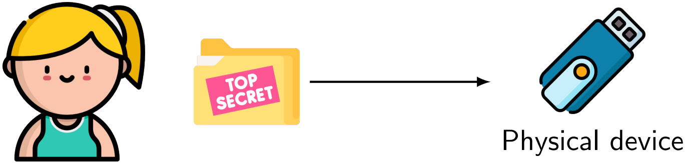
Secure Keyless Multi-Party Storage Scheme

Pascal Lafourcade, Lola-Baie Mallordy, Charles Olivier-Anclin, Léo Robert

FIC, April 2025

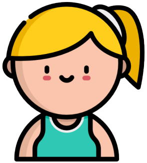
How to store a secret ?

How to store a secret ?

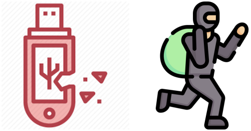


Physical device

How to store a secret ?

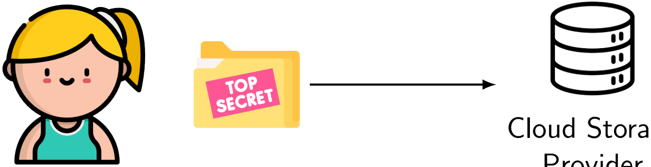


Secret lost!



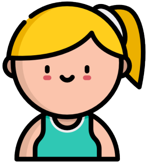
Physical loss

How to store a secret ?



Cloud Storage Provider

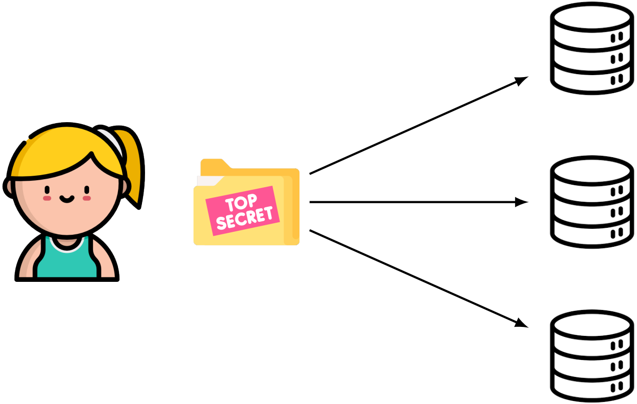
How to store a secret ?



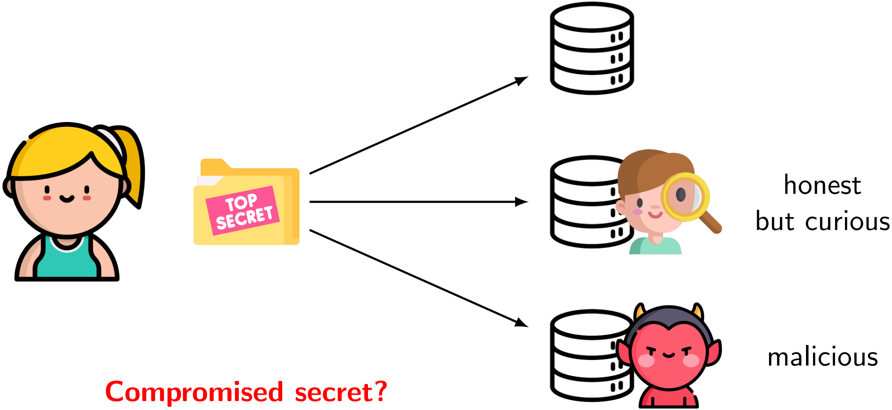
Secret lost!

Single Point of Failure

Multi-Cloud Storage



Dangers in multi-cloud storage – Trust issues

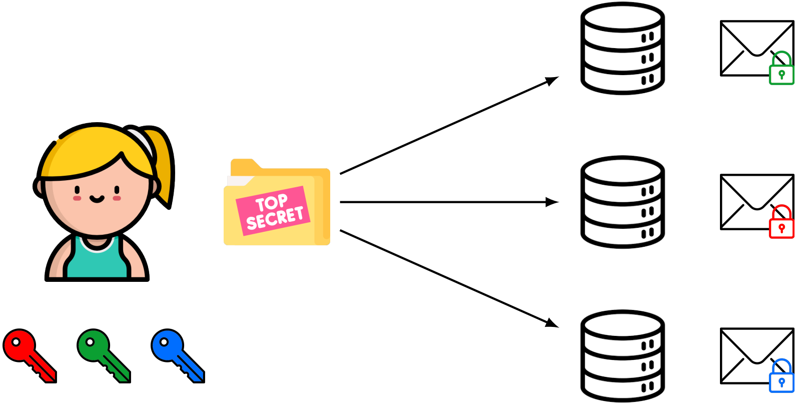


honest
but curious

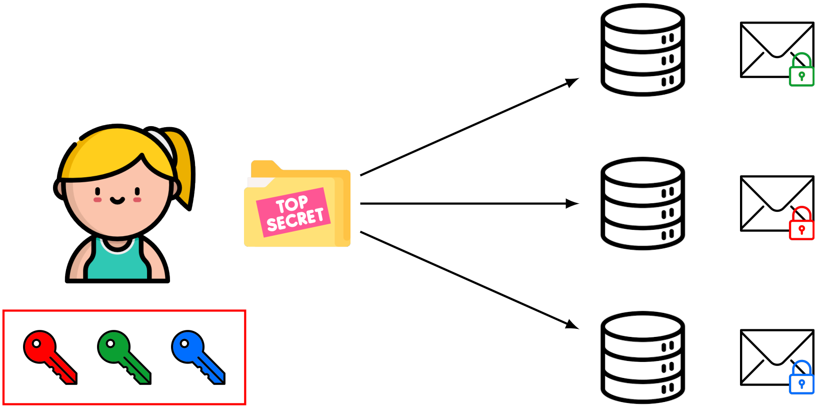
malicious

Compromised secret?

Dangers in multi-cloud storage – Key(s) management

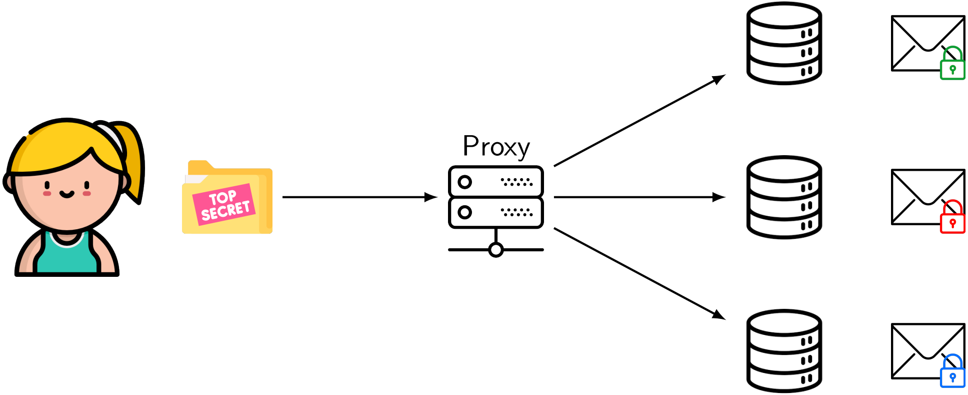


Dangers in multi-cloud storage – Key(s) management



The new top secret

Keyless in a multi-cloud setting



- Centralized authentication
- Only the owner know the secret
- Detection of modifications on the secret must be detected
- Accountability

State of the art

Multi-cloud Protocols	Confidential w.r.t. proxy	Providers collusion	Proxy collusion	Keyless
E. Stefanov et al. 2013	—	✗	—	✗
R. D. Pietro et al. 2017	✗	✗	✗	✗
M. Leila et al. 2020	✗	✗	✗	✗
A. Niknia et al. 2021	—	✓	—	✓
A. N. Bessani et al. 2013	—	✗	—	✓
M. Sulochana et al. 2015	✗	✗	✗	✗
E. N. Witanto et al. 2023	✗	✓	✗	✗
KAPRE	✓	✓	✗	✓
KAME	✓	✓	✓	✓

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

KAPRE

KAME

Common download

Experiments

Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

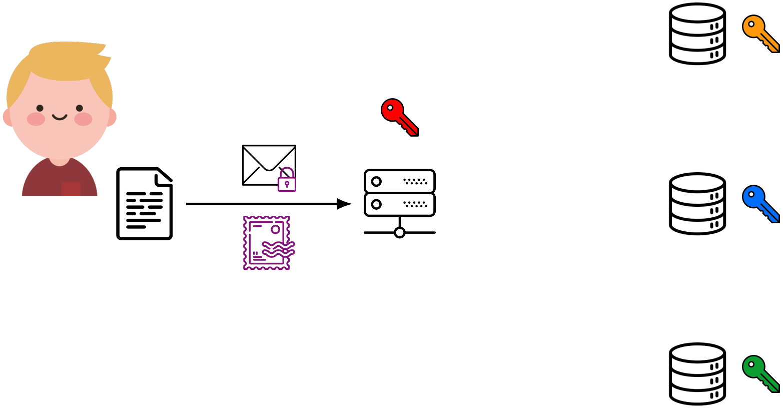
KAPRE

KAME

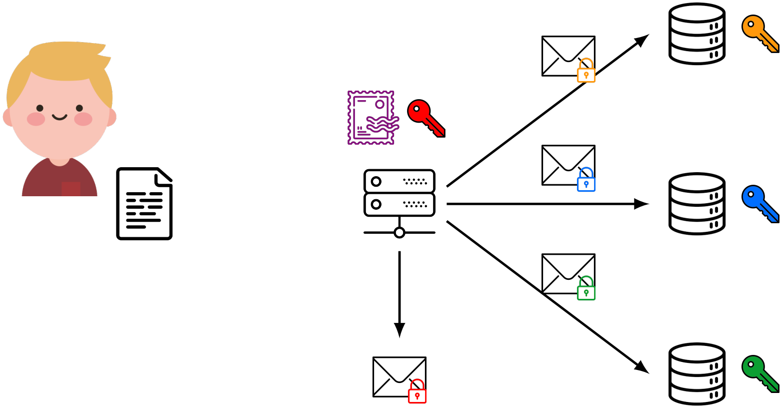
Common download

Experiments

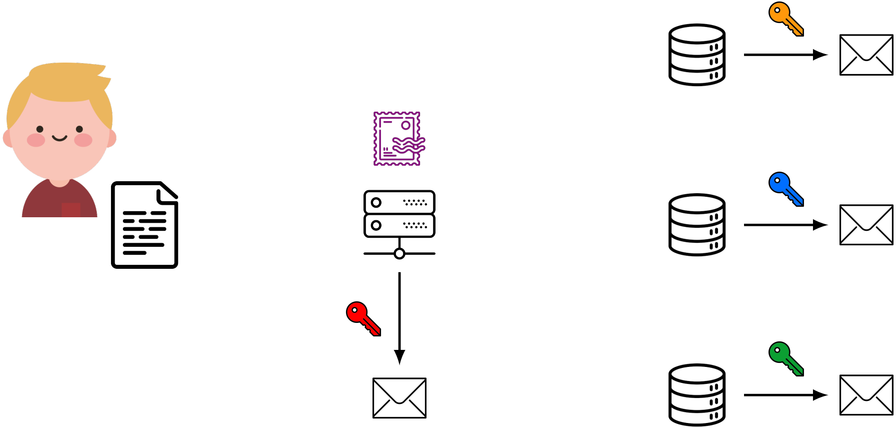
Upload – Transform



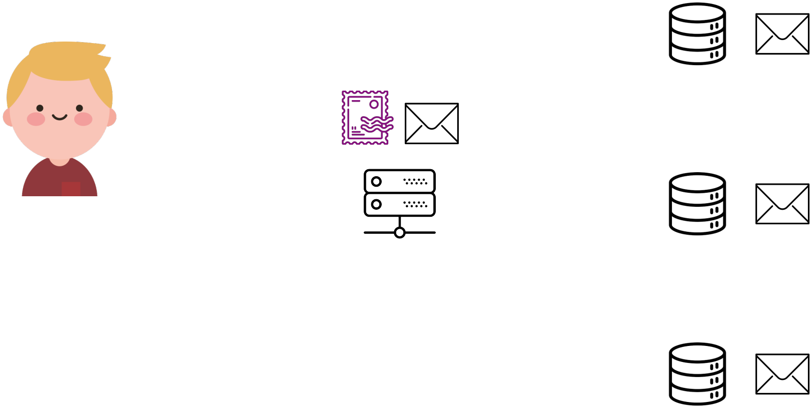
Upload – Distrib



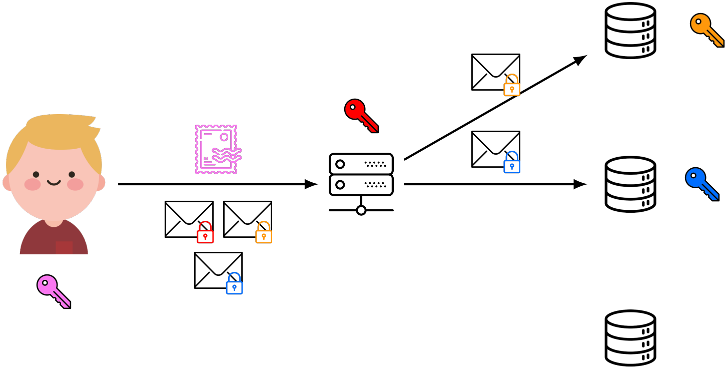
Upload – Open



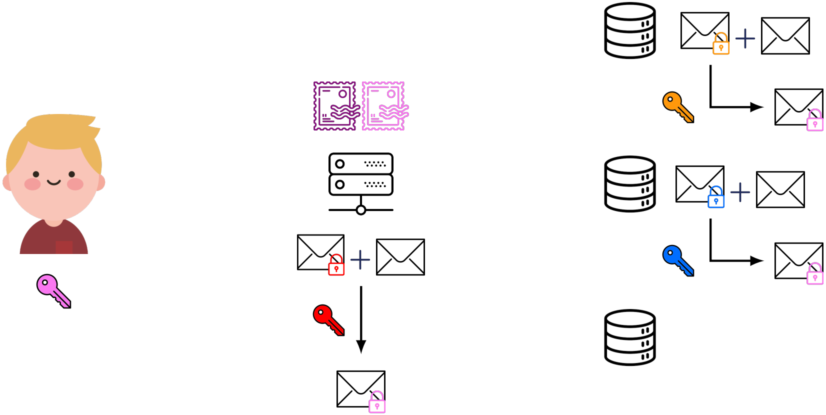
Upload – Final State



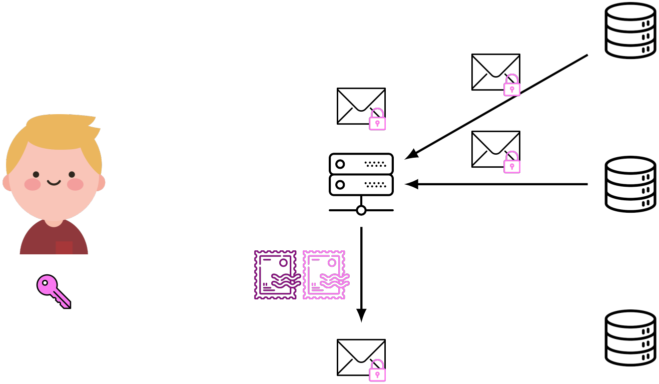
Download – Designate



Download – Hide

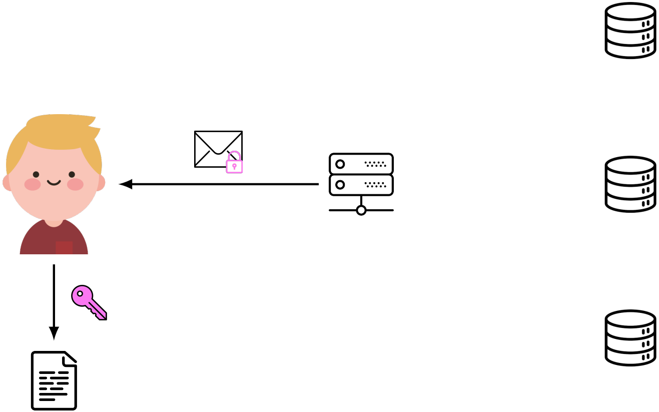


Download – Merge



or blame the culprit(s)!

Download – Recover



or $\perp$

Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

KAPRE

KAME

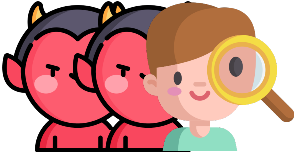
Common download

Experiments

Adversary model

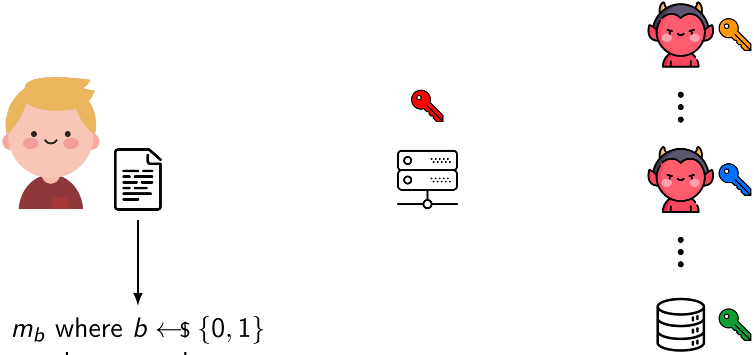
Proxy
Honest but curious

Servers
Malicious



Collusion
of adversaries

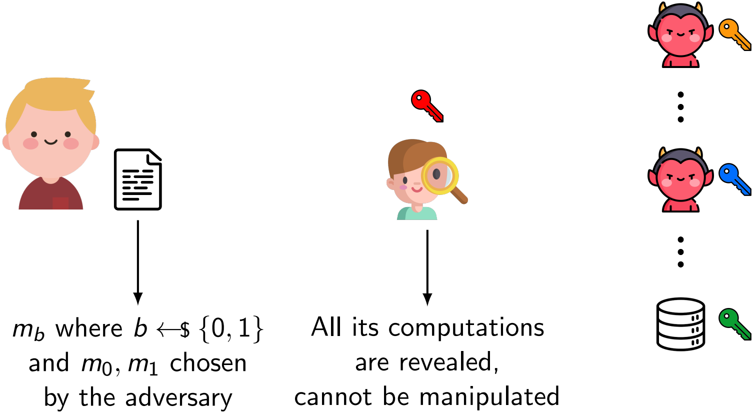
$k-$providers secrecy



m_b where $b \leftarrow\!\!\$ \{0, 1\}$
and m_0, m_1 chosen
by the adversary

$k-$collusion secrecy

Guess the bit b ?

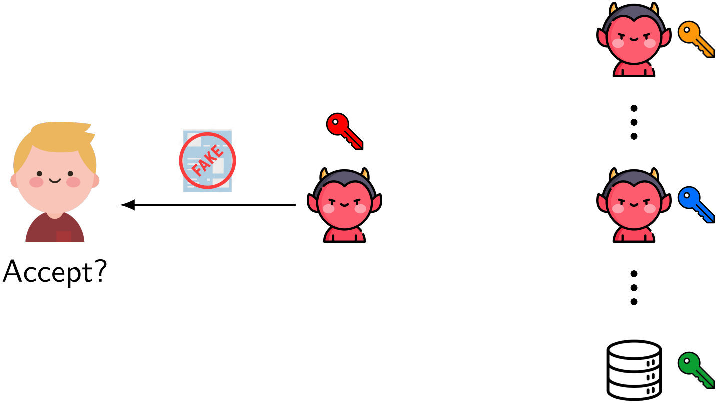


m_b where $b \leftarrow\!\!\$ \, \{0,1\}$ and m_0, m_1 chosen by the adversary

All its computations are revealed, cannot be manipulated

User integrity

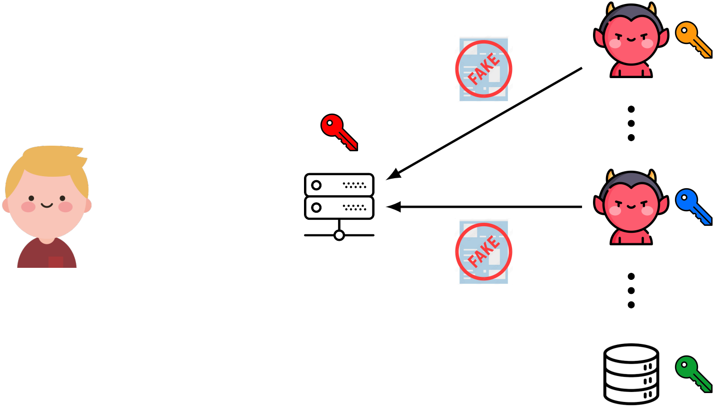
After an honest upload of a message chosen by the adversary, send a corrupted secret accepted by the user.



Accept?

Accountability

After an upload of a message chosen by the adversary, send back corrupted shares such that either the proxy accepts them, or blame uncorrupted shares.



Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

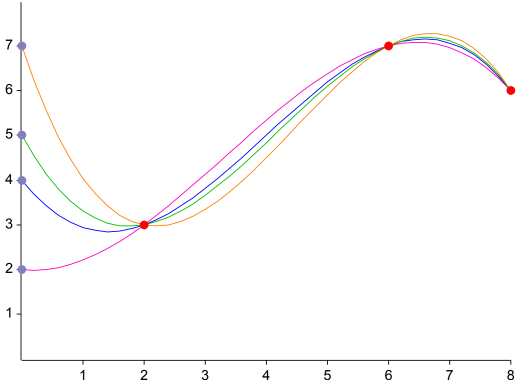
KAPRE

KAME

Common download

Experiments

Shamir's secret sharing – Shamir, 1979

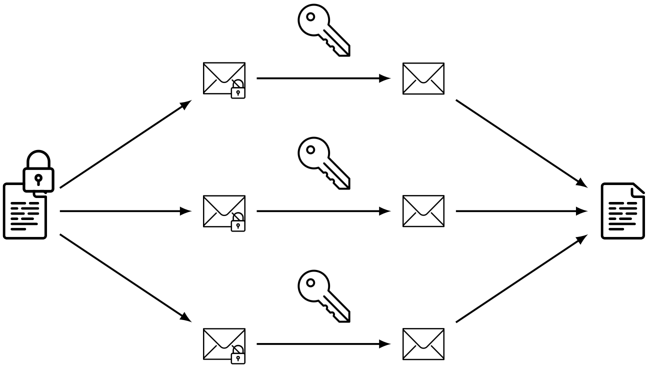


```
Split $(k, n, m \in \mathbb{Z}_p)$ :
$a_1, \dots, a_{k-1} \leftarrow\!\!\$\ \mathbb{Z}_p$,
$x_1, \dots, x_n \leftarrow\!\!\$\ \mathbb{Z}_p^\times$ pairwise distinct,
$P(x) = m + \sum_{i=1}^{k-1} a_i X^i$,
  return $(x_1, P(x_1)), \dots, (x_n, P(x_n))$

Reconstruct $(k, (x_1, y_1), \dots, (x_k, y_k))$ :
  return $\sum_{i=1}^{k} y_i \prod_{j \neq i} \frac{-x_j}{x_i - x_j}$.
```

Homomorphic encryption – Brakerski, Gentry, Vaikuntanathan, 2014

$$\mathrm{Dec}(\mathrm{Enc}(m, \mathrm{pk}) + \mathrm{Enc}(n, \mathrm{pk}), \mathrm{sk}) = m + n$$



Key homomorphic pseudorandom function family – Banerjee, Peikert 2014

For all $x \in D$,

$$F_a(x) \cdot F_b(x) = F_{a+b}(x).$$

Information Dispersal Algorithm (IDA) – Rabin, 1989

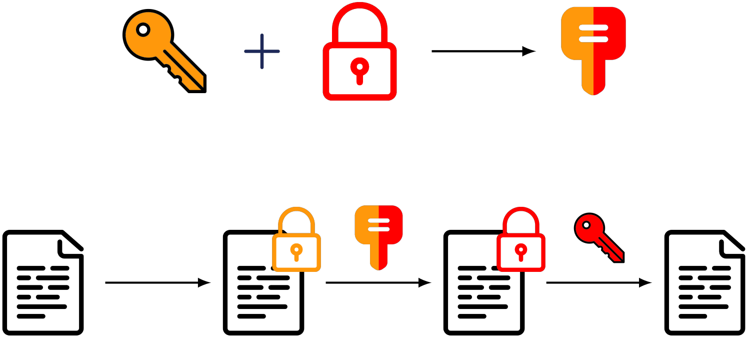
$\mathrm{Split}((m_1, \ldots, m_k) \in \mathbb{Z}_p^k, n, k)$: $A \leftarrow\$\, \mathbb{Z}_p^{k \times n}$ such that every $k \times k$ submatrix of A is invertible,

$$\textbf{return} \quad A \; m = \begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix} \in \mathbb{Z}_p^n.$$

$\mathrm{Rec}\big(A, r_{i_1}, \ldots, r_{i_k}\big)$: Let A' be the $k \times k$ submatrix formed by the lines $i_1, \ldots, i_k$ of A,

$$\textbf{return} \quad {A'}^{-1} \begin{bmatrix} r_{i_1} \\ \vdots \\ r_{i_k} \end{bmatrix} = m \in \mathbb{Z}_p^k.$$

Proxy Re-Encryption – KeySwitching (BGV)



Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

KAPRE

KAME

Common download

Experiments

Upload KAPRE $(n = 3, k)$ – Transform

User:

$$\mathsf{recK} \leftarrow \mathsf{E.KeyGen}$$

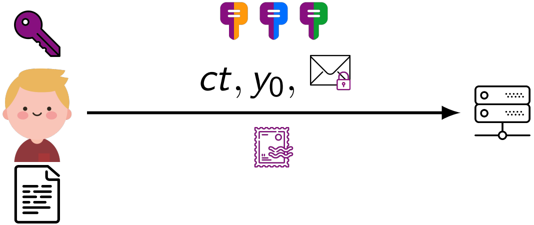
$$ct \leftarrow \{\text{📄}\}_{\mathsf{recK}}$$

$$a_1, \ldots, a_{k-1} \xleftarrow{\$} \mathbb{Z}_p$$

$$y_0 \leftarrow \mathsf{recK} + \sum_{i=1}^{k-1} a_i$$

$$\text{✉} \leftarrow \{\mathsf{recK}\}_{\text{🔒}}, \{\{a_i\}_{\text{🔒}}\}_{i=1}^{k-1}$$

$$\text{🖼} \leftarrow x, F_{\mathsf{recK}}(x), \{F_{a_i}(x)\}_{i=1}^{k-1}$$



Upload KAPRE ($n = 3, k$) – Distrib

Proxy:

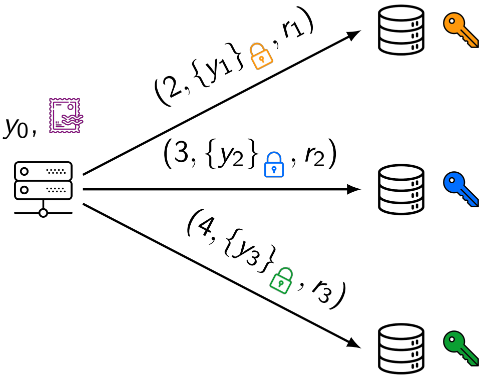
$\{r_i\}_i \leftarrow \mathrm{IDA.Split}(ct, n+1, k)$

$\{y_i\}_{\text{🔒}} \leftarrow \{\mathrm{recK}\}_{\text{🔒}} + \sum_{j=1}^{k-1}\{a_j\}_{\text{🔒}}(i+1)^j$

$\{y_1\}_{\text{🔒}} \leftarrow \mathrm{PRE.ReEnc}(\{y_1\}_{\text{🔒}}, \text{🔑})$

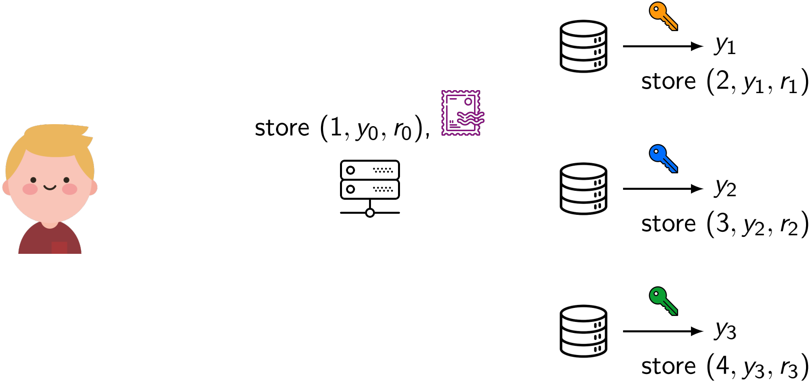
$\{y_2\}_{\text{🔒}} \leftarrow \mathrm{PRE.ReEnc}(\{y_2\}_{\text{🔒}}, \text{🔑})$

$\{y_3\}_{\text{🔒}} \leftarrow \mathrm{PRE.ReEnc}(\{y_3\}_{\text{🔒}}, \text{🔑})$



y_0,
$(2, \{y_1\}_{\text{🔒}}, r_1)$
$(3, \{y_2\}_{\text{🔒}}, r_2)$
$(4, \{y_3\}_{\text{🔒}}, r_3)$

Upload KAPRE $(n = 3, k)$ – Open



store $(1, y_0, r_0)$,

$\xrightarrow{} y_1$

store $(2, y_1, r_1)$

$\xrightarrow{} y_2$

store $(3, y_2, r_2)$

$\xrightarrow{} y_3$

store $(4, y_3, r_3)$

Weakness of KAPRE

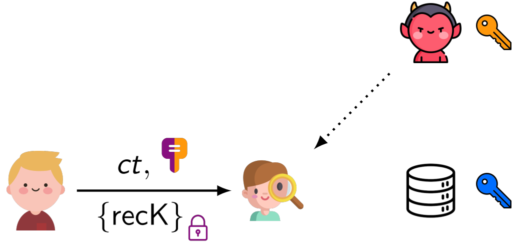
Adversary:

{recK}🔒 ← PRE.ReEnc({recK}🔒, 🗝)

recK ← PRE.Dec({recK}🔒, 🔑)

📄 ← E.Dec(ct, recK)

No secrecy for the user's data!



Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

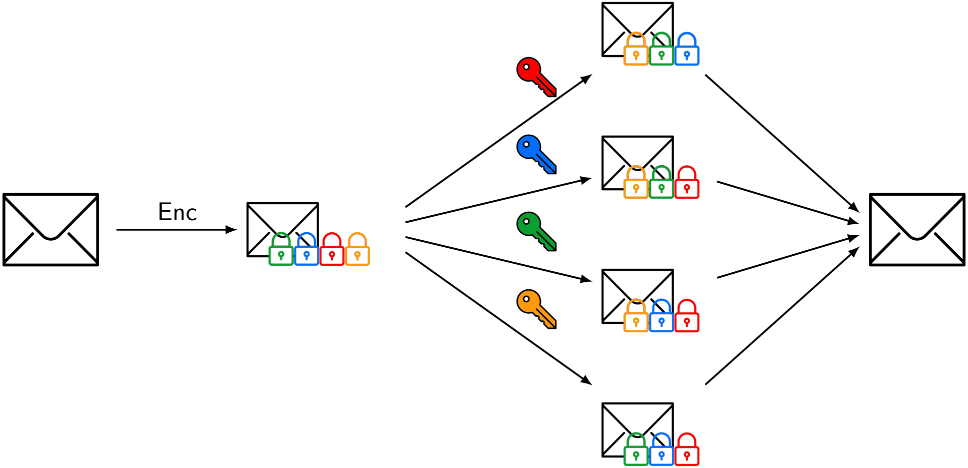
KAPRE

KAME

Common download

Experiments

Multi-Key Encryption Scheme – Lòpez-Alt et al., 2012



Upload KAME ($n = 3, k$) – Transform

User:

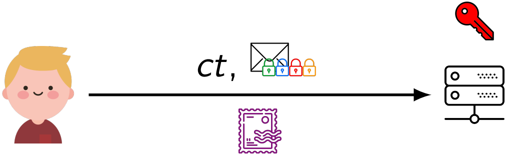
$\mathsf{recK} \leftarrow \mathsf{E.KeyGen}$

$ct \leftarrow \{$📄$\}_{\mathsf{recK}}$

$a_1, \ldots, a_{k-1} \leftarrow\!\!\$\ \mathbb{Z}_p$

✉🔒🔒🔒🔒 $\leftarrow \{\mathsf{recK}\}_{🔒🔒🔒🔒}, \{\{a_i\}_{🔒🔒🔒🔒}\}_{i=1}^{k-1}$

📮 $\leftarrow x, F_{\mathsf{recK}}(x), \{F_{a_i}(x)\}_{i=1}^{k-1}$

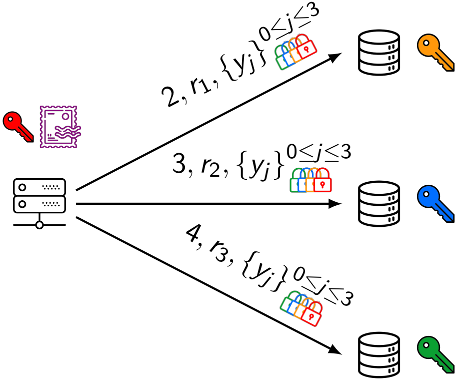


Upload KAME ($n = 3, k$) – Distrib

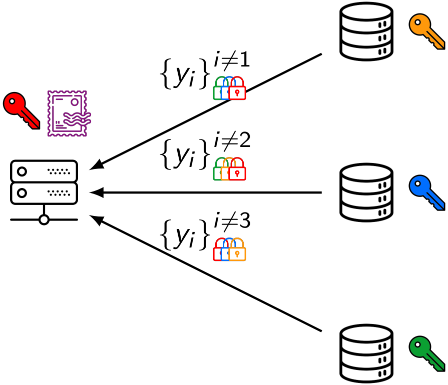
Proxy:

$$\{y_i\}_{🔒} \leftarrow \{\mathrm{recK}\}_{🔒} + \sum_{j=1}^{k-1} \{a_j\}_{🔒} (i+1)^j$$

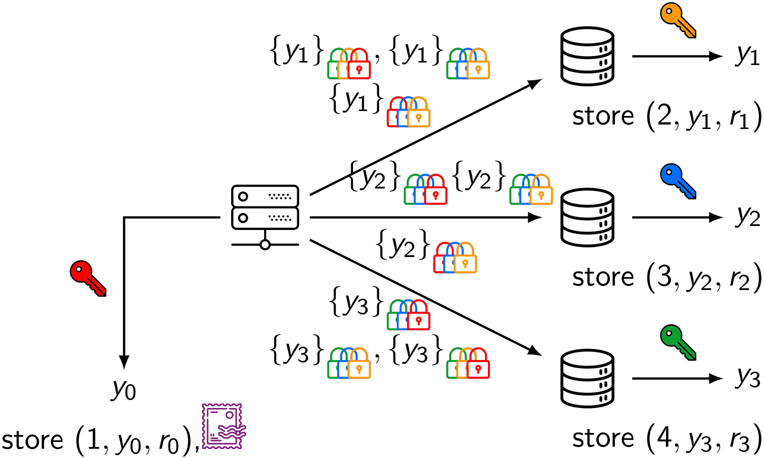
$$\{r_i\} \leftarrow \mathrm{IDA.Split}(ct, n+1, k)$$


$2, r_1, \{y_j\}_{🔒}^{0 \leq j \leq 3}$
$3, r_2, \{y_j\}_{🔒}^{0 \leq j \leq 3}$
$4, r_3, \{y_j\}_{🔒}^{0 \leq j \leq 3}$

Upload KAME $(n = 3, k)$ – Open



Upload KAME ($n = 3, k$) – Open



Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

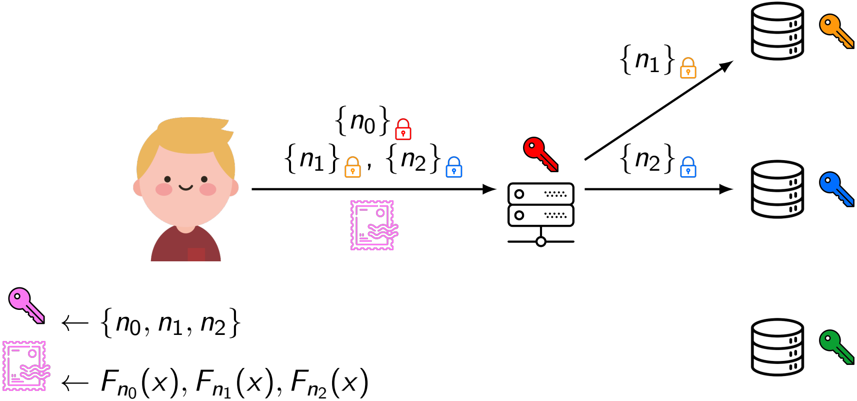
KAPRE

KAME

Common download

Experiments

Download $(n = 3, k = 3)$ – Designate



$\{n_0\}$

$\{n_1\}$, $\{n_2\}$

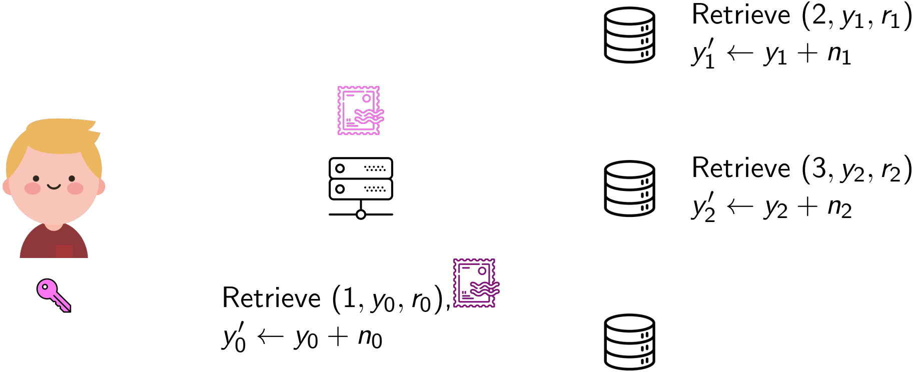
$\{n_1\}$

$\{n_2\}$

$\leftarrow \{n_0, n_1, n_2\}$

$\leftarrow F_{n_0}(x), F_{n_1}(x), F_{n_2}(x)$

Download $(n = 3, k = 3)$ – Hide



Retrieve $(2, y_1, r_1)$
$y_1' \leftarrow y_1 + n_1$

Retrieve $(3, y_2, r_2)$
$y_2' \leftarrow y_2 + n_2$

Retrieve $(1, y_0, r_0)$,
$y_0' \leftarrow y_0 + n_0$

Download $(n = 3, k = 3)$ – Merge

Proxy:

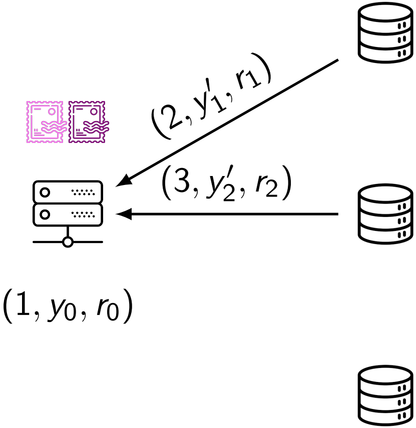
shiftK $\leftarrow \sum_{i=0}^{2} y_i' \ell_i$

if $F_{\text{recK}}(x) + \sum F_{n_i}(x)\ell_i = F_{\text{shiftK}}(x)$:

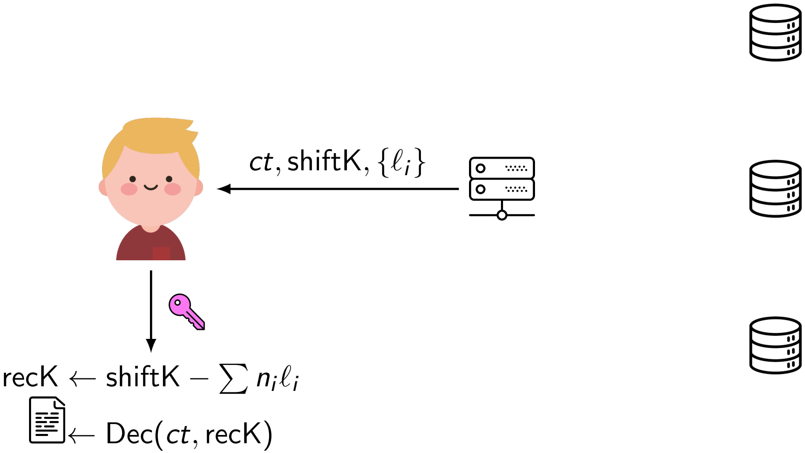
$$ct \leftarrow \text{IDA.Rec}(\{r_i\}, 3)$$

else blame every party for which

$$F_{y_i'}(x) \neq F_{n_i}(x) + F_{\text{recK}}(x) + \sum_{j=1}^{k-1} F_{a_j} x_i^j$$



Download – Recover



$ct, \mathsf{shiftK}, \{\ell_i\}$

$\mathsf{recK} \leftarrow \mathsf{shiftK} - \sum n_i \ell_i$

📄 $\leftarrow \mathsf{Dec}(ct, \mathsf{recK})$

Security

Multi-cloud Protocols	Confidential w.r.t. proxy	Providers collusion	Proxy collusion	Keyless
KAPRE	✓	✓ k-1	✗	✓
KAME	✓	✓ k-2	✓	✓

Outline

Introduction

Multi-cloud storage

Adversary and Properties

Cryptographic background

KAPRE

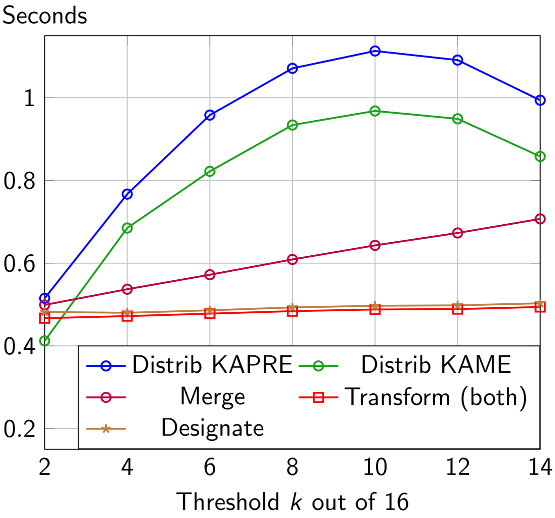
KAME

Common download

Experiments

Experiments – Average execution time comparison

Benchmarks:
Ubuntu 22.04.2 laptop
messages of 1MB



Complexity for a (n, k) sharing

Protocols	Security	Complexity	Communication
Upload KAPRE	Proxy, collusion of servers	$\mathcal{O}(nk - k^2)$	One round
Upload KAME	Proxy colluding with servers	$\mathcal{O}(nk - k^2)$	Interactive
Download	Collusion proxy with serveurs	$\mathcal{O}(k)$	One round

Thank you for your attention !