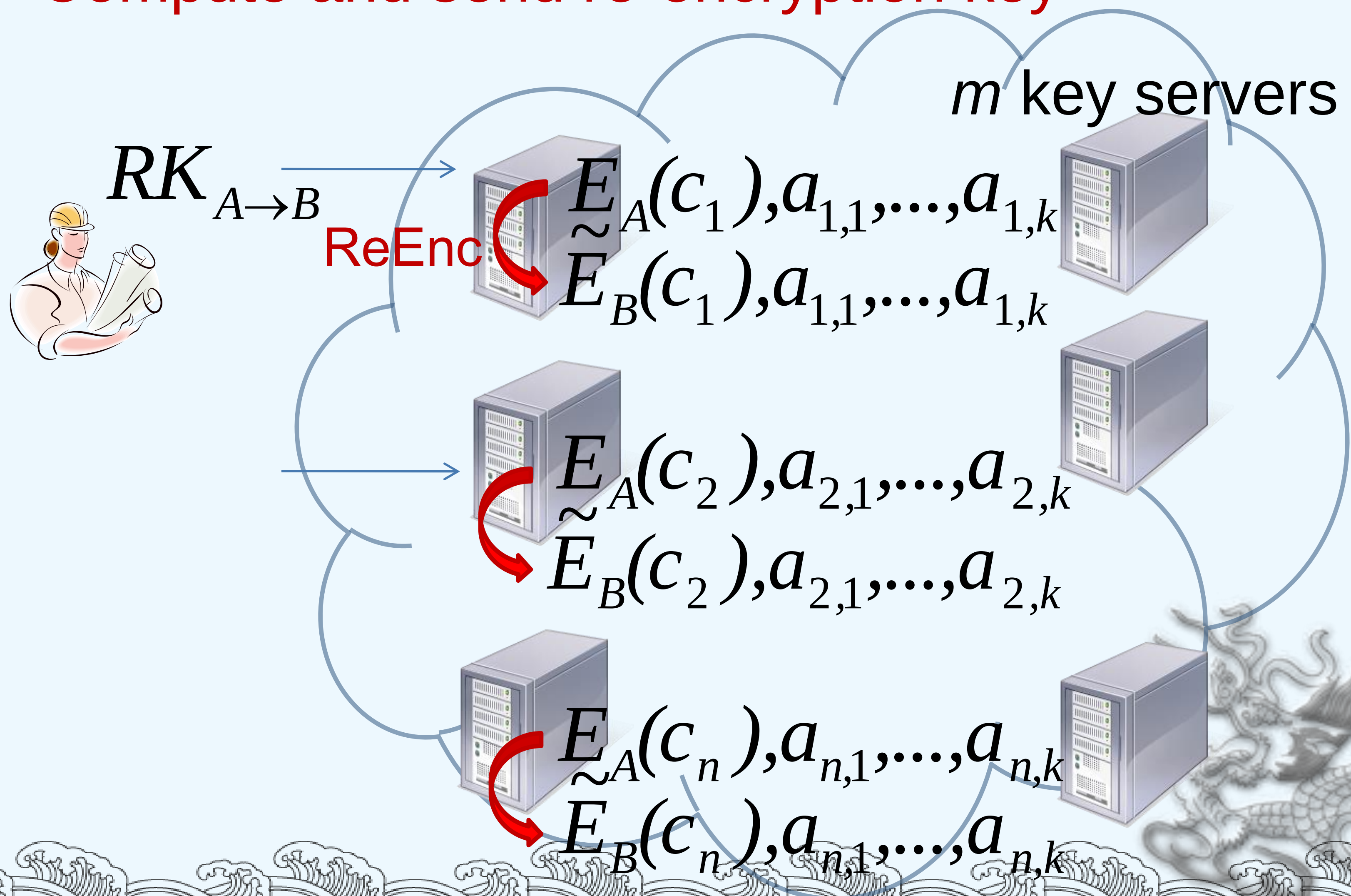
Storing Process

Coding over encrypted messages



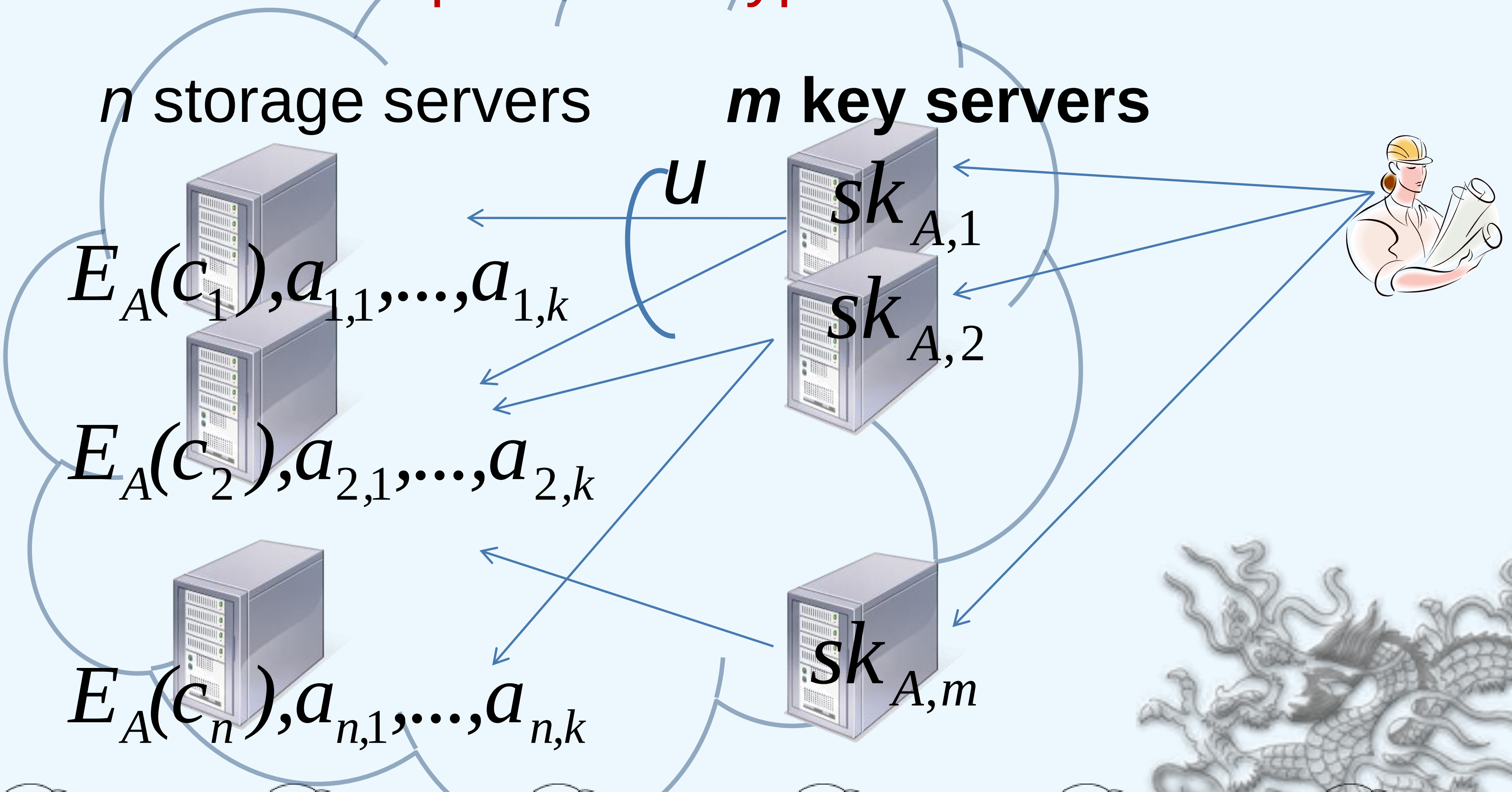
Forwarding Process

Compute and send re-encryption key



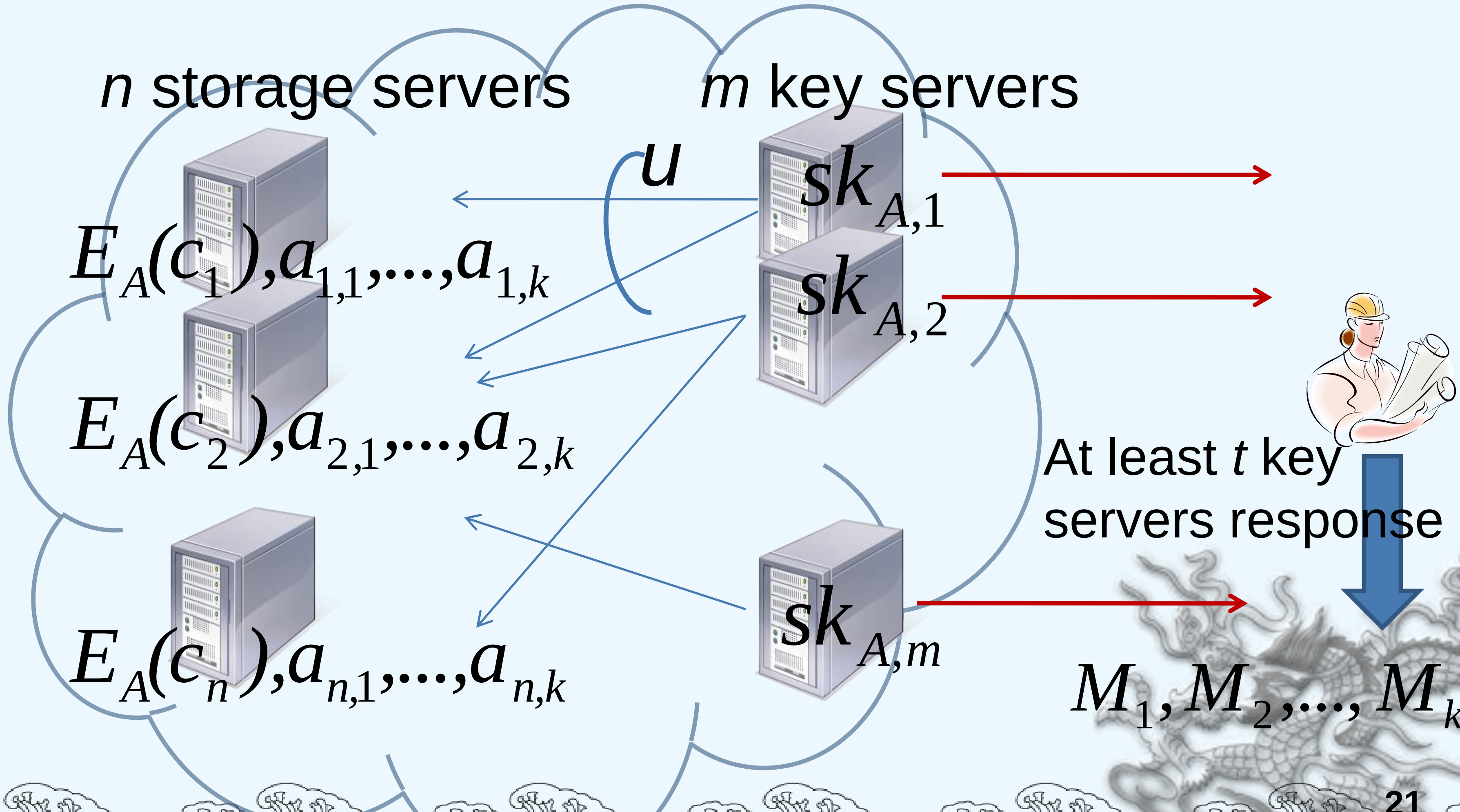
Retrieval Process - Owner

Decentralized partial decryption

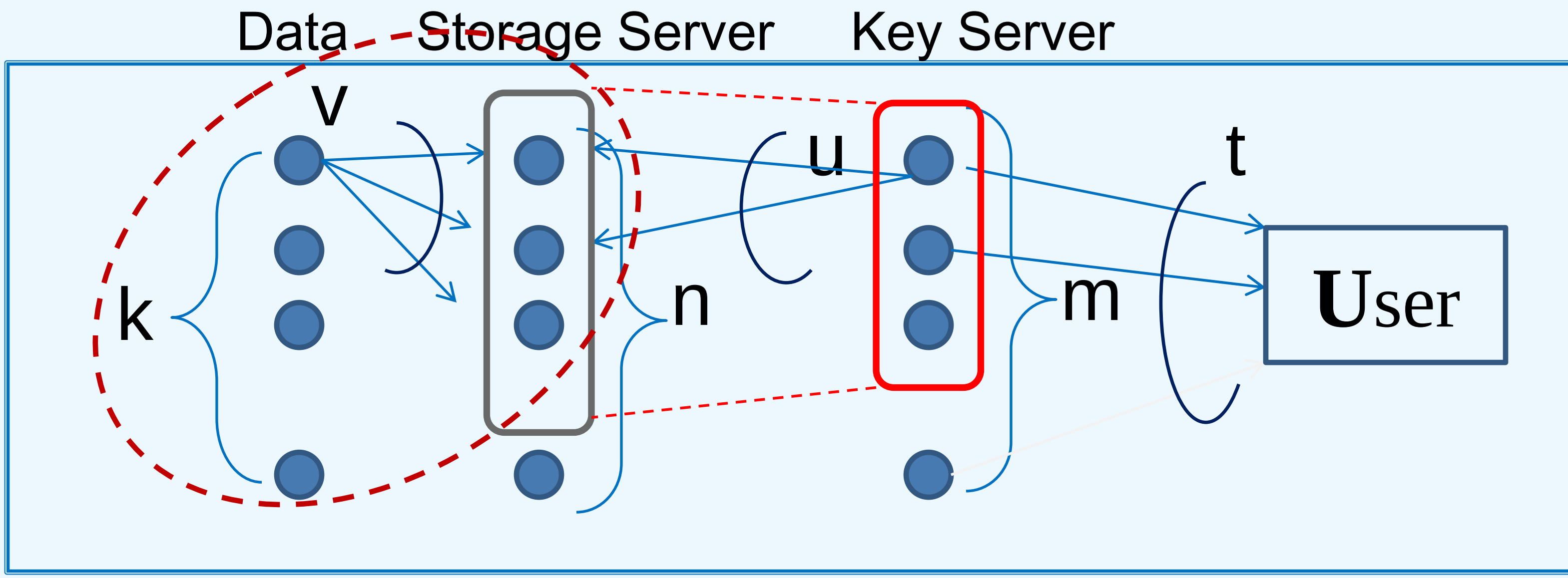


Retrieval Process - Owner

Combine partial decryption and decoding



Success Retrieval



Conditions

1. #SSs chosen by KSs is at least k
2. det(submatrix) ≠ 0 mod p ⟶ det(submatrix) ≠ 0, det(submatrix) ≠ rp

Observations

- det(submatrix) ≠ 0 iff a perfect matching

Parameters

$$\text{For } n = ak^c, m \ge t \ge k, a > \sqrt{2}, v = bk^{c-1}\ln k, c \ge 3/2$$
$$u = 2, b > 5a, \Pr[\text{success retrieval}] > 1 - k/p - o(1)$$

By combinational bound:

$$\Pr[\#\mathrm{SS} < \mathrm{k}] \le \mathrm{C}^{\mathrm{n}}_{\mathrm{k}\text{-}1}(1/n)^{uk} = o(1)$$

By Hall's Theorem:

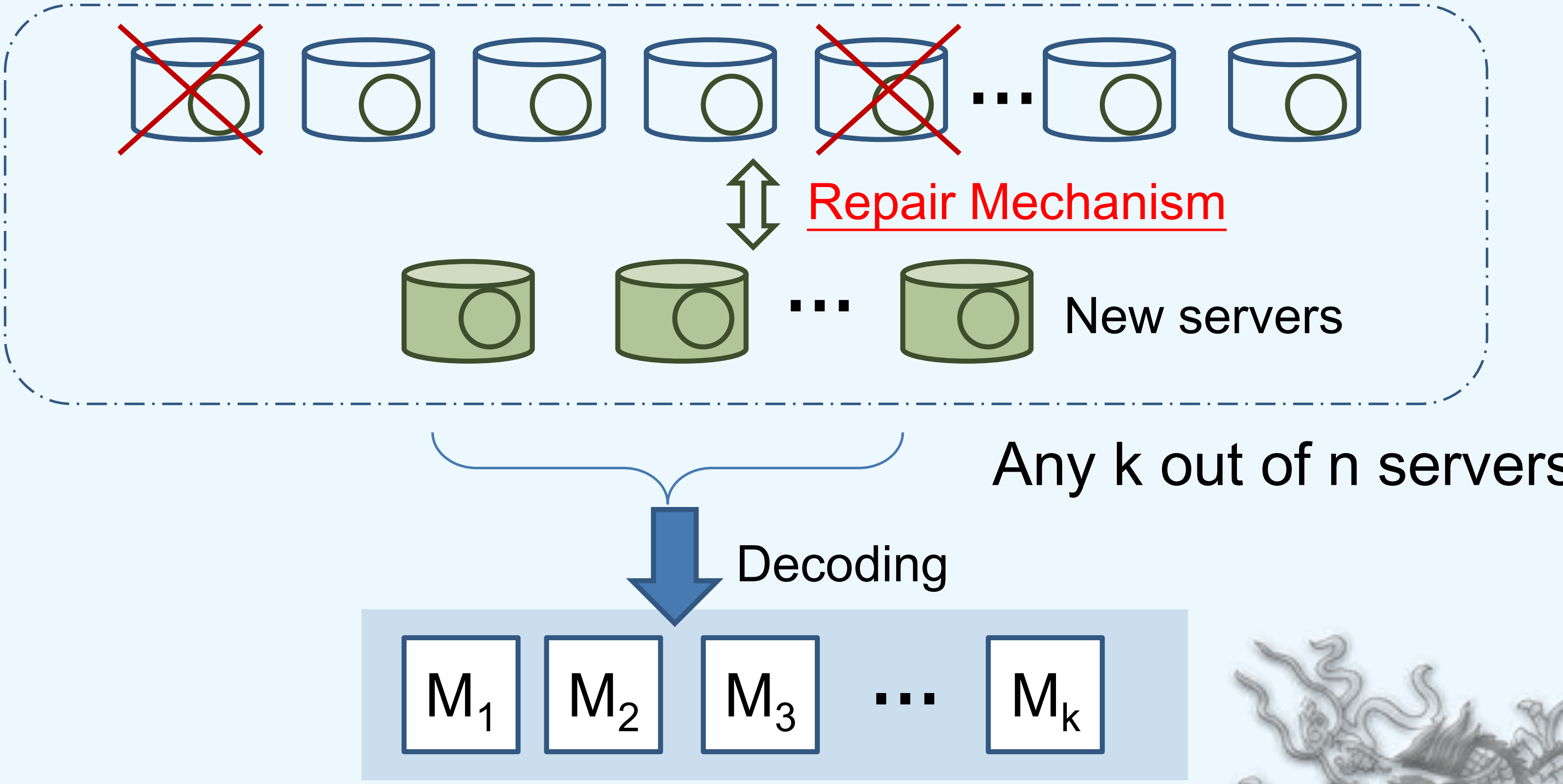
$$\Pr[\det = 0 \mid \#\,\mathrm{SS} \ge \mathrm{k}] = \Pr[\text{no perfect matching}] = o(1)$$

By Schwartz-Zippel Theorem (for random coefficients):

$$\Pr[\det = rp \text{ for some integer } r \mid \#\,\mathrm{SS} \ge \mathrm{k}, \det \ne 0] \le \mathrm{k/p}$$

Repair Issue

Maintain data robustness against server failure



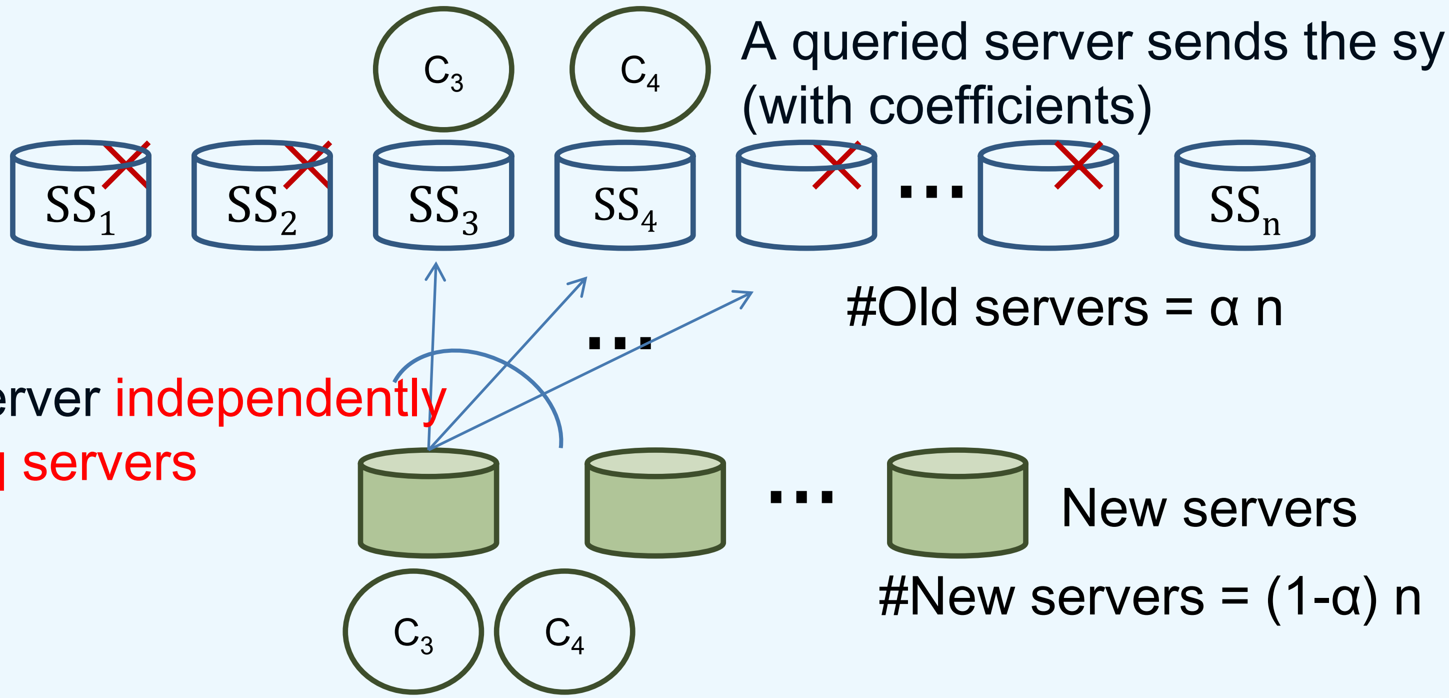
Repair Mechanisms

- Straightforward solution:
  - Reconstruct the original message and encode it again
  - Need to query k old servers
- Another approach
  - Generate a missing symbol by combining **q** available symbols from old servers
  - Objective: less repair bandwidth and storage cost
  - Question: can q be less than k?

科学技術振興機構
Japan Science and Technology Agency

行政院國家科學委員會
National Science Council

Our Repair Mechanism



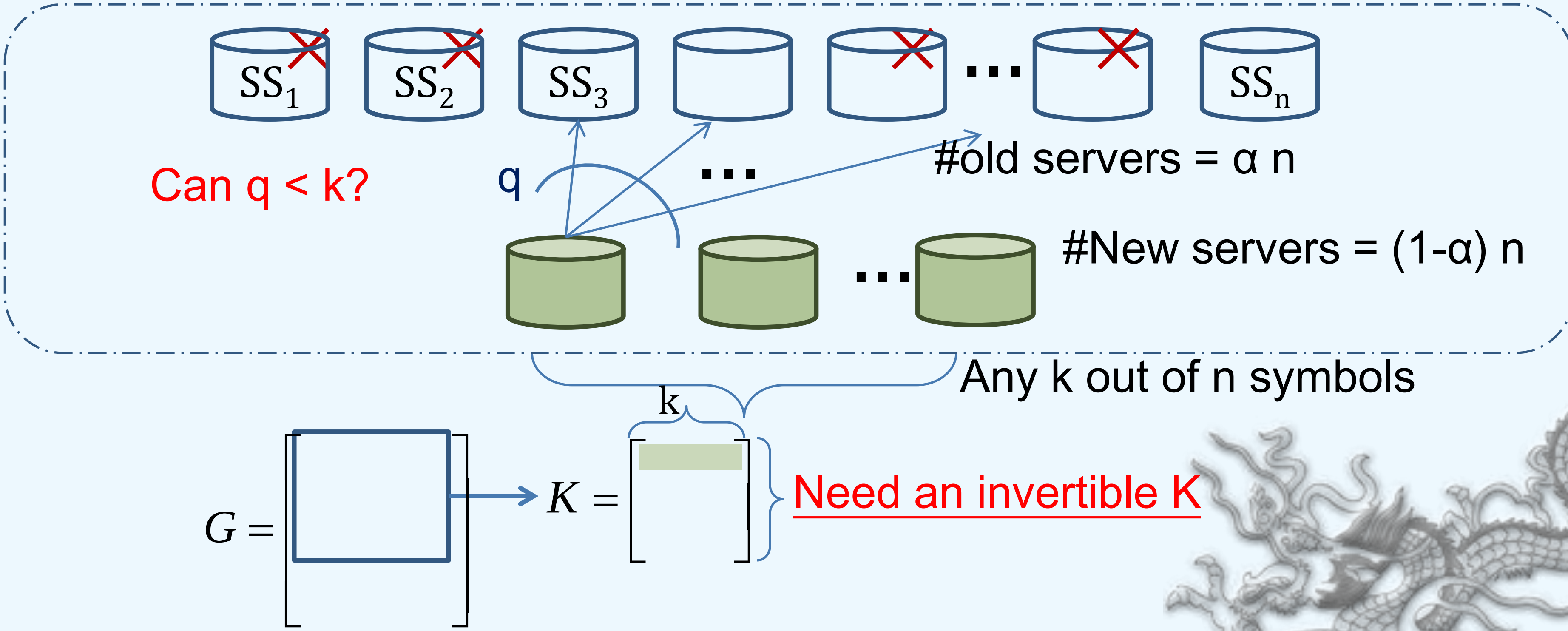
A new server independently queries q servers

A new server encodes received symbols as one (new symbol)

$$\begin{cases} a & C_3, (a_{3,1}, a_{3,2}, \ldots, a_{3,k}) \\ b & C_4, (a_{4,1}, a_{4,2}, \ldots, a_{4,k}) \end{cases} \Rightarrow C_3^{\ a} \otimes C_4^{\ b}, \; (aa_{3,1} + ba_{4,1}, aa_{3,2} + ba_{4,2}, \ldots, aa_{3,k} + ba_{4,k})$$

About q

- When q is small, a new server gets less information
  → K may not have full rank (not invertible)
  → decrease Pr[successful message retrieval]



Main Result

There are k^d old servers. $(1-\alpha)n$ new servers join the system, where

$$n = ak^c, a > \sqrt{2}, c \geq 1, \alpha n = k^d, d > 1, \alpha < 1$$

Let q be set s.t.

$$q \geq \min\{k, \max\{\frac{2k}{(d-1)\ln k}, \frac{k}{(d-1)\ln k} + \frac{d}{d-1}\}\}$$

After the system is repaired,

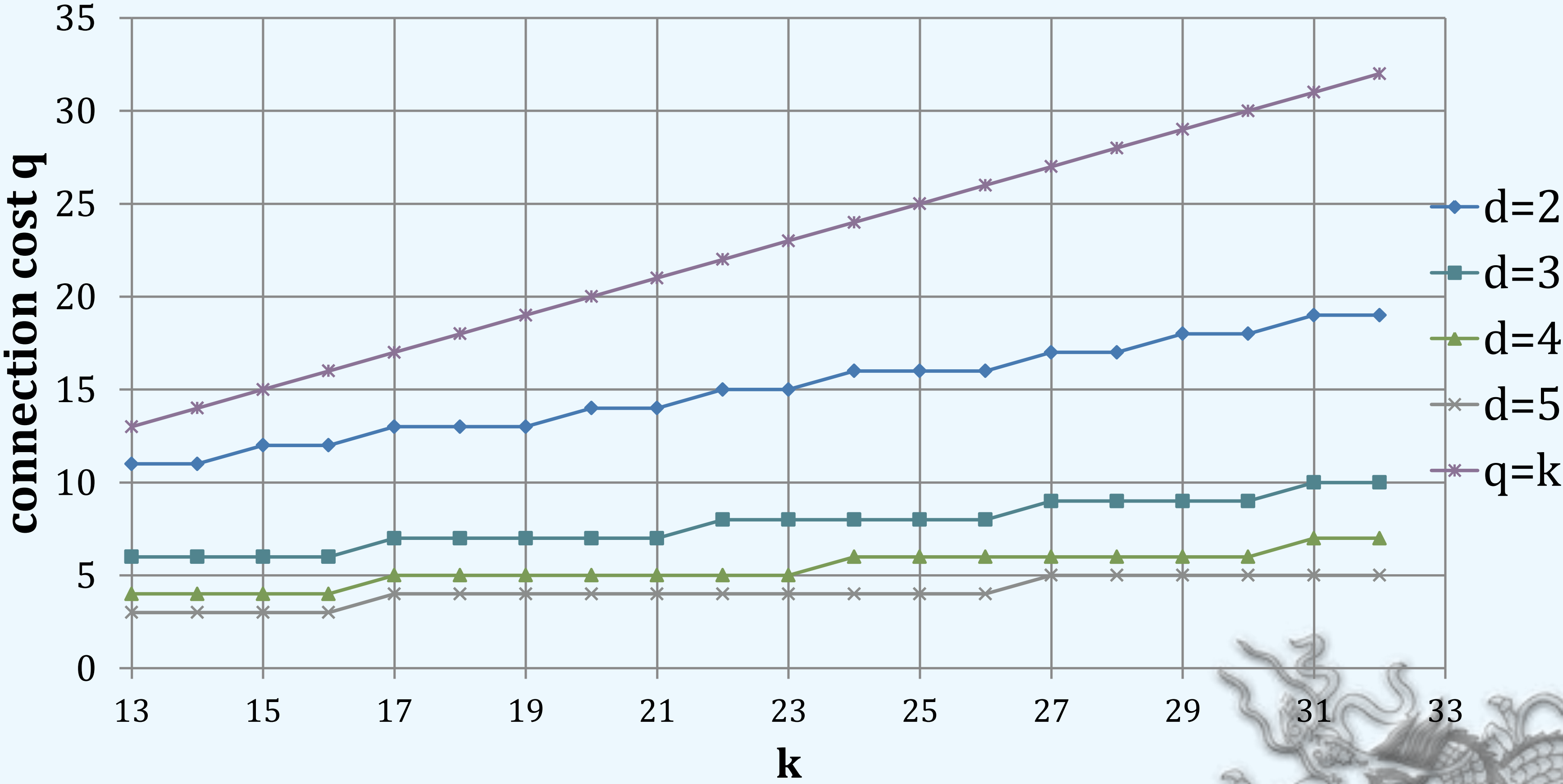
$$\Pr[\text{ successful message retrieval }] \geq 1 - \frac{2k}{p} - o(1)$$

"q can be less than k"

The bound on q is related to k and d

Numerical Results

- Bring k and d to find the smallest q



Summary

- Decentralized networked storage system
  - Data robustness
  - Strong data confidentiality
  - Key management
  - Multiple functionalities
    - Secure data forwarding
    - Repair mechanism
    - Integrity check (not in this talk)

Future Work

- Data robustness against <u>faulty errors</u>
  - Detect/correct when stored data are altered
  - Support efficient coding operations
- Different repair model
  - Mutual communication among new servers
- More functionality
  - Support decentralized integrity check
  - Support keyword search