Secure Vault Scheme in the Cloud Operating Model

Rishiraj Bhattacharyya, Avradip Mandal, **Meghna Sengupta**


UNIVERSITY OF BIRMINGHAM


Zfense Labs


THE UNIVERSITY of EDINBURGH

OUTLINE

- Data Privacy Vaults
  - *What do they do?*
  - *Who's making them? Who's using them?*
- Our framework
- Building Block: Tokenization Scheme
  - *Definition*
  - *Security Notions*
- Complete Vault Scheme Construction
  - *Construction*

1

DATA-PRIVACY VAULTS

- What do they do?
- Who is using them?

Data-Privacy Vault

A literal vault (safe) of information -

- Users can store their private information
- Generate access passes
  - give access to selected parties
- Data hidden from others

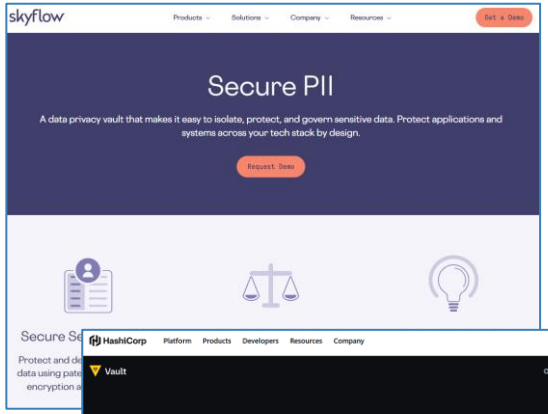
Data-Privacy Vault

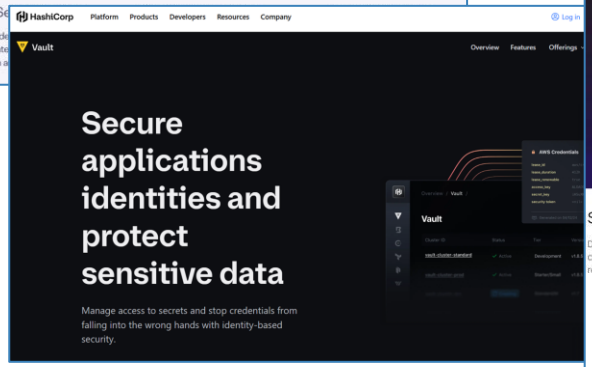
Access Levels

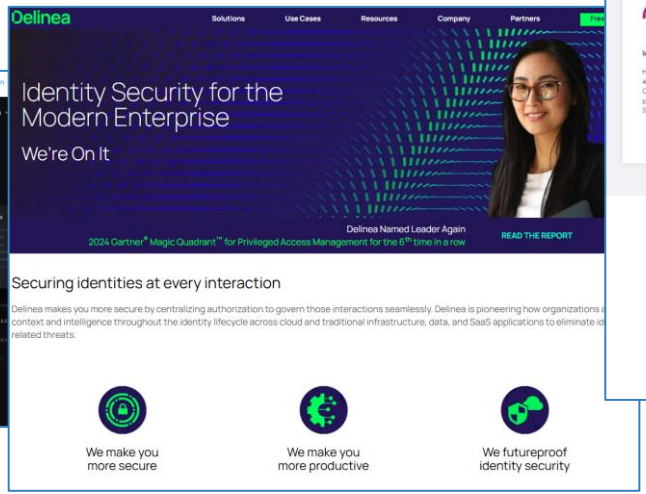
- Different levels of access passes
  - provides access to different data
- Eg. Level 1: Non-sensitive data
  Level 2: Both non-sensitive and sensitive data

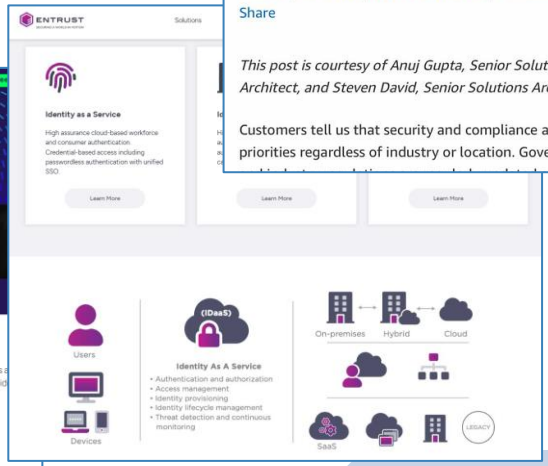
Who's making & using them?

A lot of big companies!









Who’s making & using them?

A lot of big companies!

increasingly being used for
storing information used
for training LLM models

2

OUR FRAMEWORK

Our Framework

3 parties:

TRUSTED SERVER

DATABASE

USER

ANALYST

Our Framework - Functionality

- STORE -
  - User stores the information and gets back access passes
- ACCESS -
  - Level 1 access to non-sensitive information
- RETRIEVE -
  - Level 2 access to all the information

STORE


NON-SENSITIVE DATA
SENSITIVE DATA
DATA TOKEN
ACCESS TOKEN
MASTER PASSWORD
USER
Mr Ruffles
TRUSTED SERVER
DATABASE

RETRIEVE

USER

TRUSTED
SERVER

ACCESS

ANALYST 2

USER

ANALYST 1

ANALYST 3

TRUSTED SERVER

Security Notions

IND-CATA

- Non-Sensitive Information:

  No access Token -> No non-sensitive information

- Sensitive Information:

  No master password -> No sensitive information

  (even with access token)

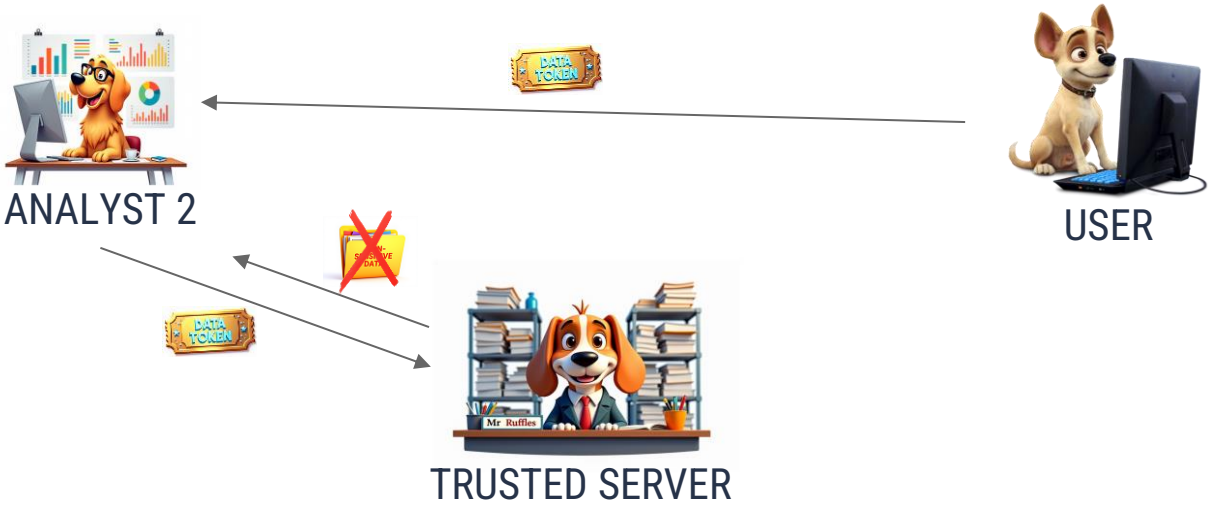
IND-SIA

Security Notions

IND-CATA

- Non-Sensitive Information:

No access Token -> No non-sensitive information



Security Notions

IND-CATA

- Non-Sensitive Information:

  No access Token -> No non-sensitive information

Game IND-CATA

$b \xleftarrow{\$} \{0,1\}$

$PP \xleftarrow{\$} \mathsf{Param}(1^\lambda)$

$(state, m_0 = (m_{s,0}, m_{ns,0}), m_1 = (m_{s,1}, m_{ns,1})) \leftarrow \mathcal{A}^{\mathsf{Store}(PP,.),\mathsf{Access}(PP,.)}(PP, 1^\lambda)$

if $m_{ns,0} = m_{ns,1}$:

  return 0

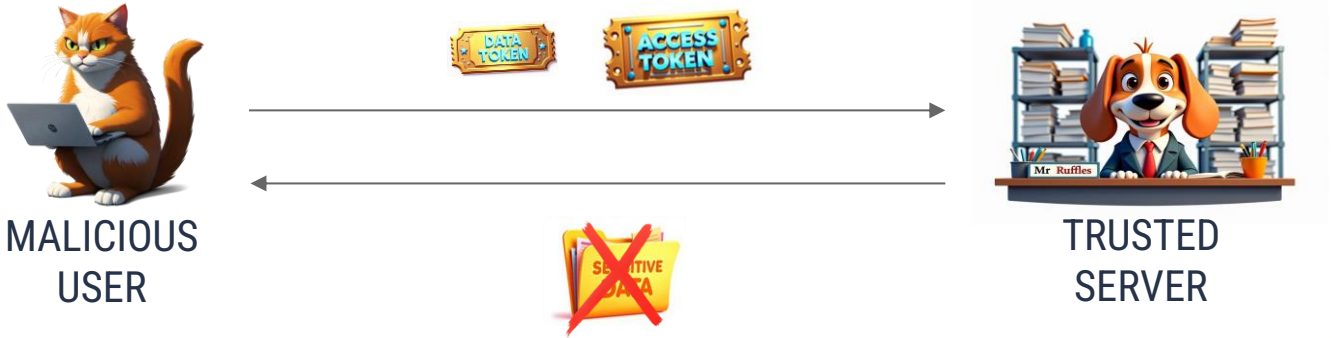
$(m\text{-}tok^*, data\text{-}tok^*) = \mathsf{Store}(\perp, PP, m_b)$

$access\text{-}tok^* = \mathcal{H}(10, m\text{-}tok^*)$

$b' \leftarrow \mathcal{A}^{\mathsf{Store}(PP,.),\mathsf{Access}(PP.)}(PP, access\text{-}tok^*, state)$

return $(b = b')$

Security Notions



- Sensitive Information:

  No master password -> No sensitive information

  (even with access token)

IND-SIA

Security Notions

Game IND-SIA

1 : $b \leftarrow\$ \{0,1\}$

2 : $PP \leftarrow\$ \mathsf{Param}(1^\lambda)$

3 : $(state, m_0 = (m_{s,0}, m_{ns,0}), m_1 = (m_{s,1}, m_{ns,1})) \leftarrow \mathcal{A}^{\mathsf{Store}(PP,.),\mathsf{Access}(PP,.)}(PP, 1^\lambda, DB)$

4 : $((m\text{-}tok^*, data\text{-}tok^*), DB') = \mathsf{Store}(\perp, PP, m_{ns,b}, DB)$

5 : $b' \leftarrow \mathcal{A}^{\mathsf{Store}(PP,.),\mathsf{Access}(PP.)}(PP, state, DB')$

6 : **return** $(b = b')$

IND-SIA

- Sensitive Information:

  No master password -> No sensitive information

  (even with access token)

3

BUILDING BLOCK: TOKENIZATION SCHEME

Tokenization Scheme - Model

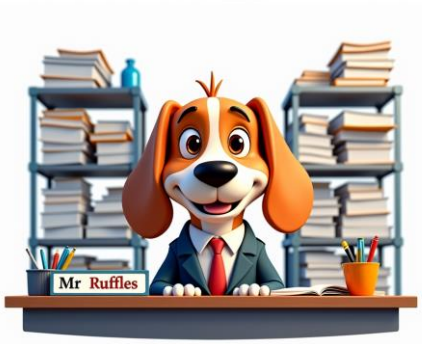
- Cloud Operating Model
  - Database may get compromised
- LLM-training friendly
  - Must be deterministic (same token for same data)
- The user should not have to store a secret key

Tokenization Scheme

USER

DATA

TOKEN



TRUSTED SERVER

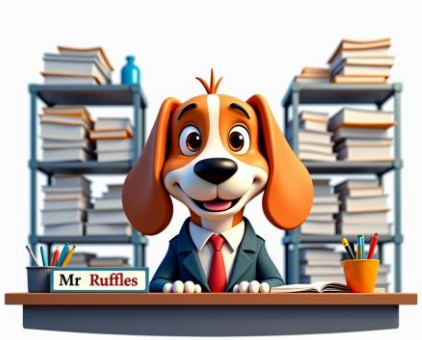
DATABASE

Tokenization Scheme

USER

TOKEN

DATA



TRUSTED SERVER

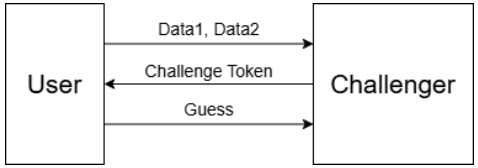
DATABASE

Tokenization Scheme

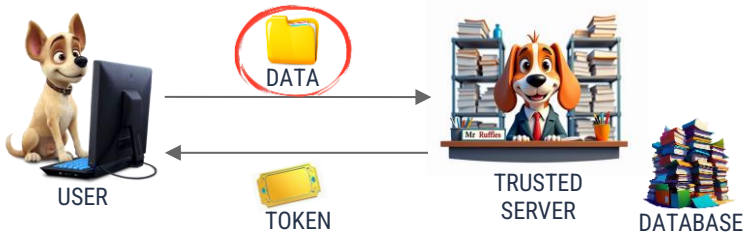
- Tokenization Procedure:
  - Takes message *m* as input
  - Generates a token *tok* and adds an entry to the database
  - Returns *tok*
- DeTokenization Procedure:
  - Takes token *tok* as input
  - If it exists in database, returns corresponding message m, else returns $\perp$.

Security Notions

IND-CDA



- **Chosen Data Attack**



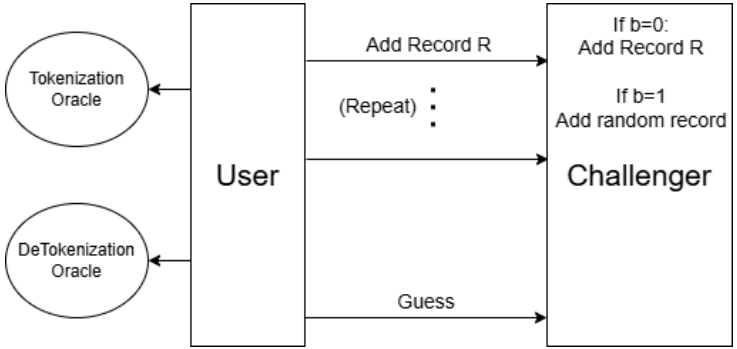
- Chosen Distribution Database Attack

IND-CDDA

Security Notions

IND-CDA

- Chosen Data Attack



- **Chosen Distribution Database Attack**

IND-CDDA

Our Construction

$\mathsf{Tok}(m, \overline{m}, k_1)$ // for basic construction $k_1 = \perp, m = \overline{m}$

1 : **if** $k_1 = \perp$
2 : $\quad k_1 \leftarrow \mathcal{H}_1(m)$
3 : **if** $\exists (c_1 || t_1 || c_2 || t_2) \in DB$ *s.t.* $t_1 = \mathcal{H}_2(m)$
4 : $\quad tok \leftarrow \mathsf{Dec}(k_1, c_1)$
5 : $\quad$ **return** tok
6 : $tok \leftarrow\!\!\$\ \{0,1\}^{\tau}$
7 : $c_1 \leftarrow \mathsf{Enc}(k_1, tok)$
8 : $t_1 \leftarrow \mathcal{H}_2(m)$
9 : $k_2 \leftarrow \mathcal{H}_3(tok)$
10 : $c_2 \leftarrow \mathsf{Enc}(k_2, \overline{m})$
11 : $t_2 \leftarrow \mathcal{H}_4(tok)$
12 : $d = c_1 || t_1 || c_2 || t_2$
13 : $DB = DB \cup \{d\}$
14 : **return** tok

$\mathsf{DeTok}(tok)$

1 : **if** $\exists (c_1 || t_1 || c_2 || t_2) \in DB$ *s.t.* $t_2 = \mathcal{H}_4(tok)$
2 : $\quad m \leftarrow \mathsf{Dec}(\mathcal{H}_3(tok), c_2)$
3 : $\quad$ **return** m
4 : **else return** $\perp$

token generated randomly,

c_1 = Enc(H(m), tok), t_1 = H'(m)
c_2 = Enc(H(tok), m), t_2 = H'(tok)

store {$c1$, $t1$, $c2$, $t2$} as one record

Our Construction

Tok$(m, \overline{m}, k_1)$ // for basic construction $k_1 = \perp, m = \overline{m}$

1 : **if** $k_1 = \perp$
2 : $\quad k_1 \leftarrow \mathcal{H}_1(m)$
3 : **if** $\exists (c_1 || t_1 || c_2 || t_2) \in DB \; s.t. \; t_1 = \mathcal{H}_2(m)$
4 : $\quad tok \leftarrow \mathsf{Dec}(k_1, c_1)$
5 : $\quad$ **return** tok
6 : $tok \leftarrow\!\!\$ \; \{0,1\}^\tau$
7 : $c_1 \leftarrow \mathsf{Enc}(k_1, tok)$
8 : $t_1 \leftarrow \mathcal{H}_2(m)$
9 : $k_2 \leftarrow \mathcal{H}_3(tok)$
10 : $c_2 \leftarrow \mathsf{Enc}(k_2, \overline{m})$
11 : $t_2 \leftarrow \mathcal{H}_4(tok)$
12 : $d = c_1 || t_1 || c_2 || t_2$
13 : $DB = DB \cup \{d\}$
14 : **return** tok

DeTok(tok)

1 : **if** $\exists (c_1 || t_1 || c_2 || t_2) \in DB \; s.t. \; t_2 = \mathcal{H}_4(tok)$
2 : $\quad m \leftarrow \mathsf{Dec}(\mathcal{H}_3(tok), c_2)$
3 : $\quad$ **return** m
4 : **else return** $\perp$

if a record exists with t_2 = H'(*tok*),

return m = Dec(H(*tok*), c_2)

Our construction is both IND-CDA and IND-CDDA secure

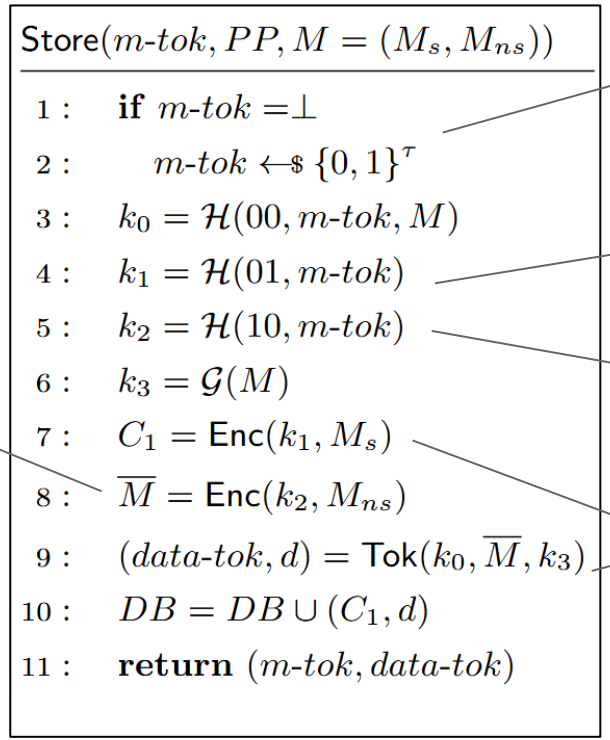
4

FULL VAULT SCHEME CONSTRUCTION

Vault Scheme Construction

- We use the Tokenization Scheme as a building block
- For simplicity of presentation, assume that the message space is same as the token space.

Vault Scheme Construction - STORE



if token does not exist beforehand, it is generated.

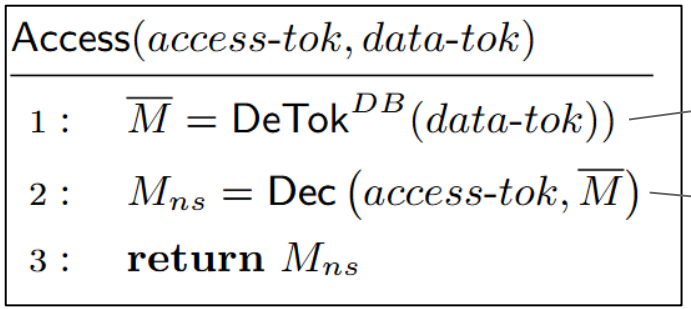
two hashes of the master token, one for sensitive, one for non-sensitive data

the one for the non-sensitive data is used as the access token

the non-sensitive data ciphertext is tokenized and that token is used as the data-token

n-s data is encrypted with the access token as key

Vault Scheme Construction - ACCESS



access needs both data token and access token

data-token is the token for the tokenization scheme

access token is the key for the decryption

Vault Scheme Construction - RETRIEVE

Retrieve$(m\text{-}tok, data\text{-}tok)$

1 : $((C_1, d), \overline{M}) = \mathsf{DeTok}^{DB}(data\text{-}tok))$

2 : $k_1 = \mathcal{H}(0, m\text{-}tok)$

3 : $k_2 = \mathcal{H}(1, m\text{-}tok)$

4 : $M_{ns} = \mathsf{Dec}\left(k_2, \overline{M}\right)$

5 : $M_s = \mathsf{Dec}\left(k_1, C_1\right)$

6 : **return** $M = (M_s, M_{ns})$

7 : **else return** $\perp$

retrieve needs both master password and access token

data-token is the token for the tokenization scheme

derive the access token and use it to access the non-sensitive data

use master password to decrypt sensitive data

Security

- The IND-CATA of the vault construction follows from -
  - Randomness of the hash function
  - IND-CDA and IND-CDDA security of the underlying Tokenization scheme.
- The IND-SIA follows from -
  - randomness of the hash functions
  - IND-CPA security of encryption scheme
  - IND-CDDA of tokenization scheme

Thank You!