Token generation & security

beys
The wise side of data

supervised by Marius Lombard-Platet

Agathe Blaise[1], Paul Huynh[2], Diane Leblanc-Albarel[3], Simon Masson[4], Cyrius Nugier[5], Brice Wandji[6]

[1] Thales & Sorbonne Université - Paris, France
[2] Université de Lorraine, INRIA, Loria, CNRS - Nancy, France
[3] CNRS, Irisa - Rennes, France
[4] Thales & Université de Lorraine, INRIA, Loria, CNRS - Nancy, France
[5] Université de Toulouse, LAAS - CNRS, Toulouse, France
[6] Fime - ENSICAEN, GREYC, Caen, France

REDOCS

441 billion

purchase transactions on payment cards worldwide in 2019

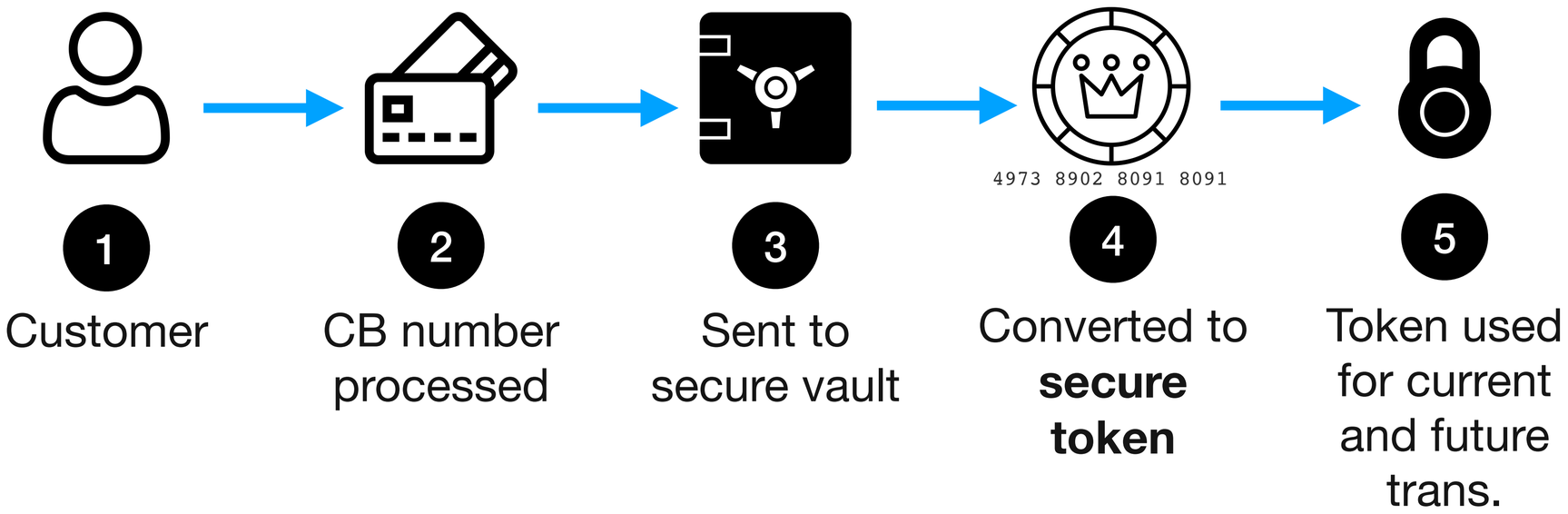
441 billion purchase transactions on payment cards worldwide in 2019

Secure storing not possible for online payment: risk of **data leak**

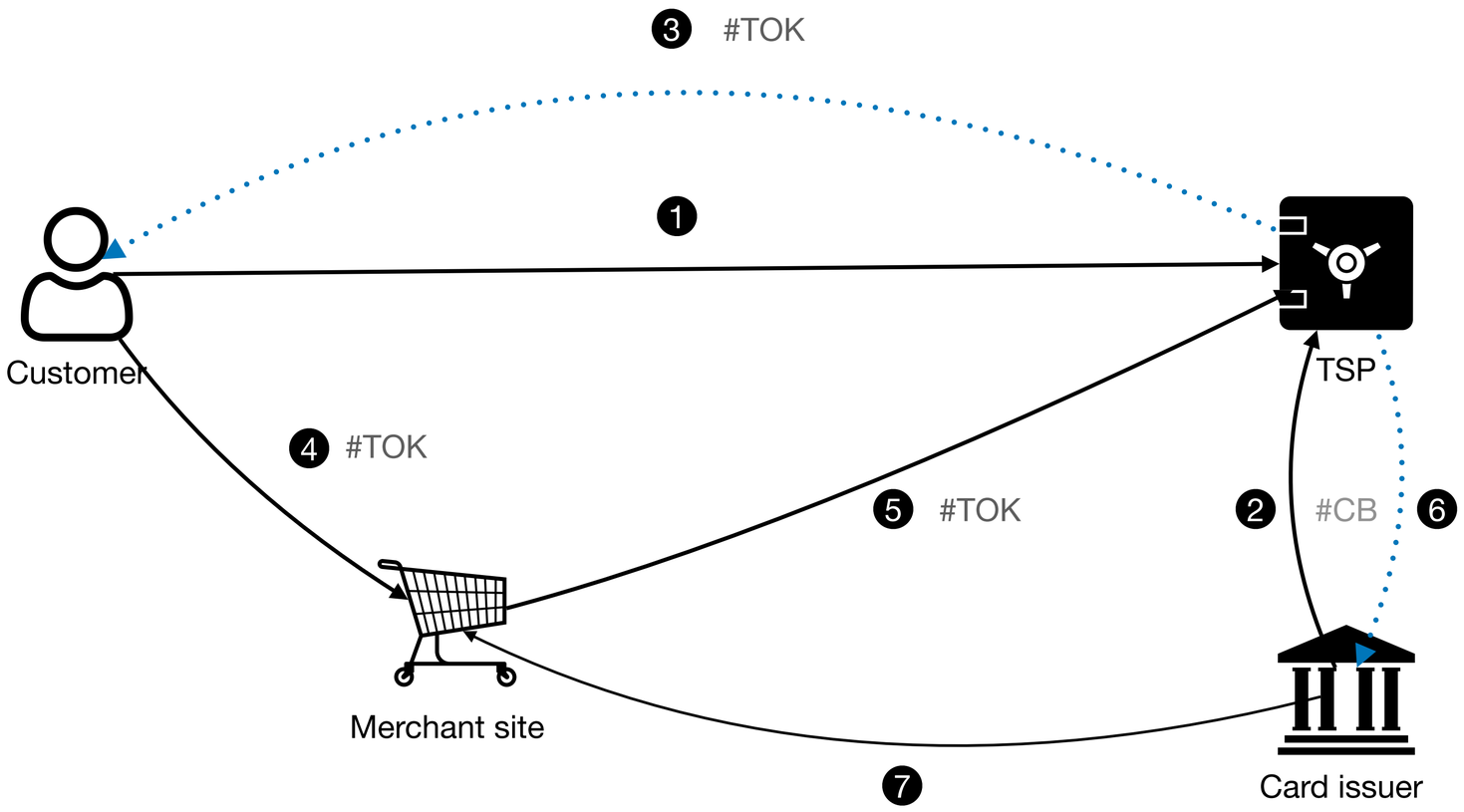
→ Need for an **alternative format** to avoid storing credit card numbers

Card number tokenization

- Credit card number $\xrightarrow{\text{tokenization}}$ string of random digits
- **(De)tokenization = Token Service Provider (TSP)**



Use case



1 Token request

2 Data retrieval

3 Token emission

4 Payment attempt

5 Payment request

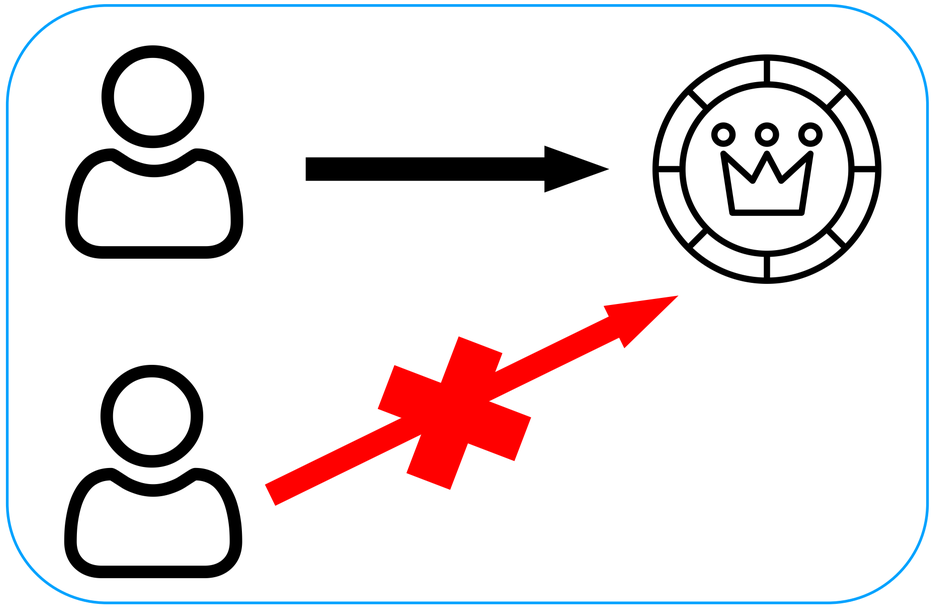
6 Detokenization

7 Payment

Time-sensitive op

Specifications

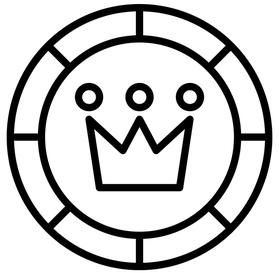
- **Unicity**

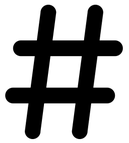


- Expiry time
- Format
- Identical distribution
- Unlinkability
- Timeframe
- Unforgeability

Specifications

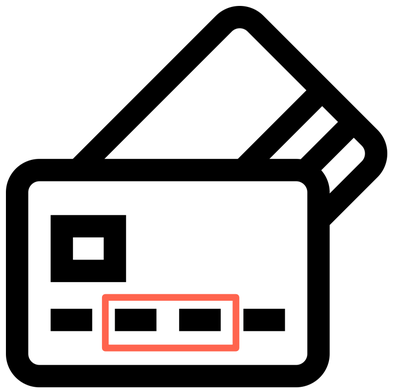
- Unicity
- **Expiry time**
- Format
- Identical distribution
- Unlinkability
- Timeframe
- Unforgeability



1.  Max. # of uses
2.  Expiry time

Specifications

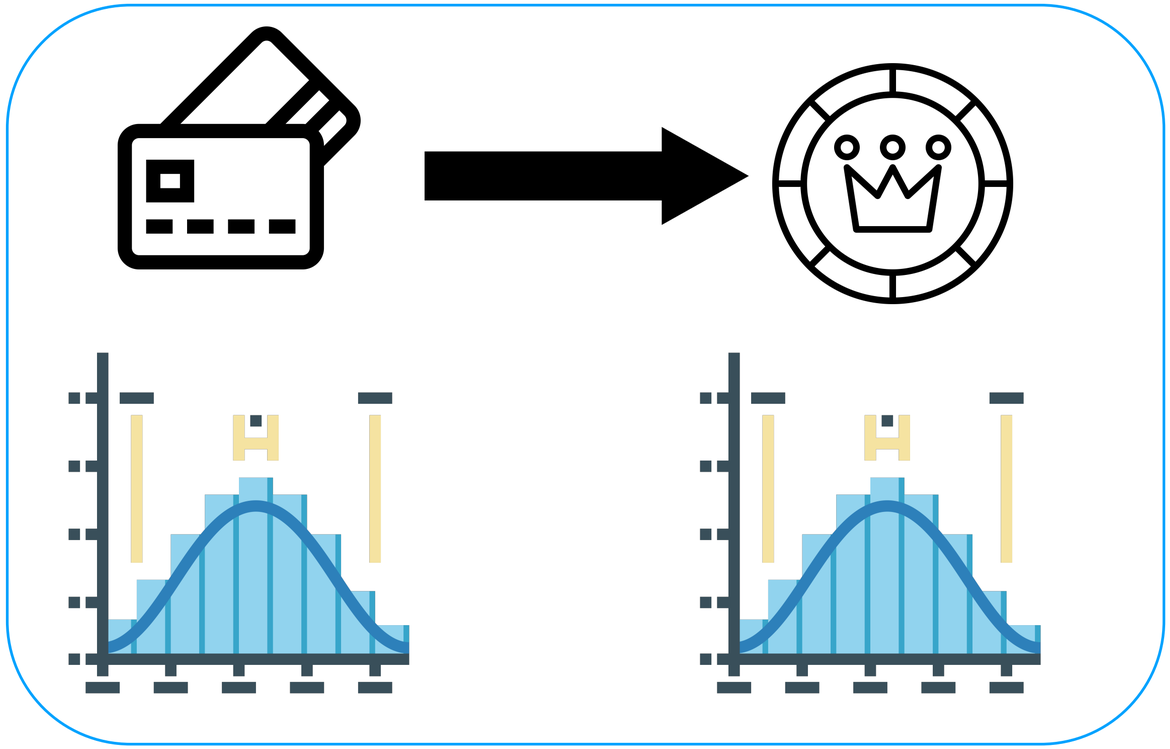
- Unicity
- Expiry time
- **Format**
- Identical distribution
- Unlinkability
- Timeframe
- Unforgeability



8-digit token

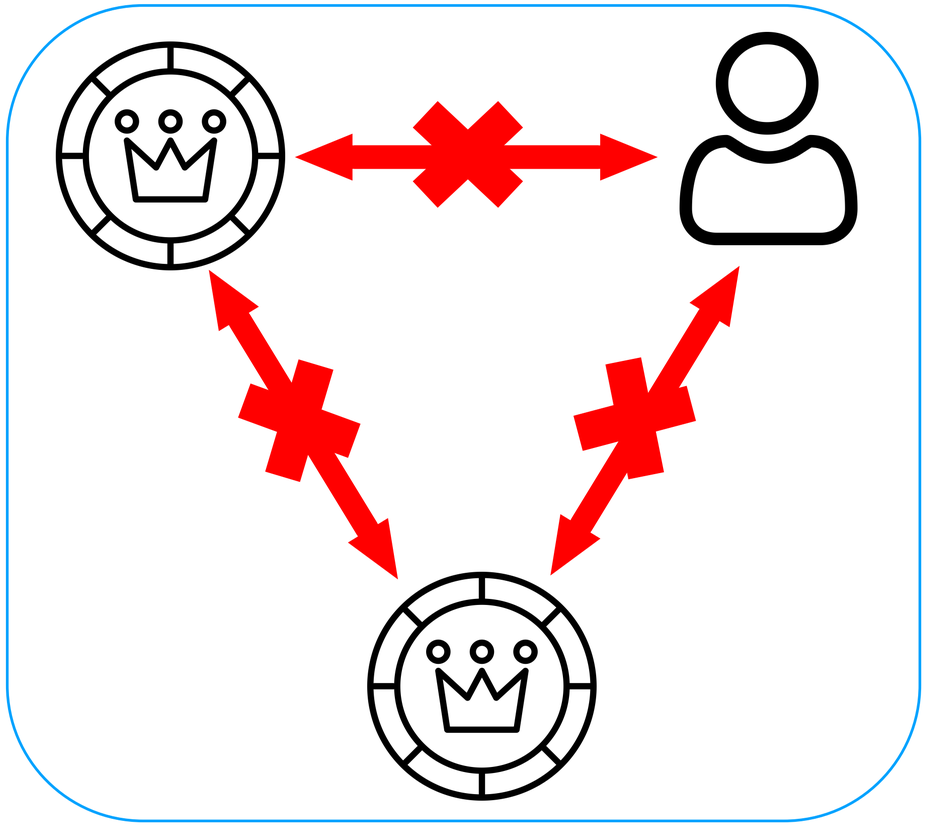
Specifications

- Unicity
- Expiry time
- Format
- **Identical distribution**
- Unlinkability
- Timeframe
- Unforgeability



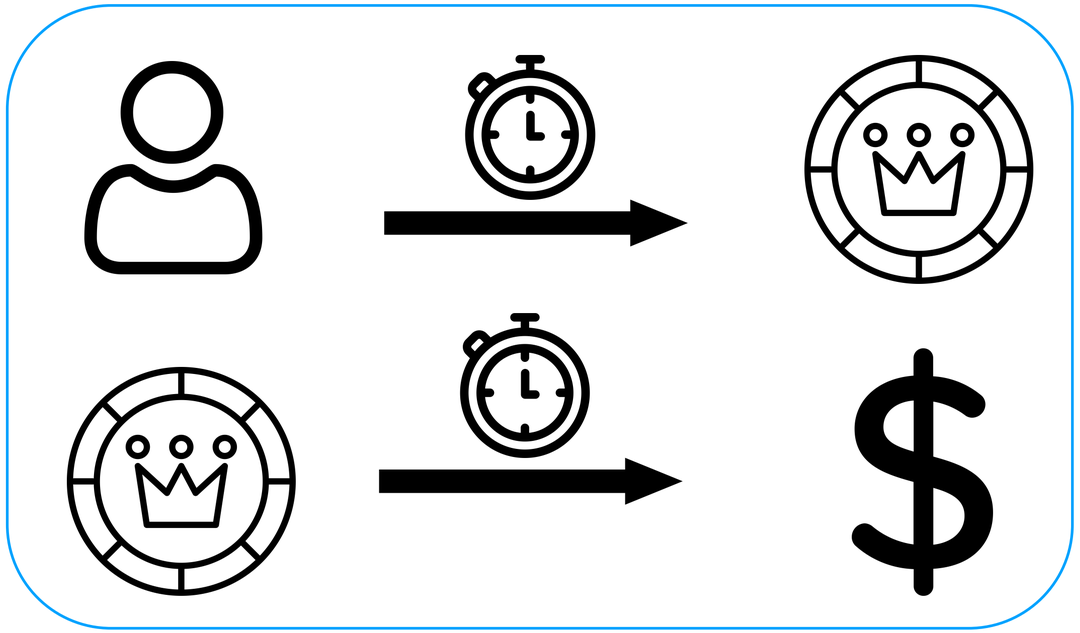
Specifications

- Unicity
- Expiry time
- Format
- Identical distribution
- **Unlinkability**
- Timeframe
- Unforgeability



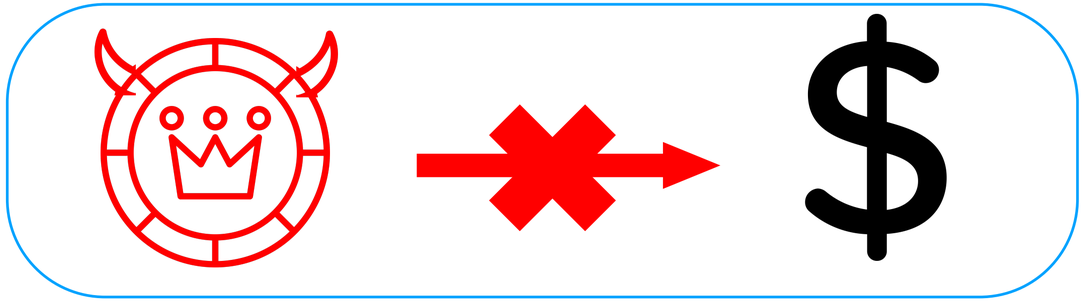
Specifications

- Unicity
- Expiry time
- Format
- Identical distribution
- Unlinkability
- **Timeframe (100ms)**
- Unforgeability



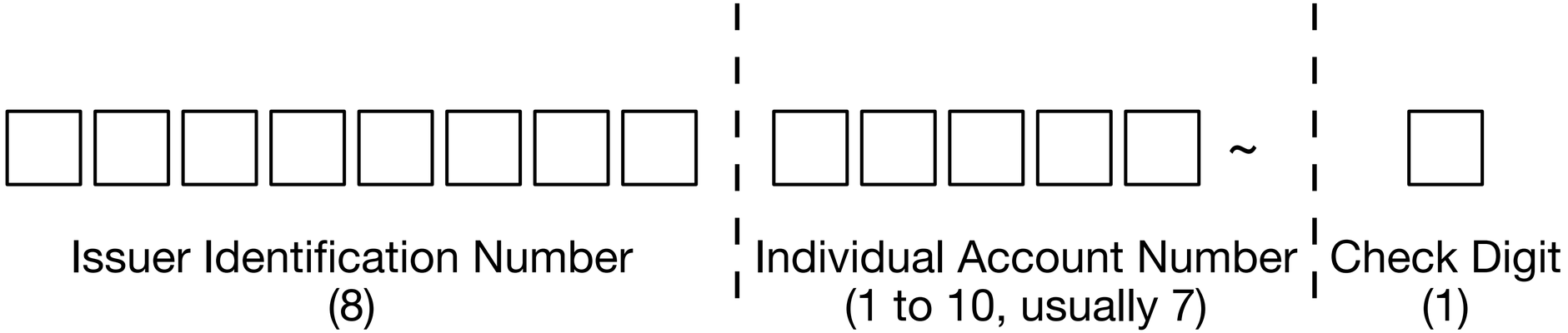
Specifications

- Unicity
- Expiry time
- Format
- Identical distribution
- Unlinkability
- Timeframe
- **Unforgeability**

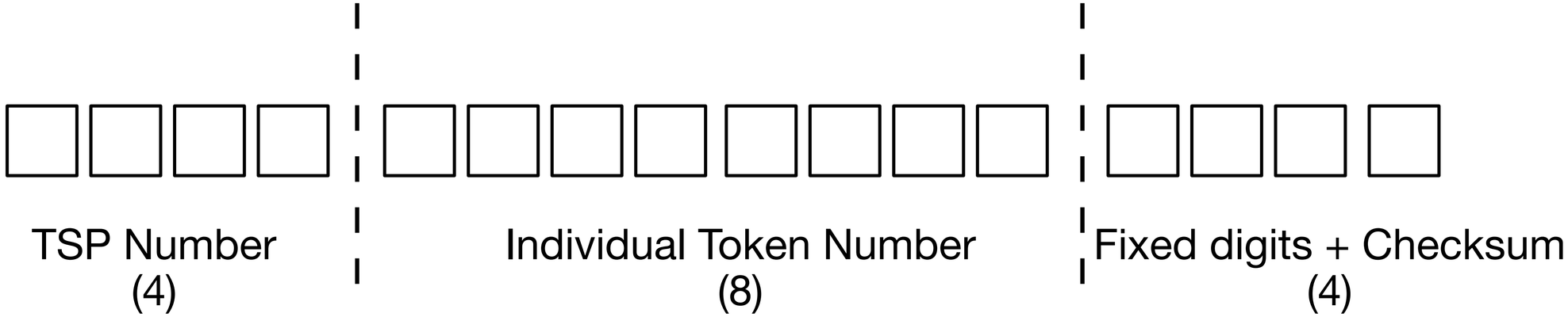


Primary Account Number

[ISO/IEC 7812]



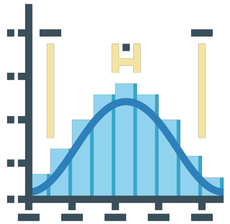
Token format



Main goals

- Quick detection of **collisions** during token generation

- **Uniformity** of the generated tokens

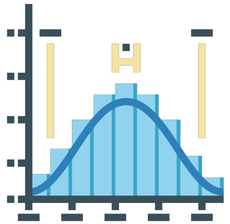


- **Untraceability** of the different tokens generated for the same user

Main goals

- Quick detection of **collisions** during token generation ✓

- **Uniformity** of the generated tokens ✓



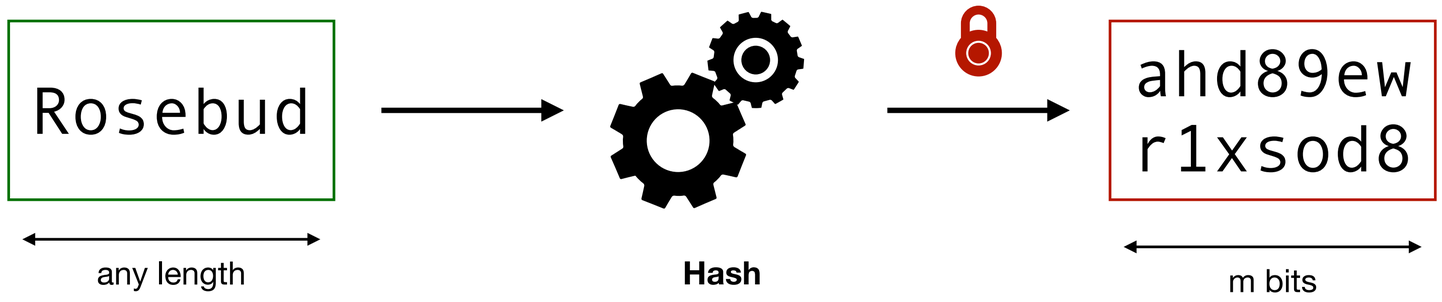
- **Untraceability** of the different tokens generated for the same user ✓

Outline

- Toolbox
- Inconclusive approaches
- Our Free-spot-tokenization approach
- Extra possibilities
- Demo
- Conclusion

Toolbox

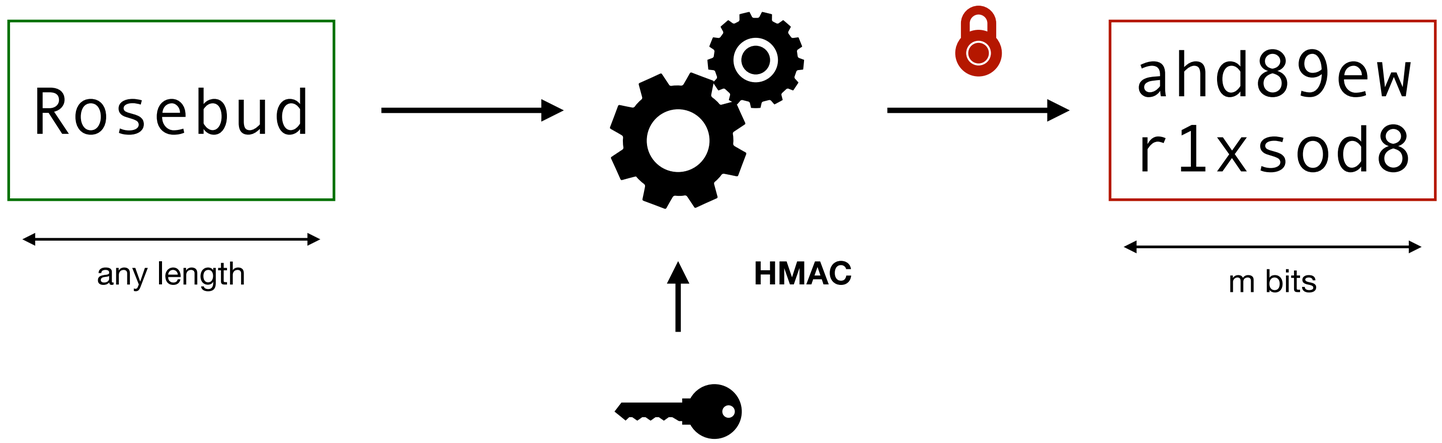
Hash function



- Collision-resistant
- Uniform output

Toolbox

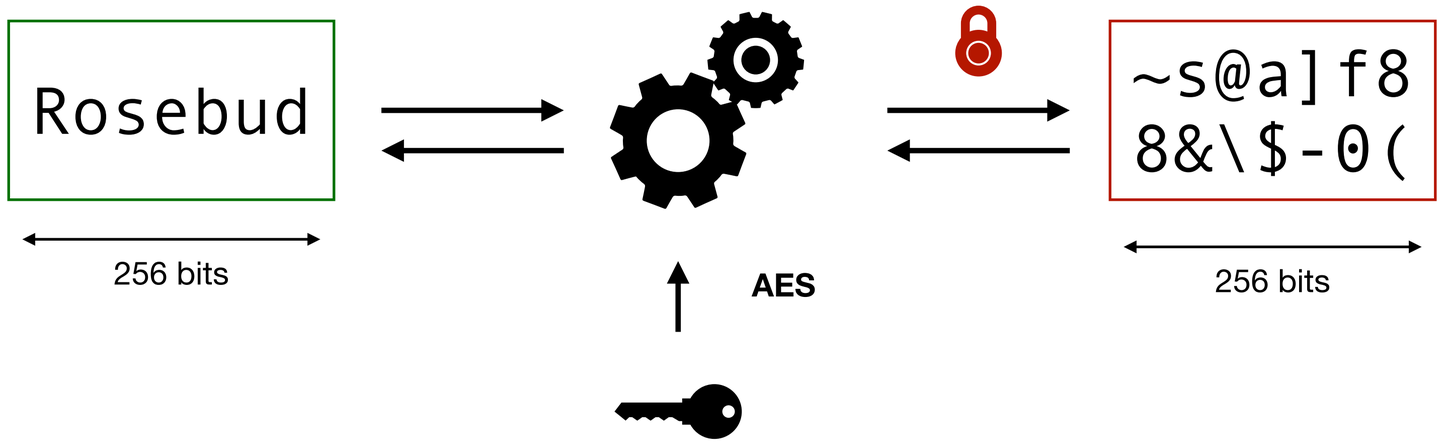
HMAC function



- Collision-resistant
- Uniform output

Toolbox

Advanced Encryption Standard-256 (AES-256)

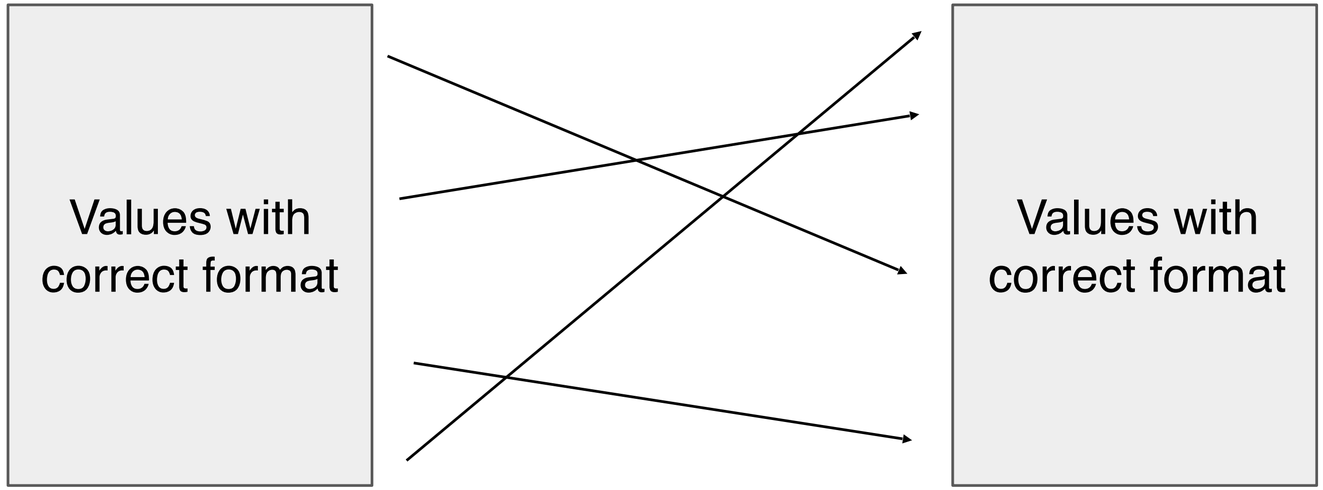


Inconclusive approaches

- **Format-Preserving Encryption (FPE)**
- **Blockchain**

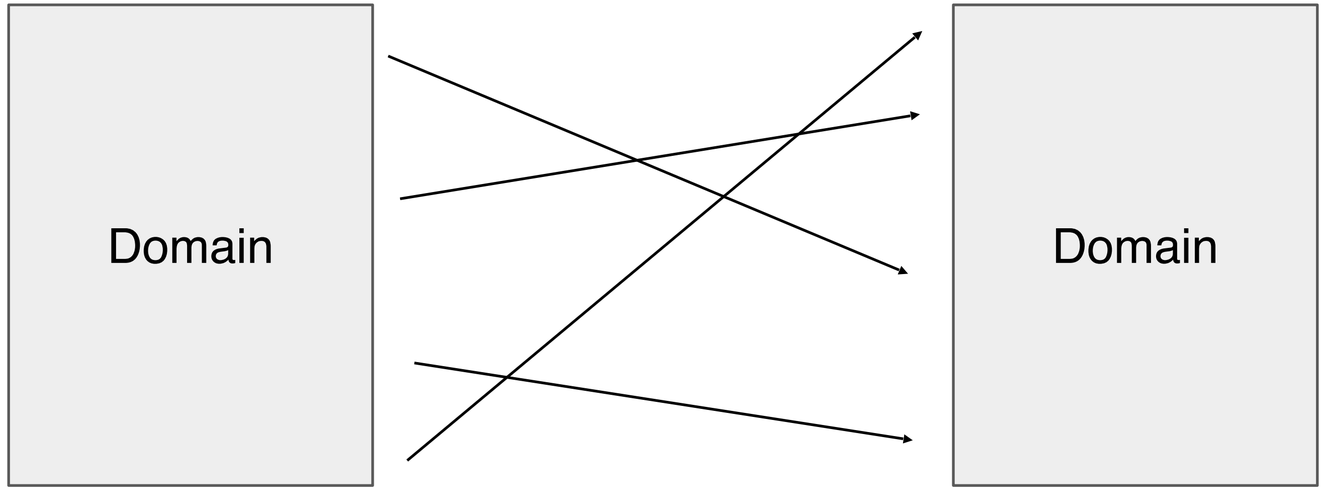
Inconclusive approaches

Format-Preserving Encryption



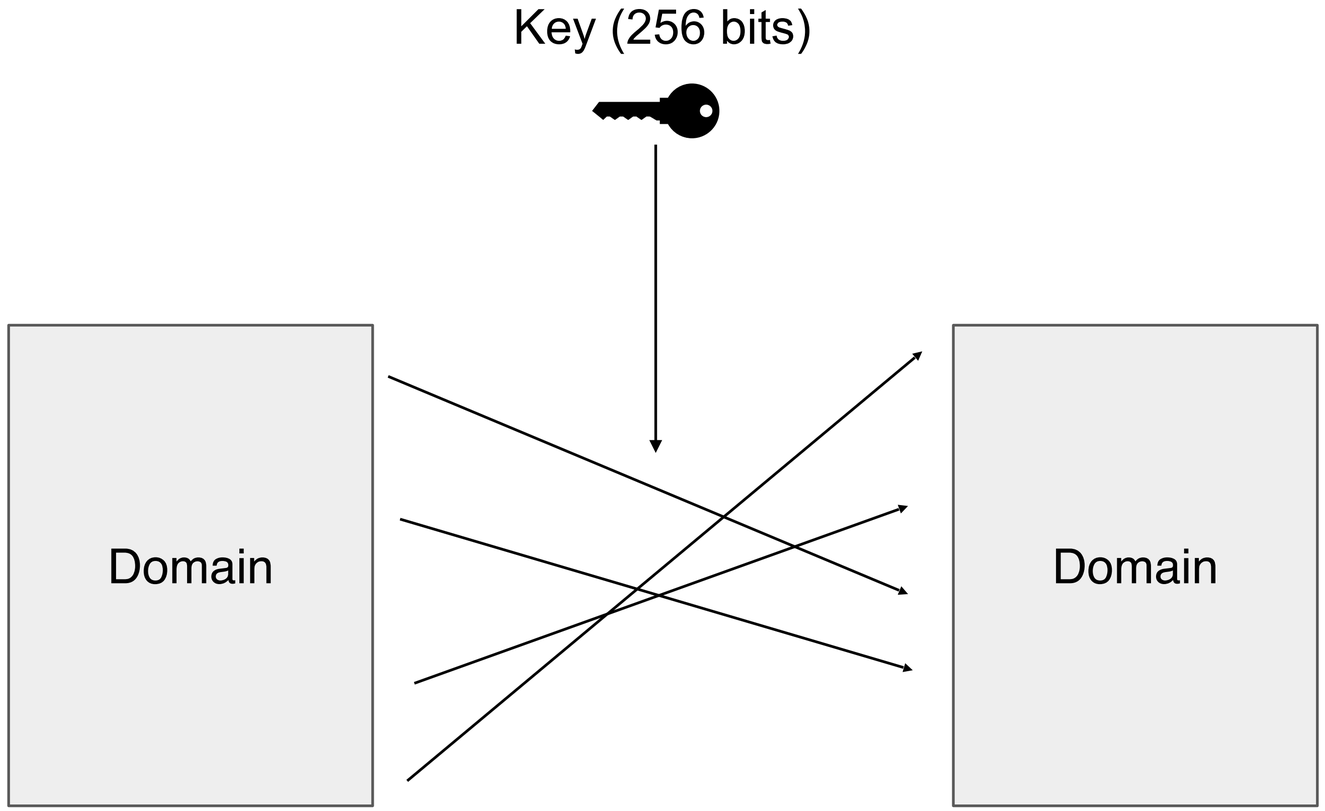
Inconclusive approaches

Format-Preserving Encryption



Inconclusive approaches

Format-Preserving Encryption

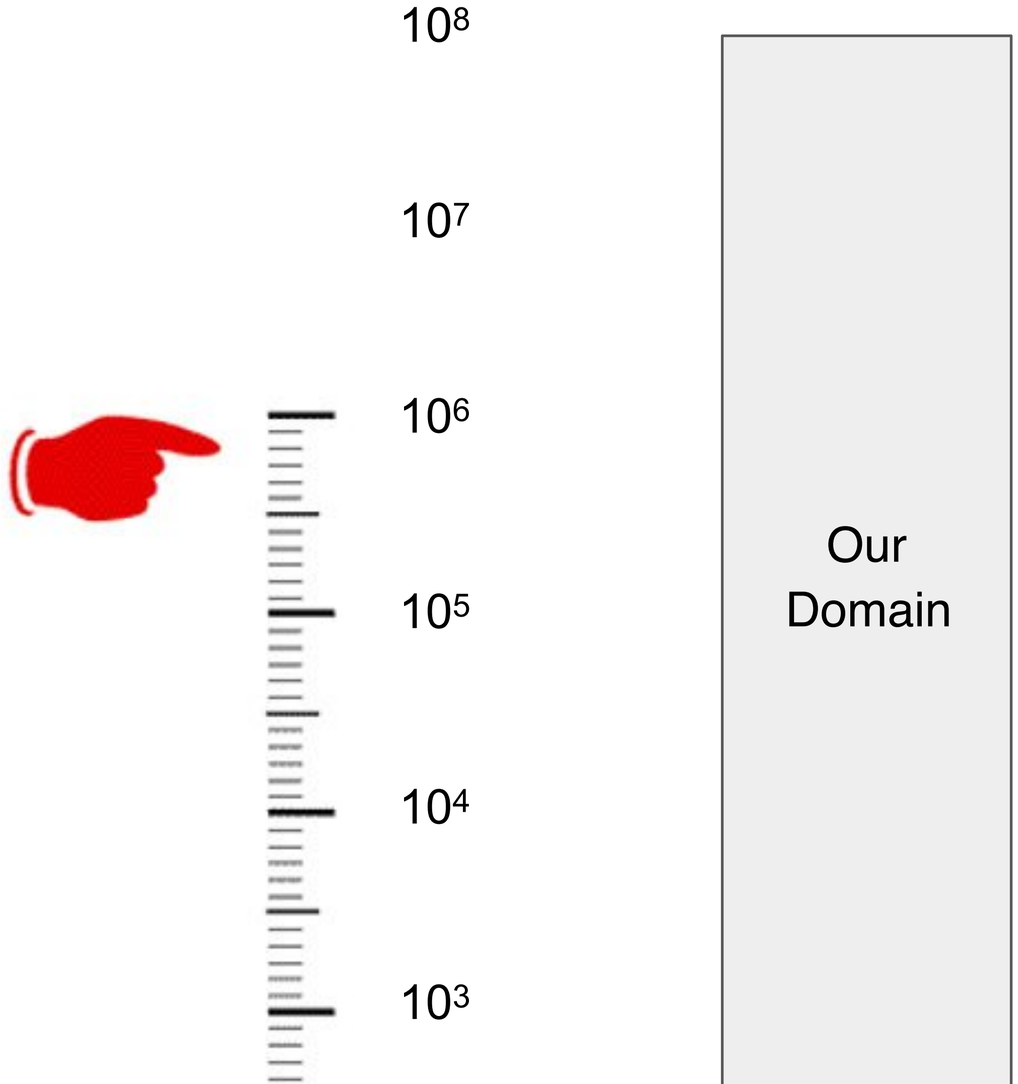


Inconclusive approaches

Format-Preserving Encryption

NIST

Your domain should be this big to be secure



Inconclusive approaches

Format-Preserving Encryption

NIST

Your domain should be this big to be secure

10^8

10^7

10^6

10^5

10^4

10^3

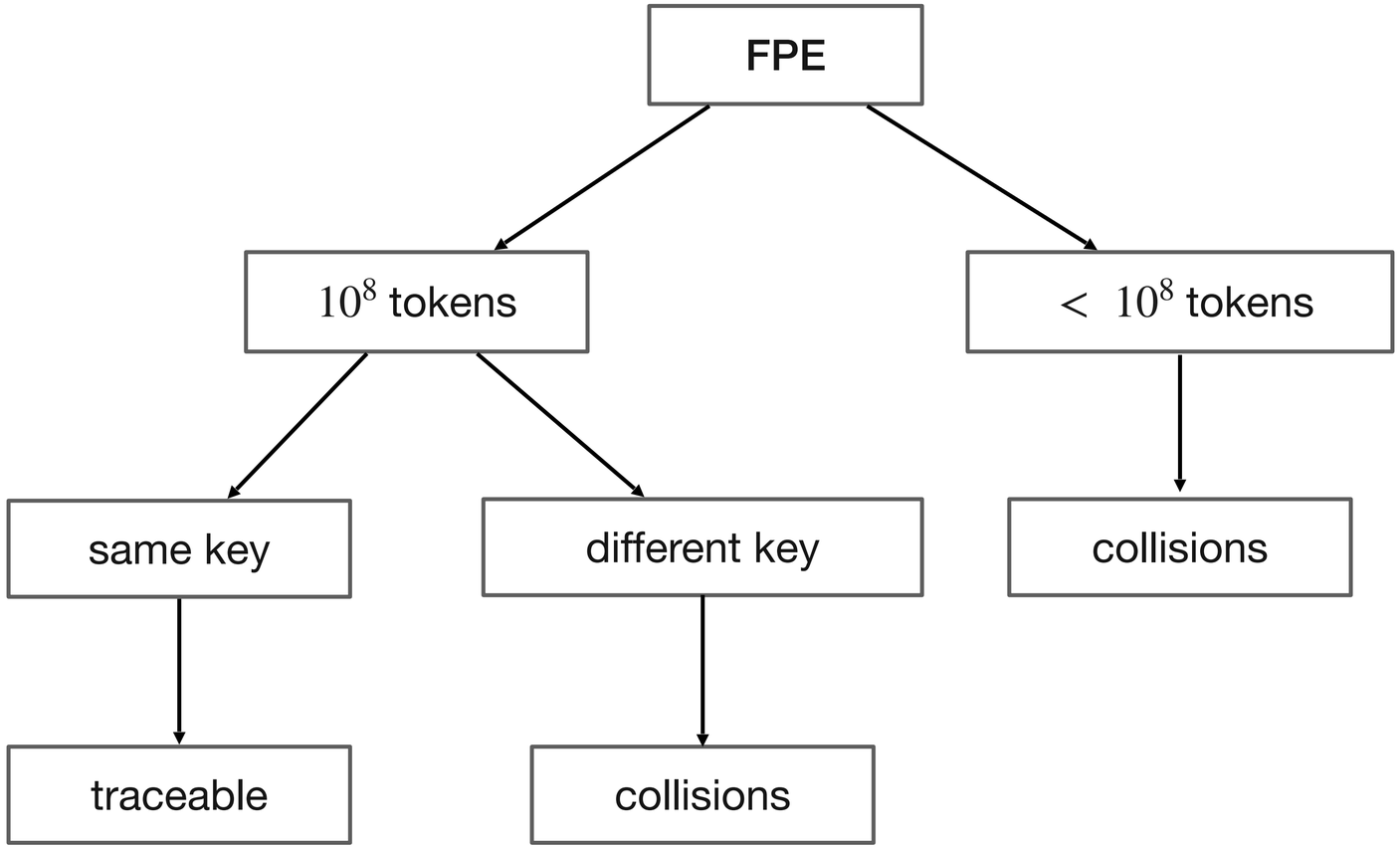
Our Domain

Soon ?

2019

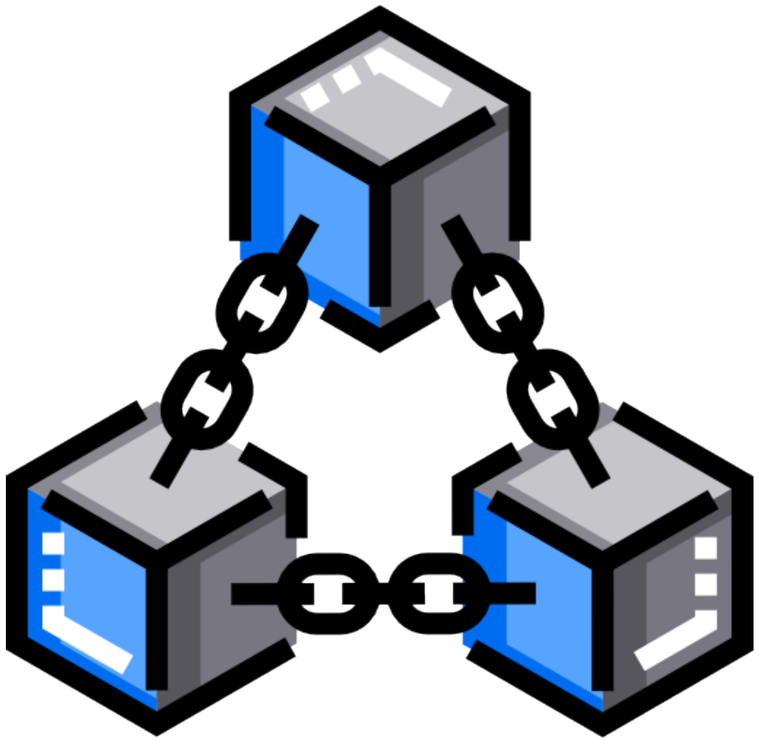
Inconclusive approaches

Format-Preserving Encryption



Inconclusive approaches

Blockchain in Tokenization Systems



- Distributed systems
- Trust
- Data integrity
- Transactions

Inconclusive approaches

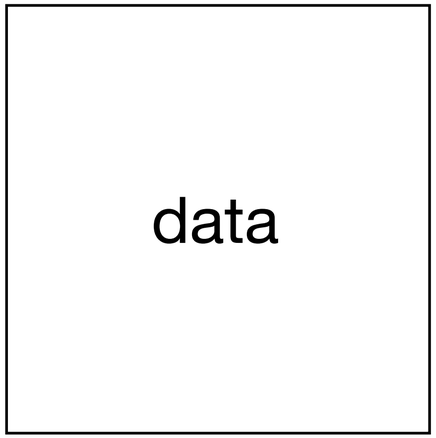
Blockchain in Tokenization Systems

Our Free-Spot-Tokenization approach

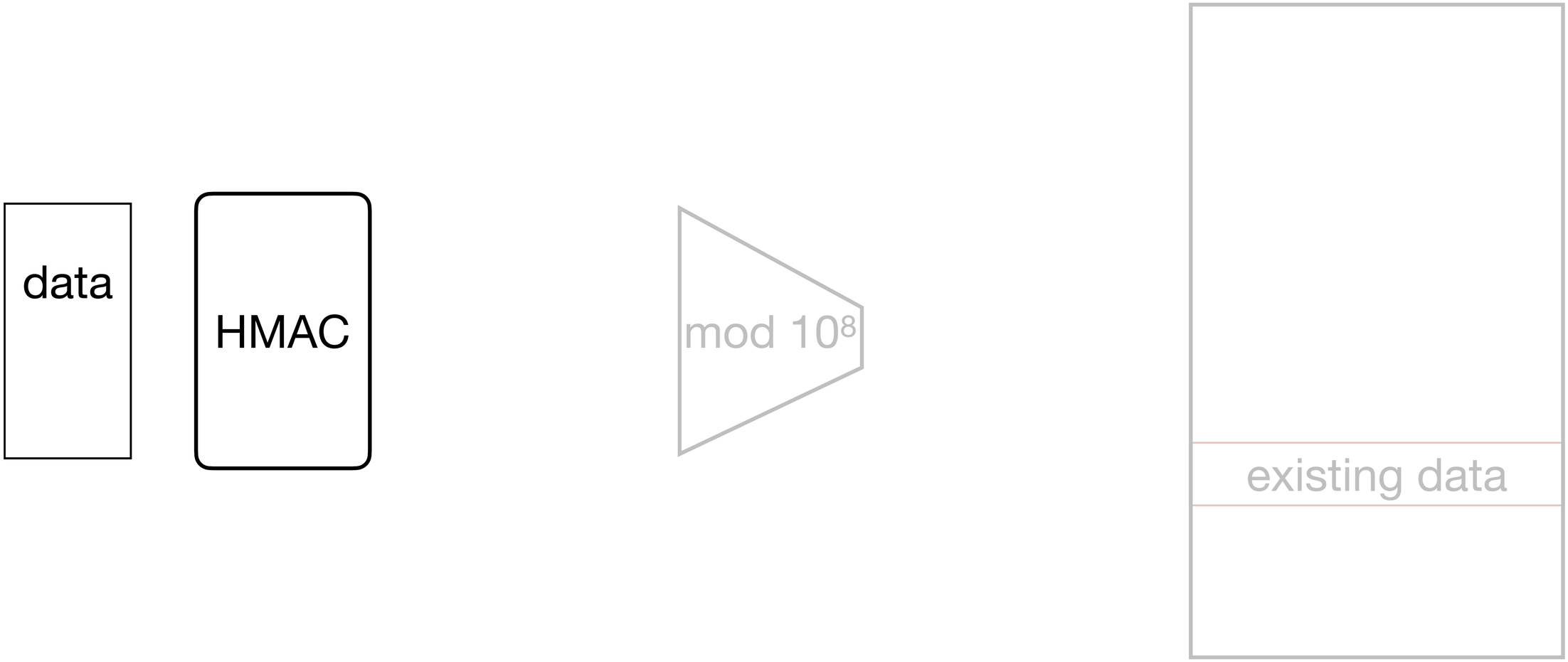
Tokenization

ID
name
card number
conditions

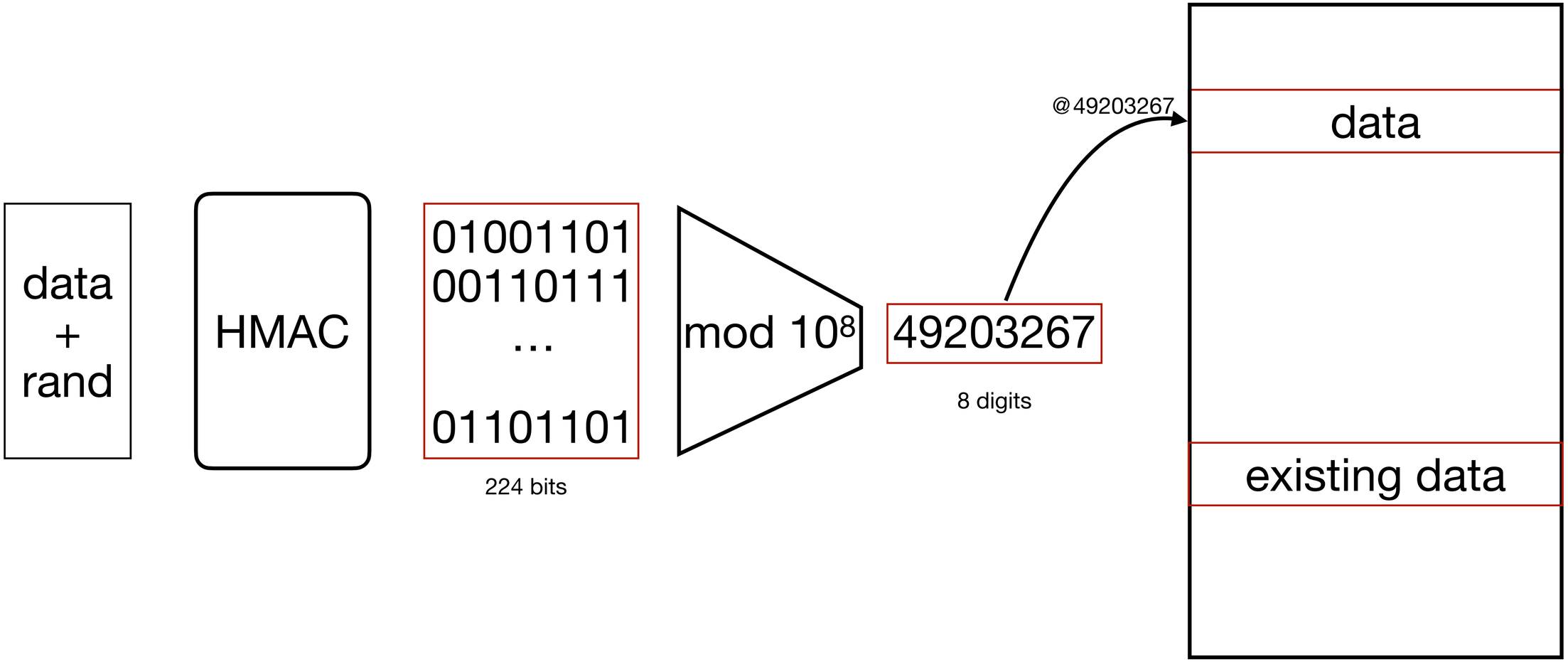
Tokenization



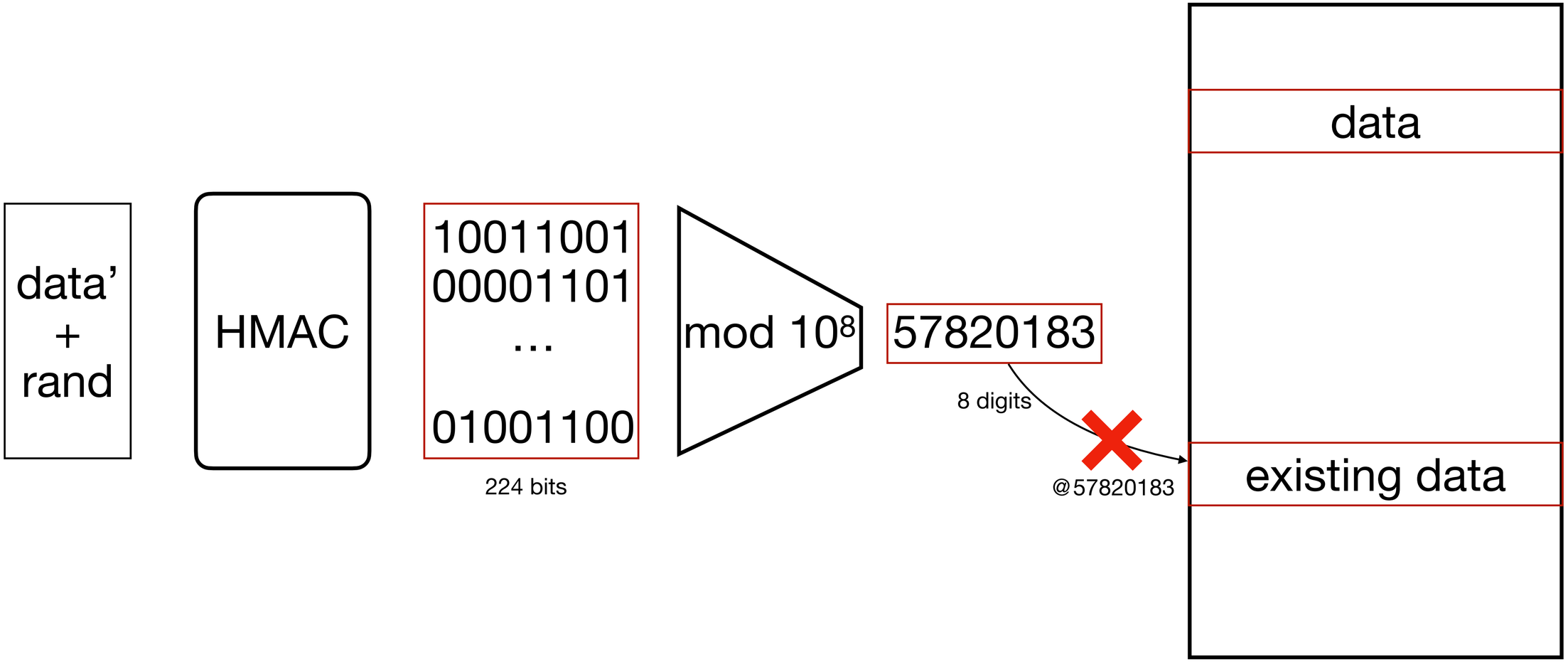
Tokenization



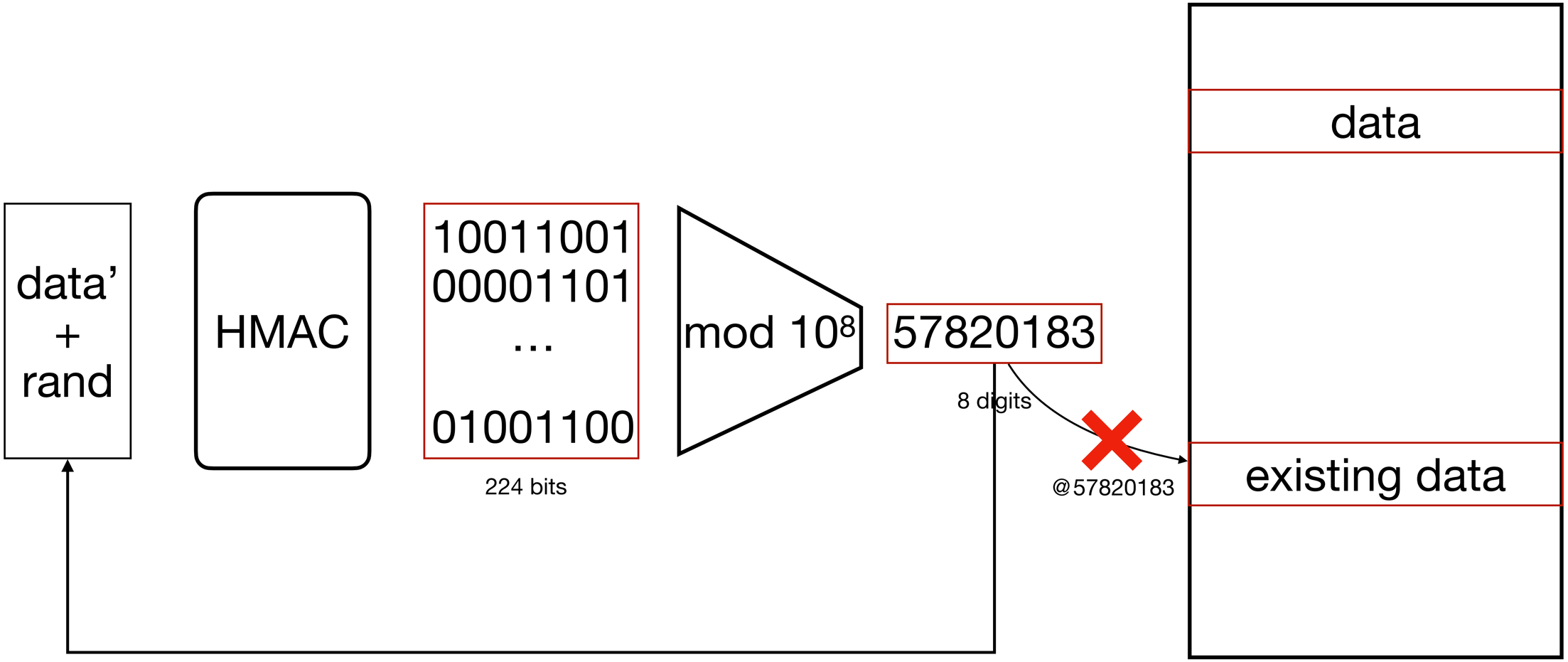
Tokenization



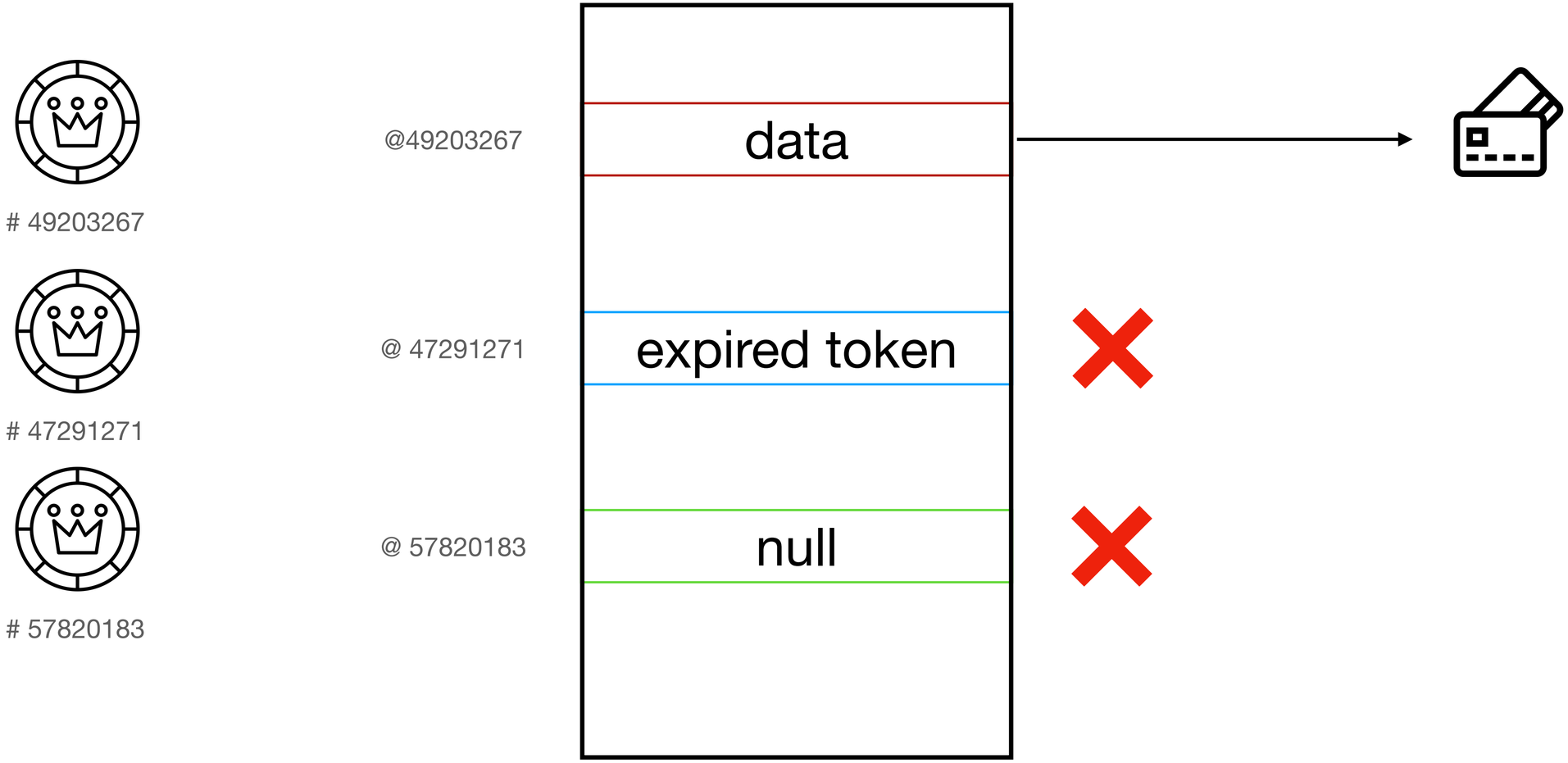
Tokenization



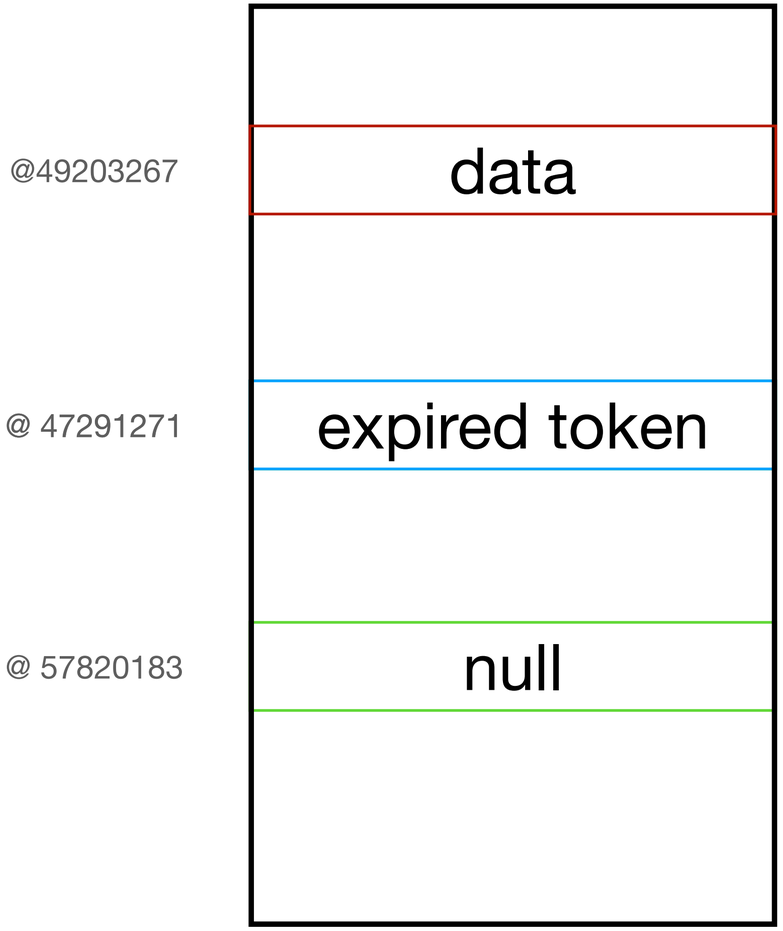
Tokenization



Detokenization

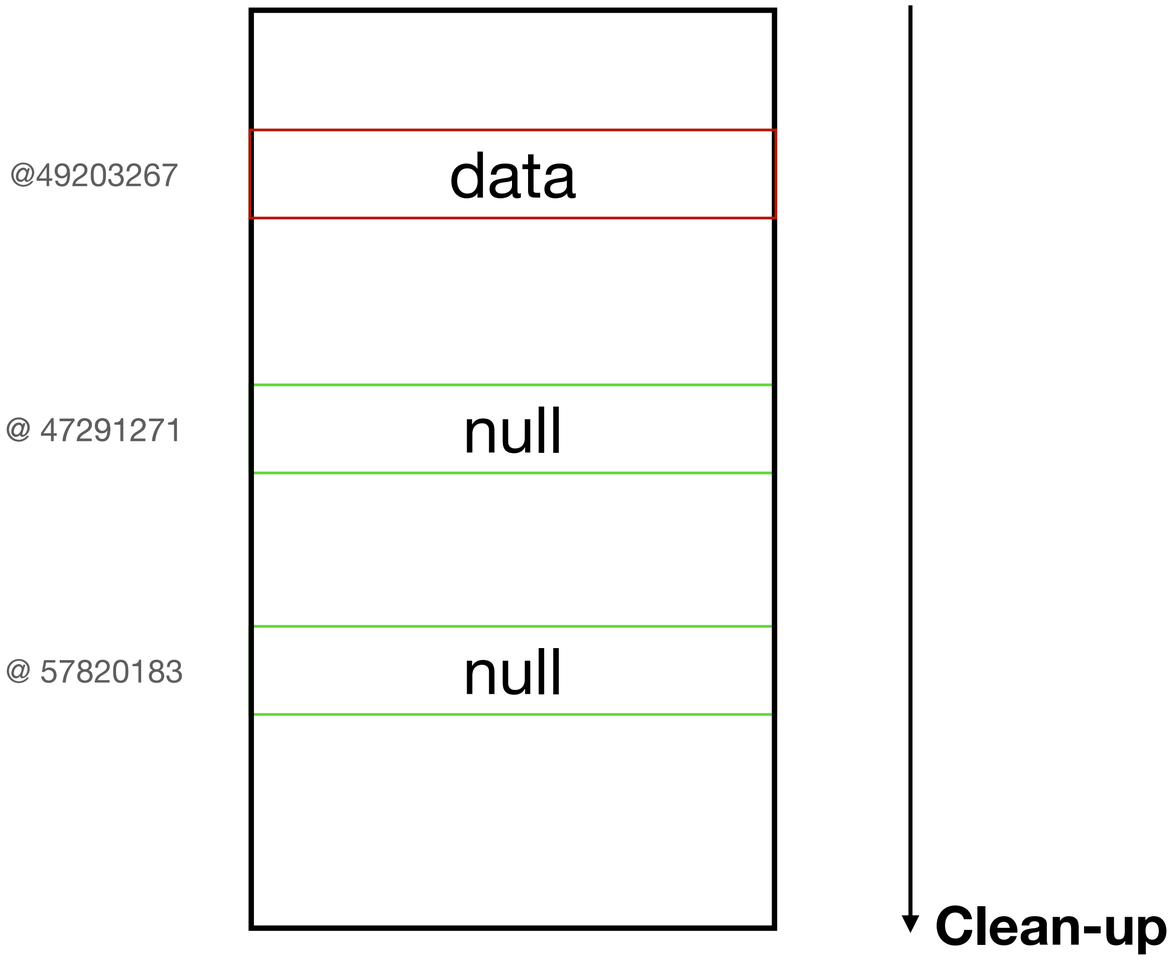


Clean table



Clean-up

Clean table



Properties

Condition to generate a valid token within a given timeframe

n number of tokens in circulation

n_{max} total token space size

T #trials per timeframe (100 ms)

λ security parameter

$$\boxed{Pr(\text{token generation fails}) < \frac{1}{2^{\lambda}}}$$

Under token uniformity assumption.

Condition to generate a valid token within a given timeframe

n number of tokens in circulation

n_{max} total token space size

T #trials per timeframe (100 ms)

λ security parameter

$$\boxed{\left(\frac{n}{n_{max}}\right)^{T} < \frac{1}{2^{\lambda}}}$$

Under token uniformity assumption.

Condition to generate a valid token within a given timeframe

n number of tokens in circulation

n_{max} total token space size

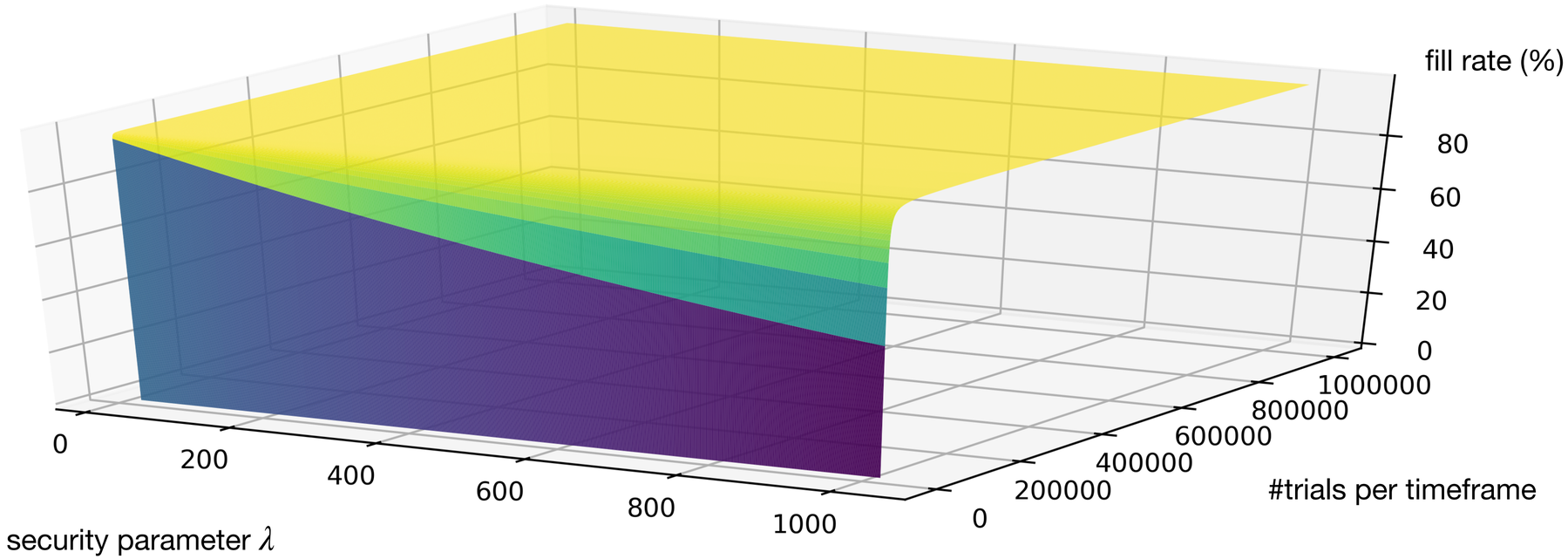
T #trials per timeframe (100 ms)

λ security parameter

$$\boxed{n < 2^{\log_2(n_{max}) - \frac{\lambda}{T}}}$$

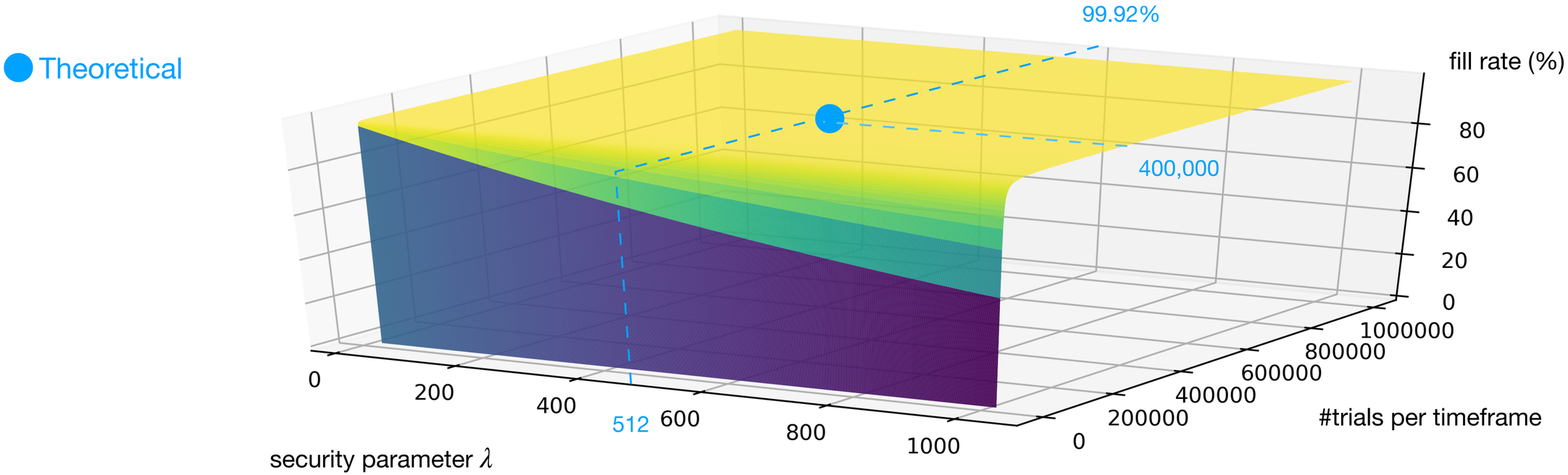
Under token uniformity assumption.

Condition to generate a valid token within a given timeframe



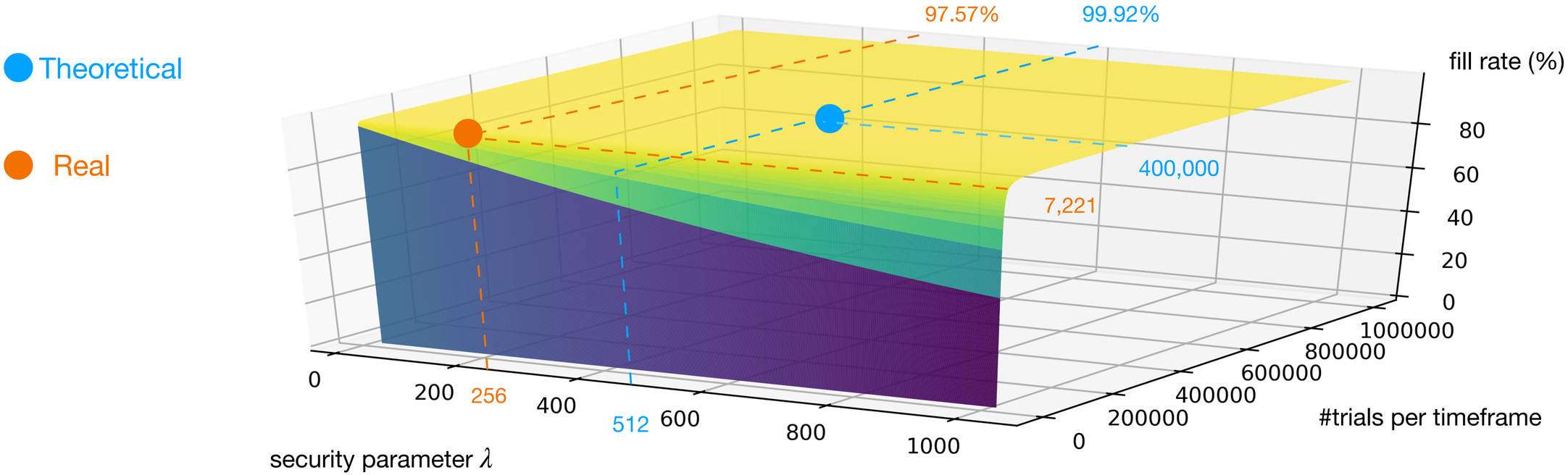
$$n = 2^{\log_2(n_{max}) - \frac{\lambda}{T}}$$

Condition to generate a valid token within a given timeframe



$$n = 2^{\log_2(n_{max}) - \frac{\lambda}{T}}$$

Condition to generate a valid token within a given timeframe



$$n = 2^{\log_2(n_{max}) - \frac{\lambda}{T}}$$

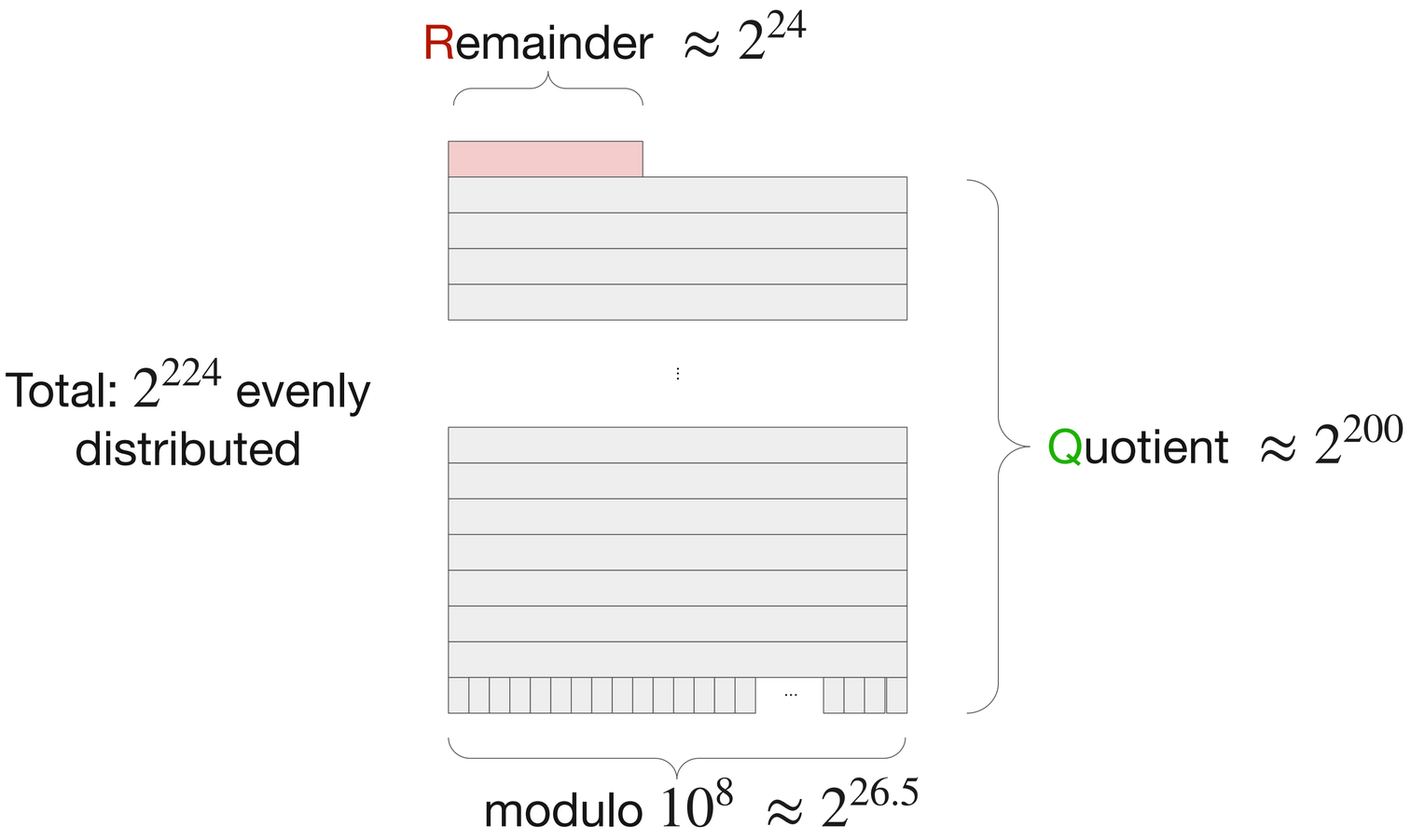
Proof of uniformity of the token generation

HMAC SHA3-224 is **uniform**

HMAC SHA3-224 mod 10^8 remains uniform.

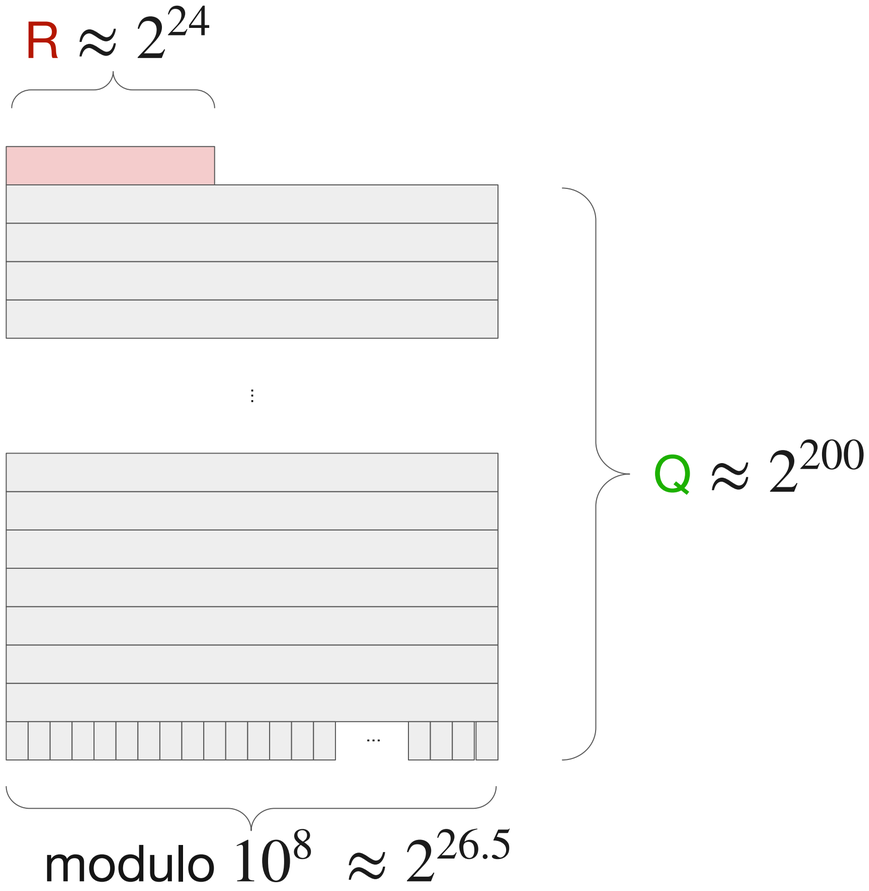
Proof of uniformity of the token generation

HMAC SHA3-224 is **uniform**



Proof of uniformity of the token generation

HMAC SHA3-224 is **uniform**



$$Pr(pathological) = Pr(Q \cdot 10^8 \leq x < 2^{224})$$

$$= \frac{R}{2^{224}}$$

$$= \frac{10{,}294{,}216}{2^{224}} \approx \frac{2^{23.3}}{2^{224}} \approx \frac{1}{2^{200}}$$

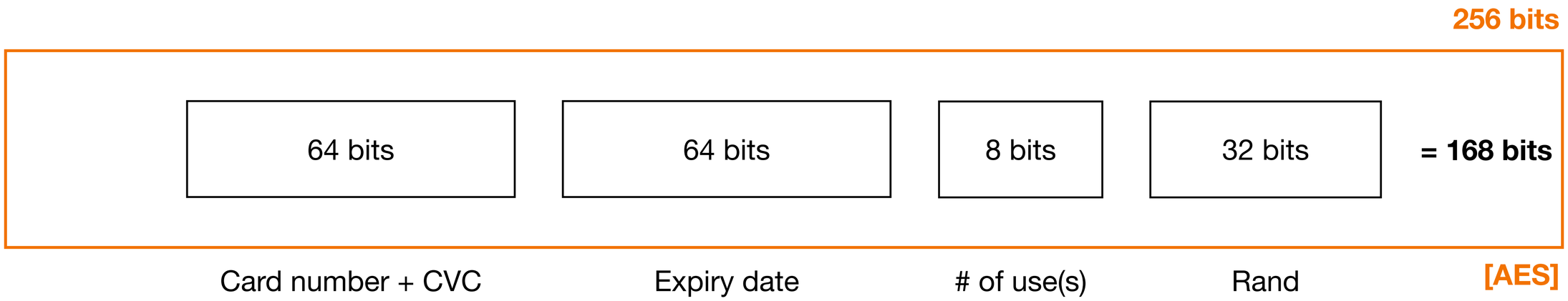
Performances

Expected memory usage

64 bits	64 bits	8 bits	32 bits	**= 168 bits**
Card number + CVC	Expiry date	# of use(s)	Rand	

Performances

Expected memory usage



Performances

	Theoretical performances
Fill rate before possible failure with probability over 2^{-512} and T=400,000	99.92%
Maximum RAM used without encryption	16.8 GB
Maximum RAM used with encryption	25.6 GB

→ 10M rows x 168 bits

Performances

	Theoretical performances
Fill rate before possible failure with probability over 2^{-512} and T=400,000	99.92%
Maximum RAM used without encryption	16.8 GB
Maximum RAM used with encryption	25.6 GB

→ 10M rows x 256 bits

Performances

	Real performances
Fill rate before failure	99.998%
Maximum RAM used without encryption	3.2 GB
Maximum RAM used with encryption	25.6 GB

Performances

	Real performances
Fill rate before failure	99.998%
Maximum RAM used without encryption	3.2 GB
Maximum RAM used with encryption	25.6 GB

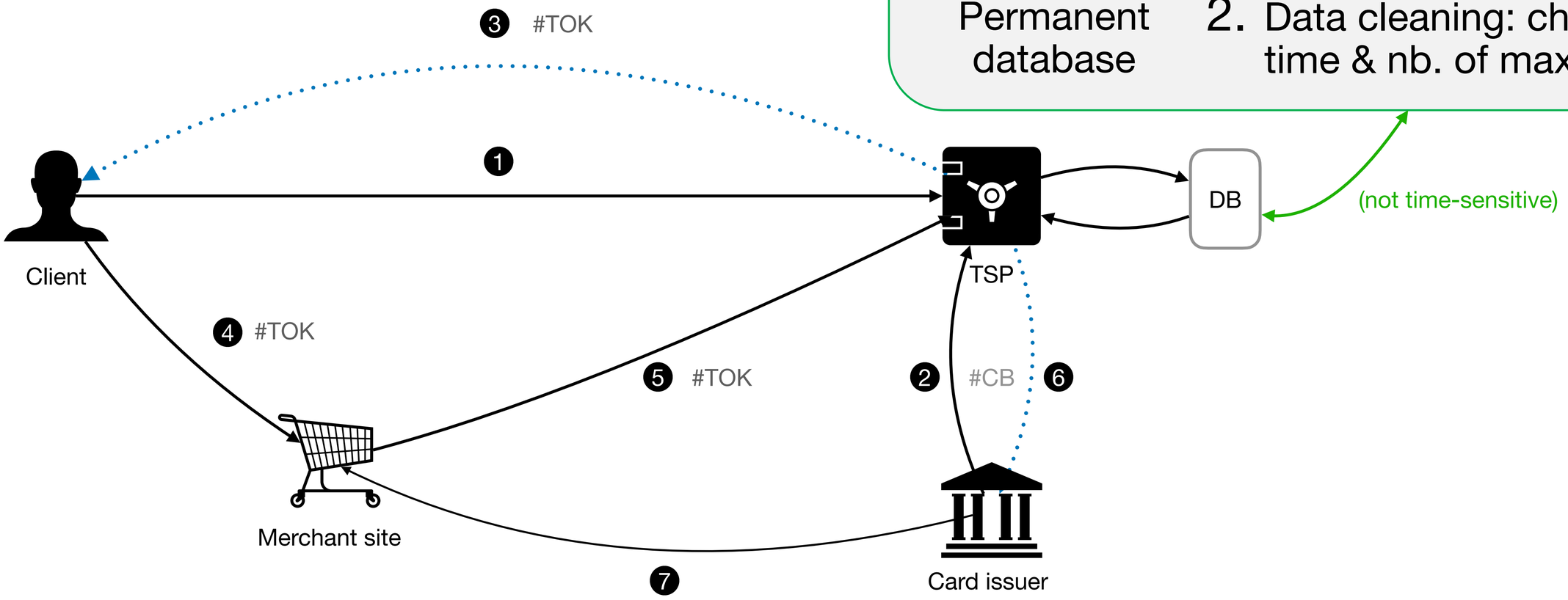
Average tokenization: 0.041ms

#trials / timeframe: 7221

```
struct row {
  unsigned long long cb;
  int r;
  unsigned char useNumber;
  unsigned long long t;
  short extra1;
  int extra2;
};
```

Extra possibilities

Database audit



Permanent database

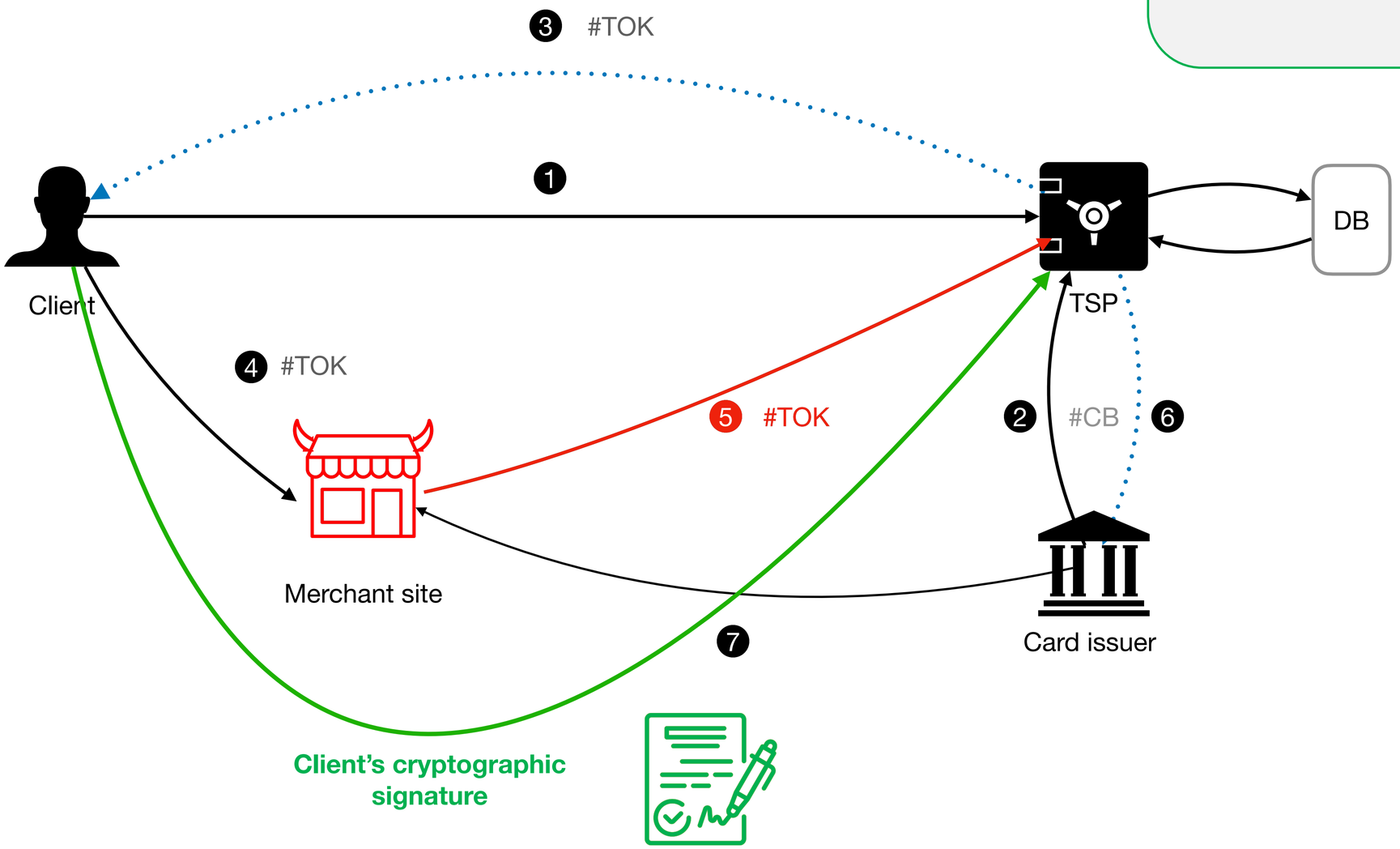
Data sustainability:

1. Logging every transaction
2. Data cleaning: check expiry time & nb. of max. uses

Unforgeability

Personal signature:

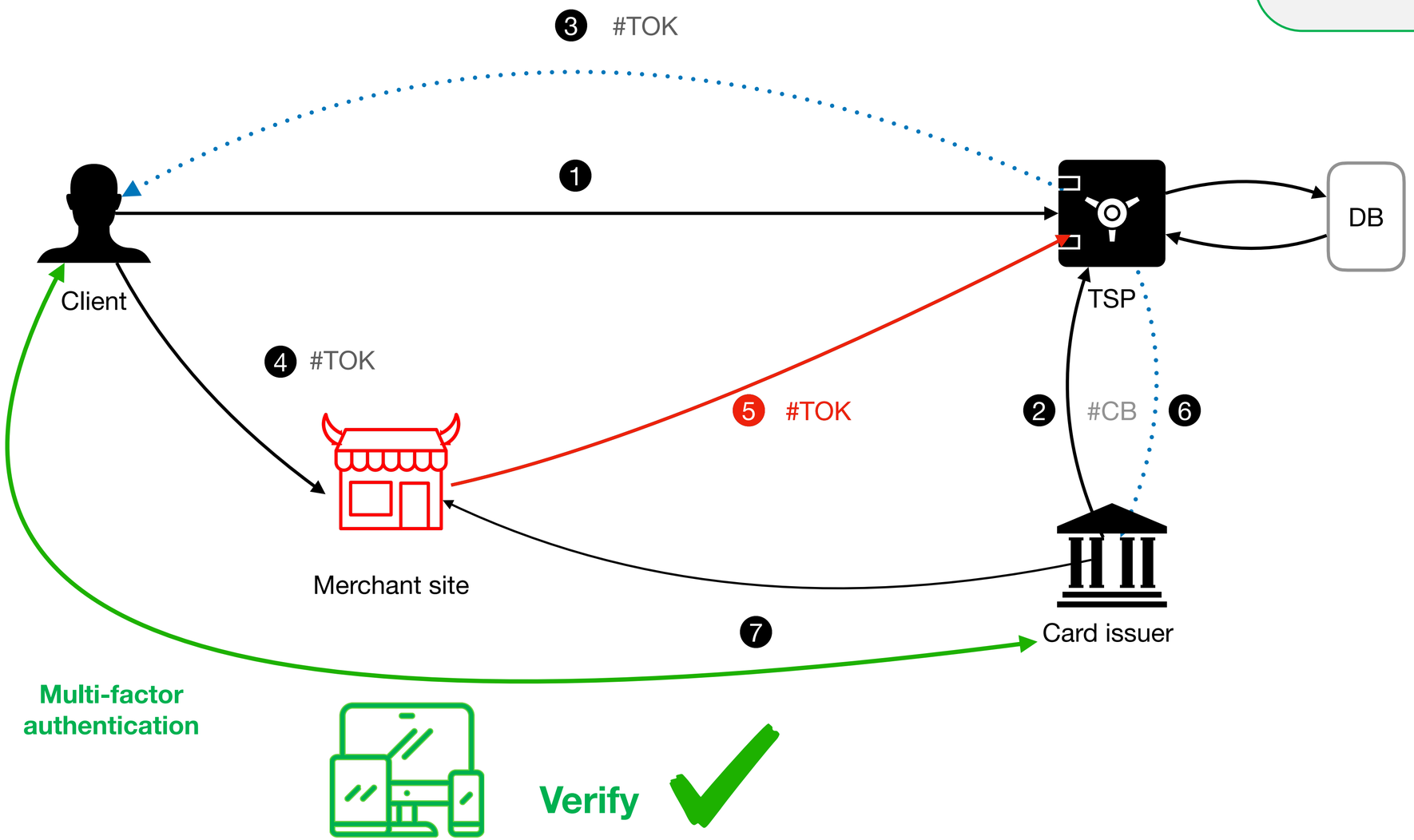
1. Assessing the customer's identity
2. Customer's anonymity guaranteed



Unforgeability

Multi-factor authentication

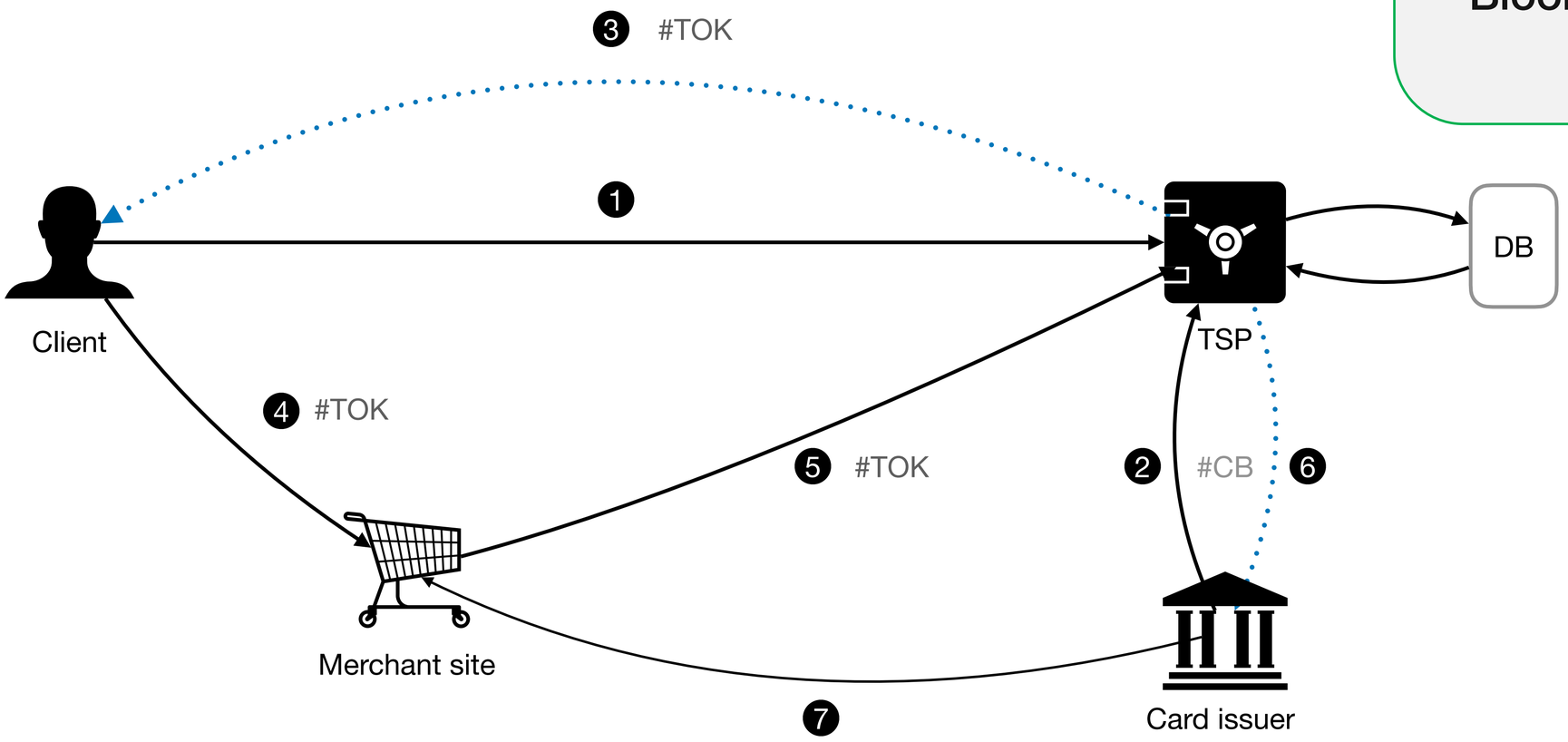
Client claim their identity from their device.



Hash tables

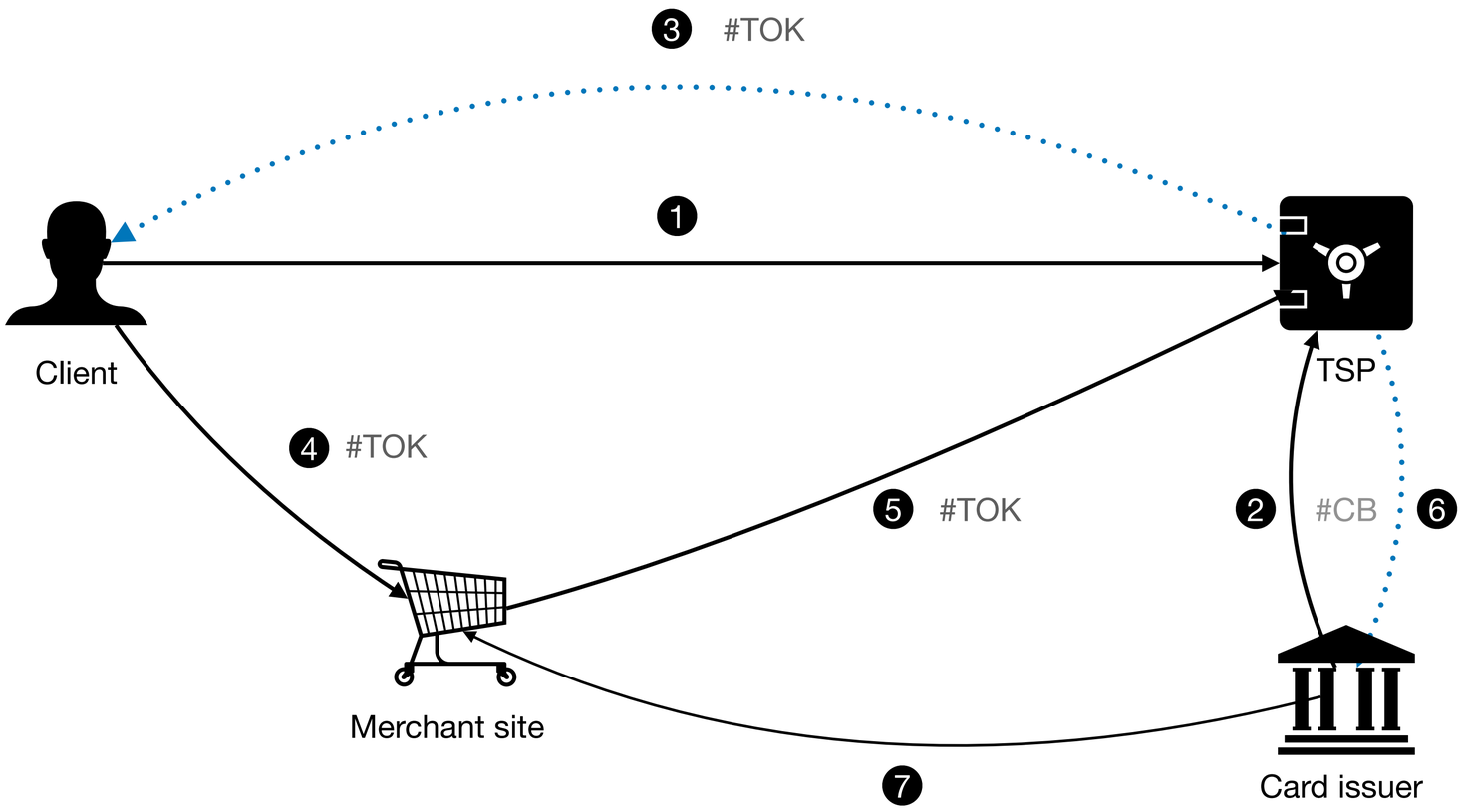
And space efficient data-structures

- Hash tables instead of basic tables to handle duplicate values
- Space-efficient data structures: Bloom and Cuckoo filters



Hash tables

And space efficient data-structures



- Hash tables instead of basic tables to handle duplicate values
- Space-efficient data structures: Bloom and Cuckoo filters

Scalable Bloom Filters ☆

Paulo Sérgio Almeida [a,*], Carlos Baquero [a,1], Nuno Preguiça [b], David Hutchison [c]

[a] *CCTC/DI, Universidade do Minho, Portugal*
[b] *CITI/DI, FCT, Universidade Nova de Lisboa, Portugal*
[c] *Computing Department, Lancaster University, UK*

Accepted 20 October 2006

Available online 22 November 2006

Communicated by F. Meyer auf der Heide

Demo

Takeaway slide

- **Collisions ?**
  7000 tries per seconds to be optimized
- **Uniformity ?**
  Proven
- **Untraceability ?**
  No link by design
- Extra possibilities: auditable DB, unforgeability, scalable Bloom filters

Thank you for your attention.

Questions?

Shoutout to Pascal and Olivier

paul.huynh@loria.fr
simon.masson@loria.fr
cyrius.nugier@laas.fr
diane.leblanc-albarel@irisa.fr
agathe.blaise@lip6.fr
brice.wandji@fime.fr