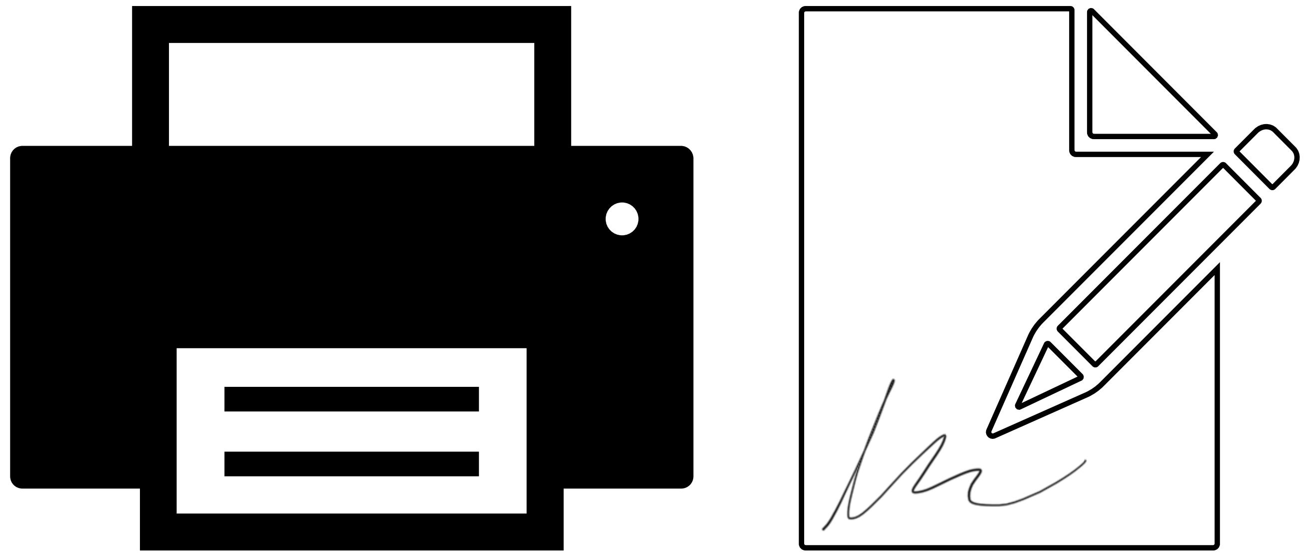
Modification-Tolerant Signature Schemes

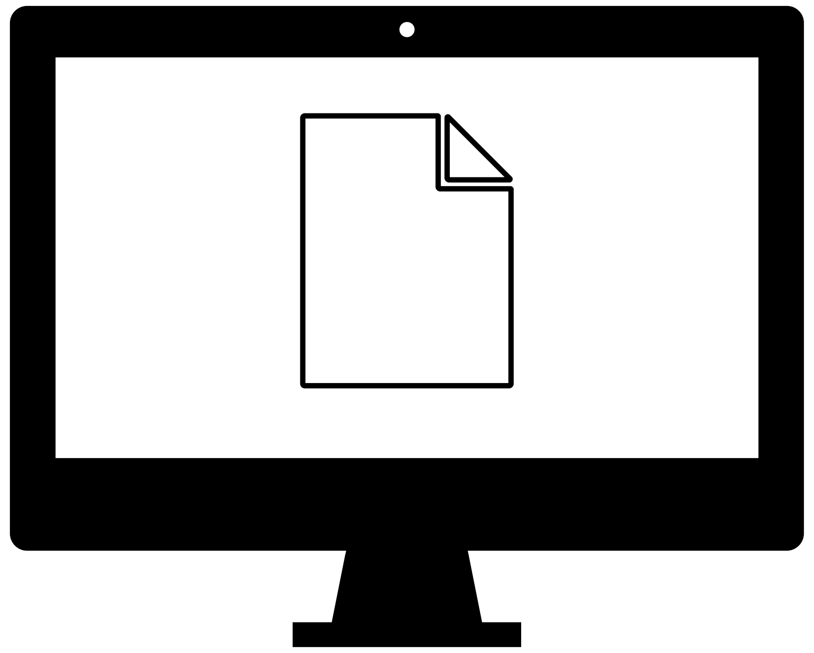
Thaís Bardini Idalino

Universidade Federal de Santa Catarina

Brazil

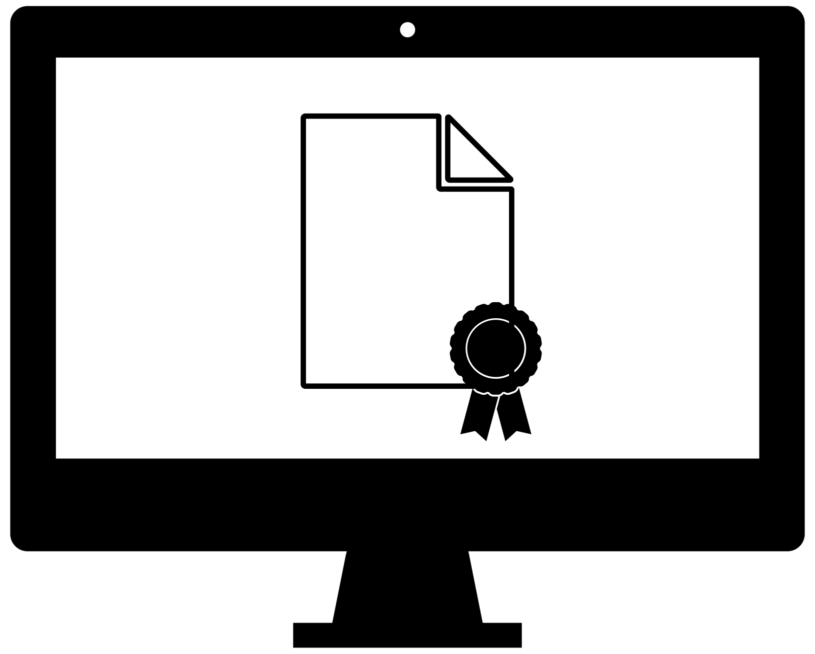
Joint work with Lucia Moura and Carlisle Adams



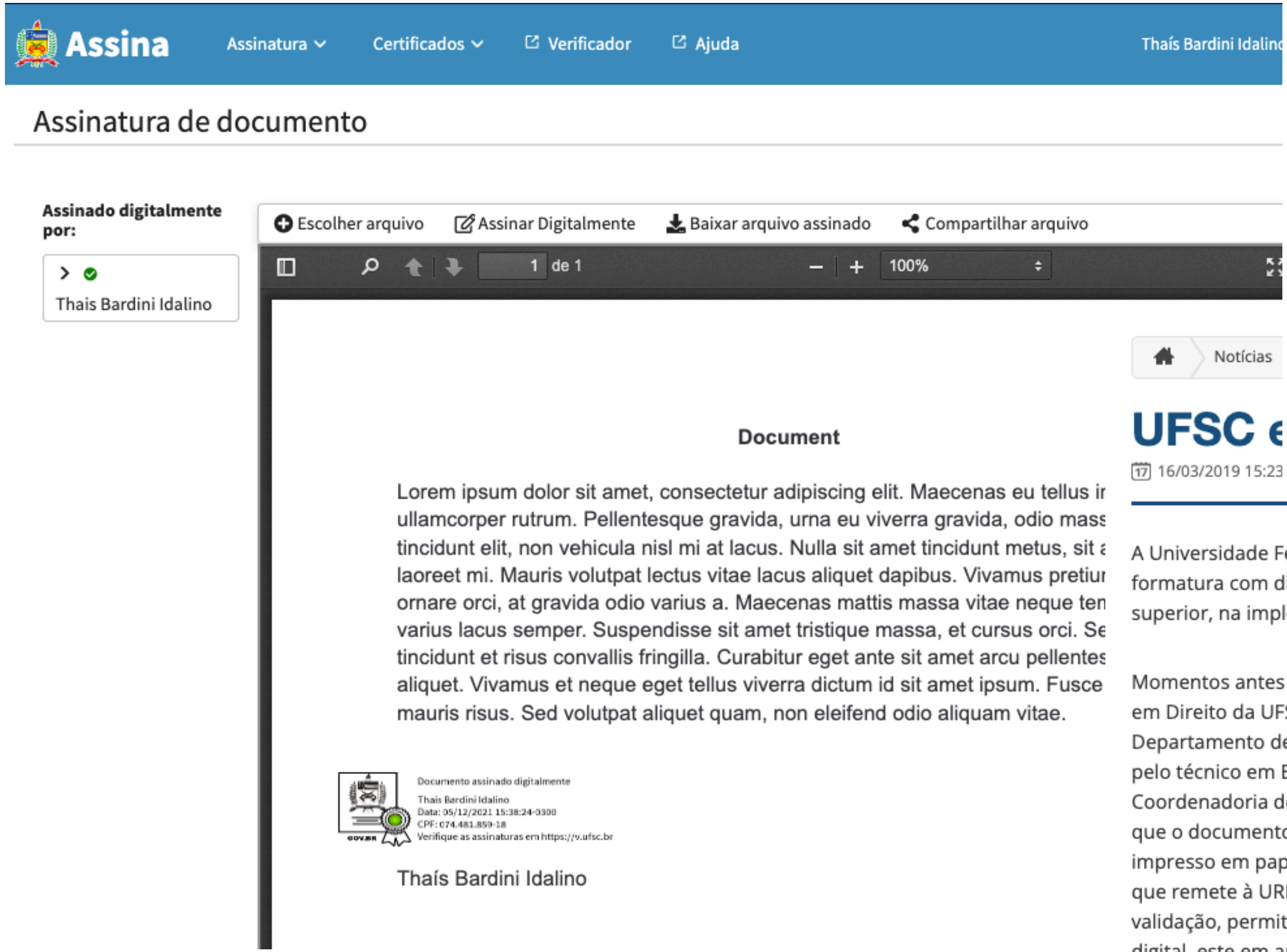


Digital Signatures

Authenticity
Integrity
Non-Repudiation



Digital Signatures in Brazil

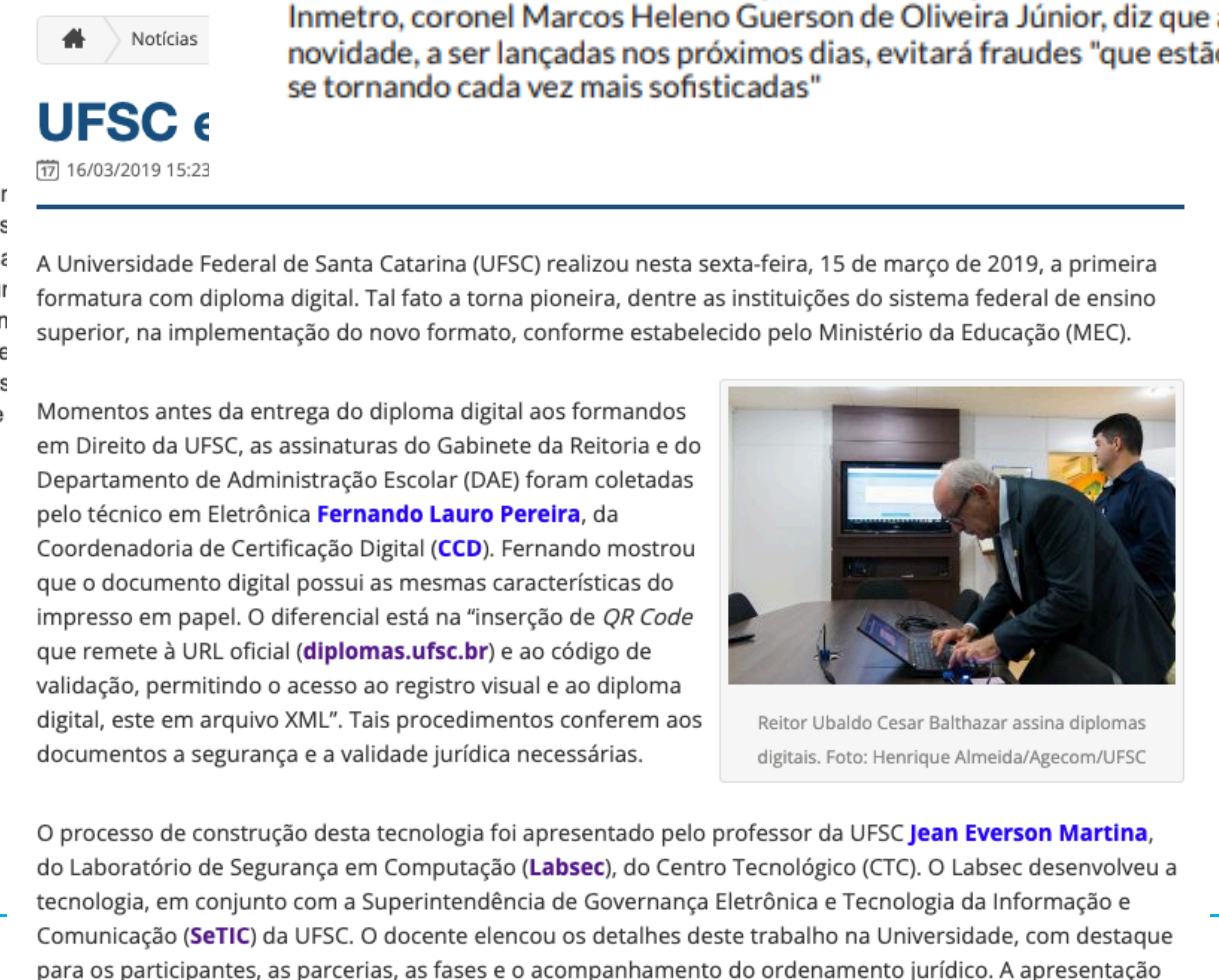


Correio Braziliense

CB.PODER

Inmetro lançará certificação digital para evitar fraude em bombas de combustíveis

Em entrevista ao Correio nesta terça-feira (8/6), o presidente do Inmetro, coronel Marcos Heleno Guerson de Oliveira Júnior, diz que a novidade, a ser lançadas nos próximos dias, evitará fraudes "que estão se tornando cada vez mais sofisticadas"



Notícias

UFSC e

16/03/2019 15:23

A Universidade Federal de Santa Catarina (UFSC) realizou nesta sexta-feira, 15 de março de 2019, a primeira formatura com diploma digital. Tal fato a torna pioneira, dentre as instituições do sistema federal de ensino superior, na implementação do novo formato, conforme estabelecido pelo Ministério da Educação (MEC).

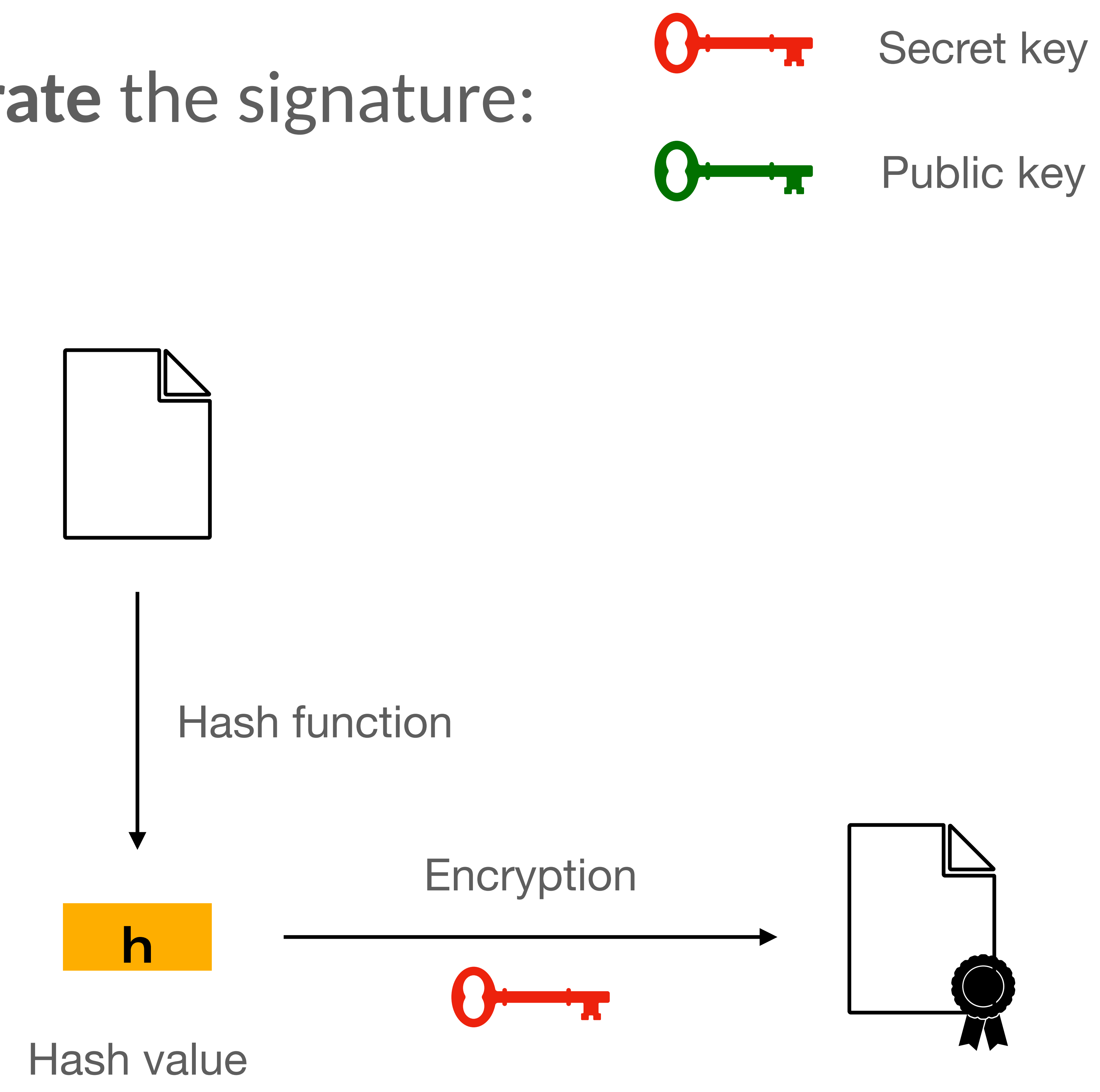
Momentos antes da entrega do diploma digital aos formandos em Direito da UFSC, as assinaturas do Gabinete da Reitoria e do Departamento de Administração Escolar (DAE) foram coletadas pelo técnico em Eletrônica **Fernando Lauro Pereira**, da Coordenadoria de Certificação Digital (**CCD**). Fernando mostrou que o documento digital possui as mesmas características do impresso em papel. O diferencial está na "inserção de *QR Code* que remete à URL oficial (**diplomas.ufsc.br**) e ao código de validação, permitindo o acesso ao registro visual e ao diploma digital, este em arquivo XML". Tais procedimentos conferem aos documentos a segurança e a validade jurídica necessárias.

Reitor Ubaldo Cesar Balthazar assina diplomas digitais. Foto: Henrique Almeida/Agecom/UFSC

O processo de construção desta tecnologia foi apresentado pelo professor da UFSC **Jean Everson Martina**, do Laboratório de Segurança em Computação (**Labsec**), do Centro Tecnológico (CTC). O Labsec desenvolveu a tecnologia, em conjunto com a Superintendência de Governança Eletrônica e Tecnologia da Informação e Comunicação (**SeTIC**) da UFSC. O docente elencou os detalhes deste trabalho na Universidade, com destaque para os participantes, as parcerias, as fases e o acompanhamento do ordenamento jurídico. A apresentação

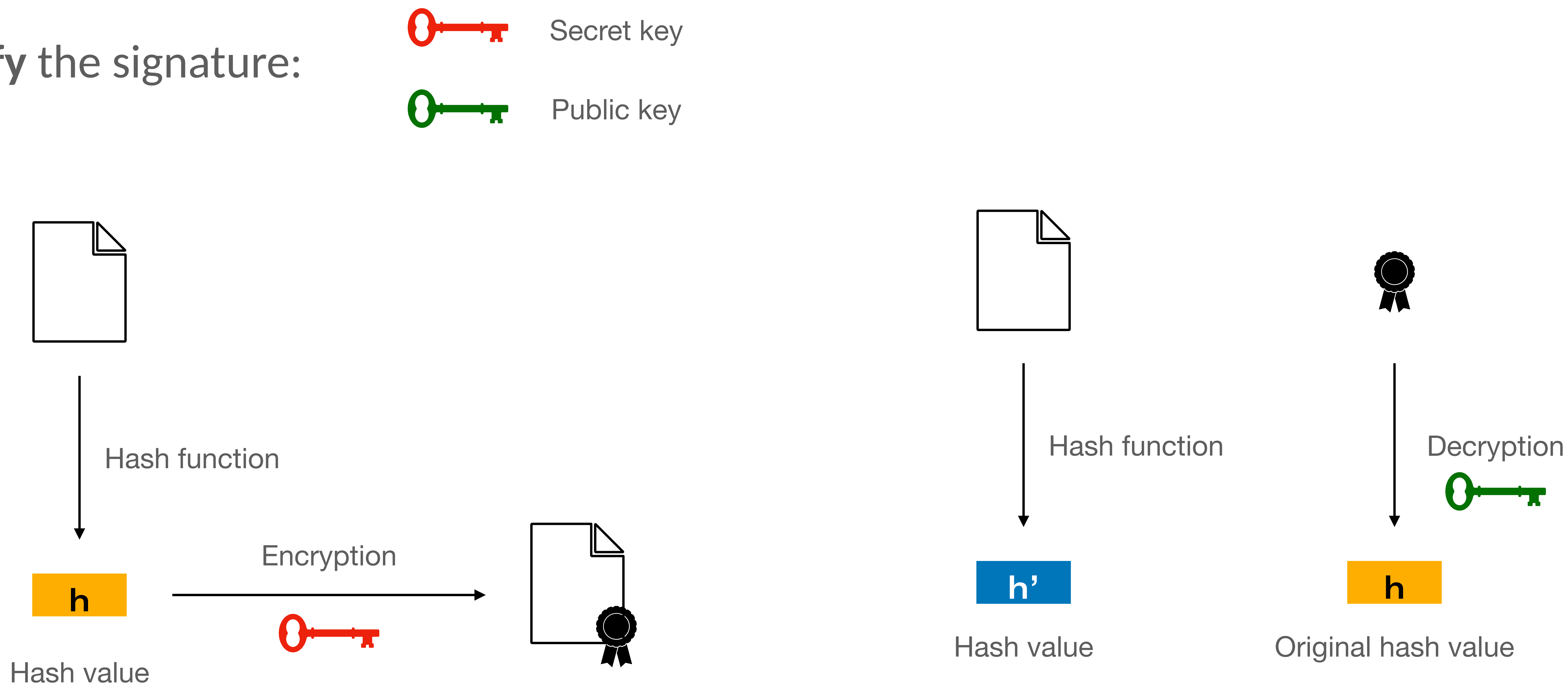
Digital Signatures

How to **generate** the signature:



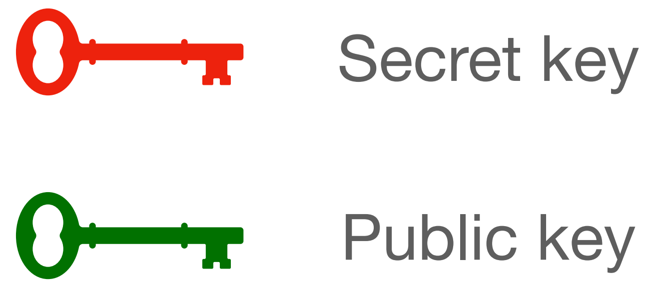
Digital Signatures

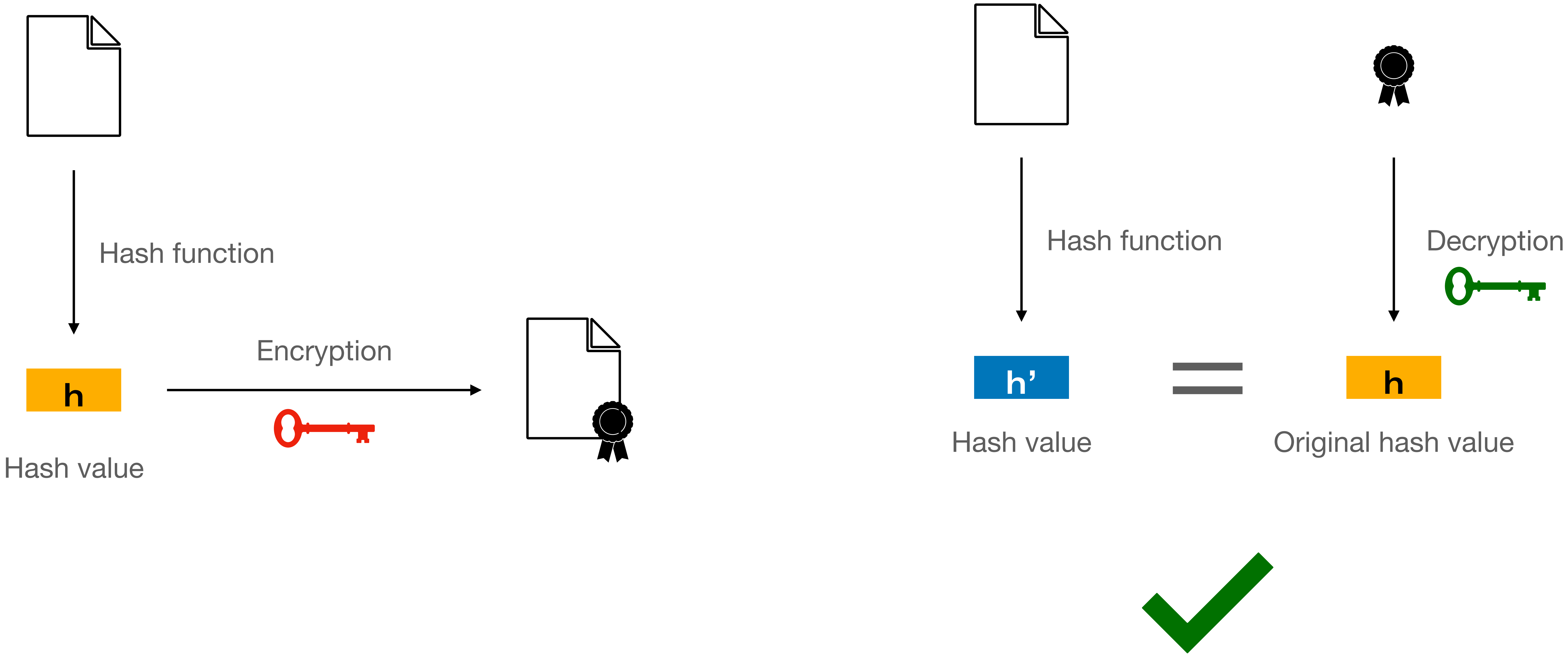
How to **verify** the signature:



Digital Signatures

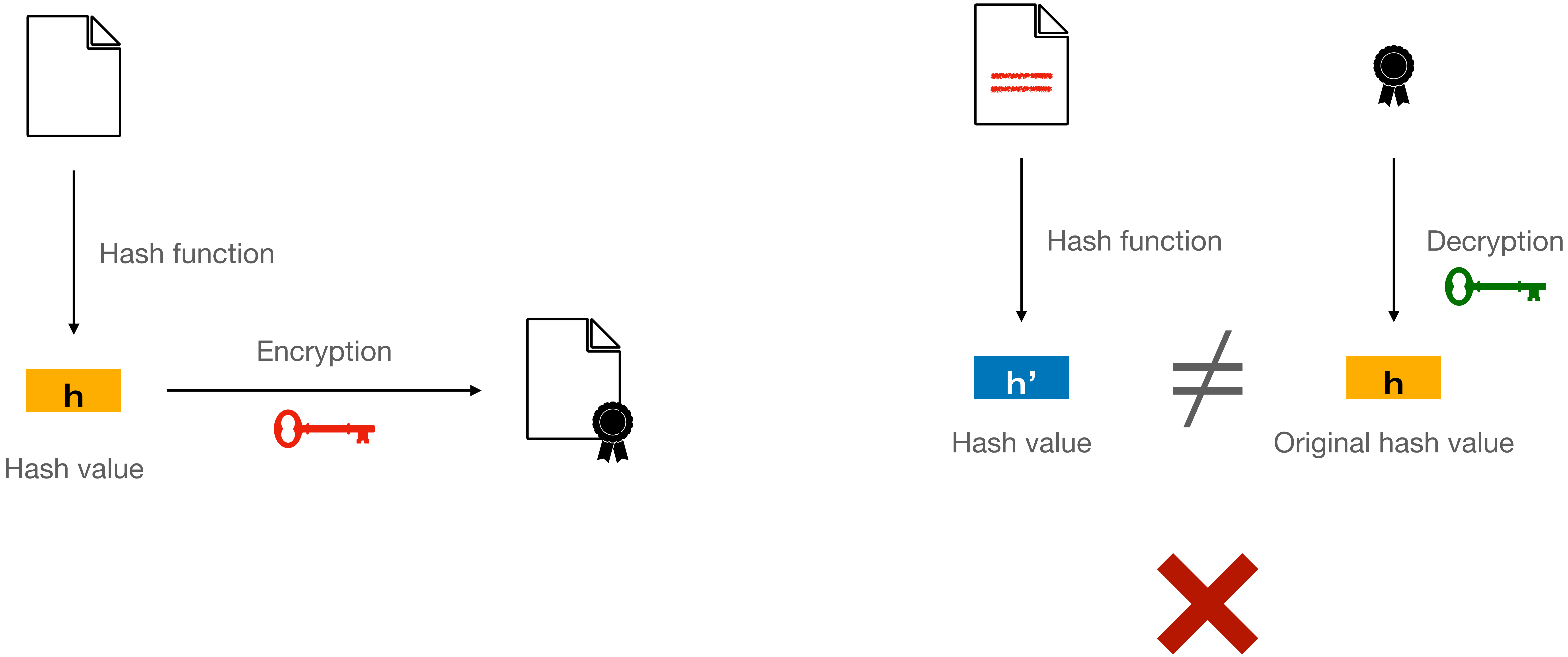
How to **verify** the signature:





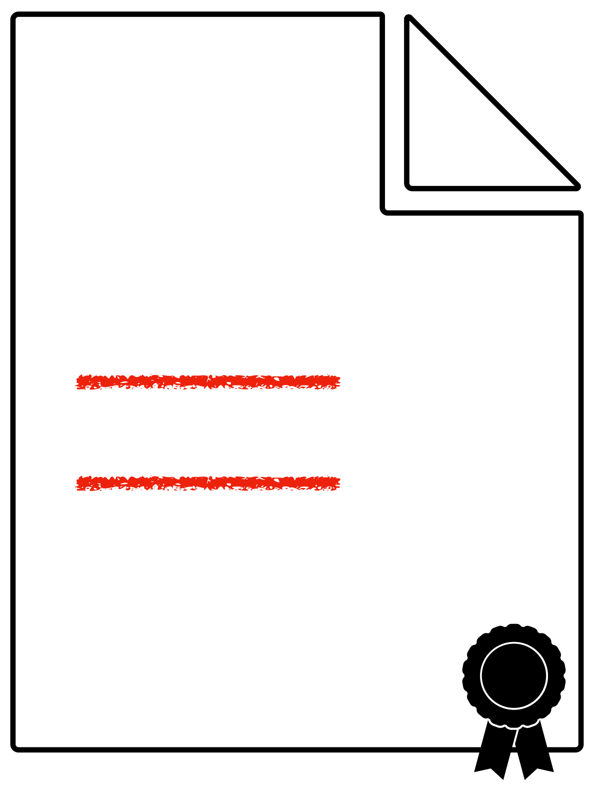
Digital Signatures

How to **verify** the signature:



Partial integrity

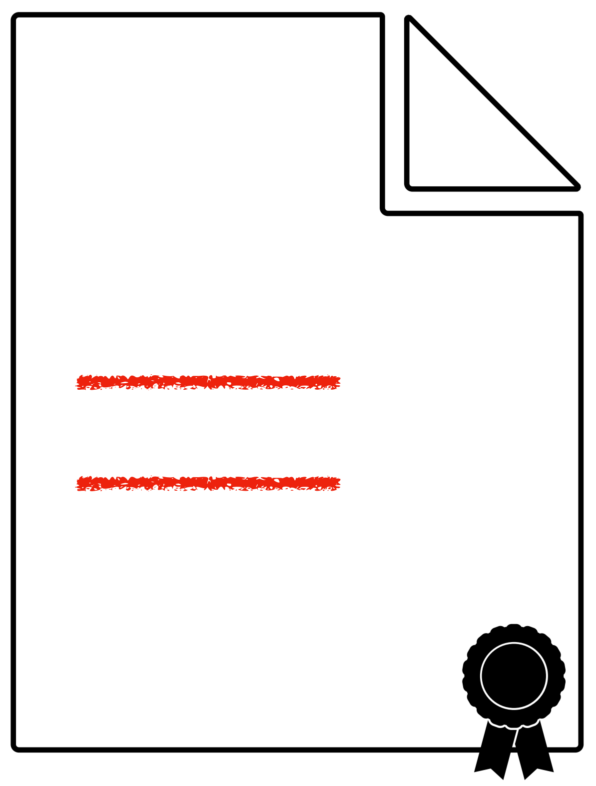
What if the modifications matter?



Partial integrity

What if the modifications matter?

- A fillable form signed by the owner but filled by another person
- A document with private sections that need to be redacted
- Errors during transmission or storage of signed data
- Malicious modifications

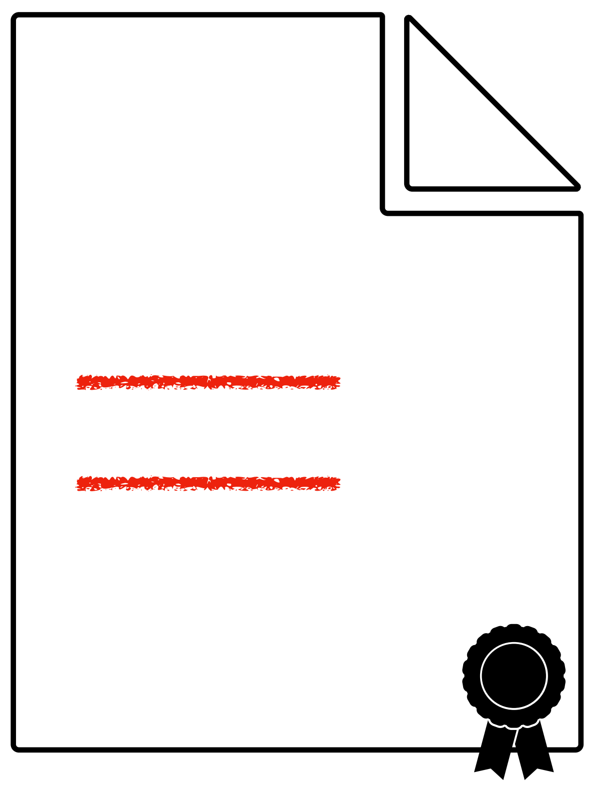


Partial integrity

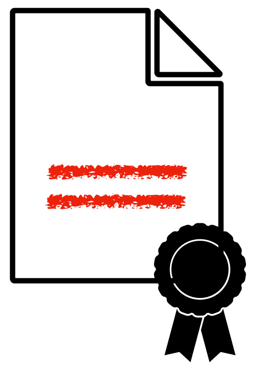
What if the modifications matter?

- A fillable form signed by the owner but filled by another person
- A document with private sections that need to be redacted
- Errors during transmission or storage of signed data
- Malicious modifications

How can we provide partial integrity of data?



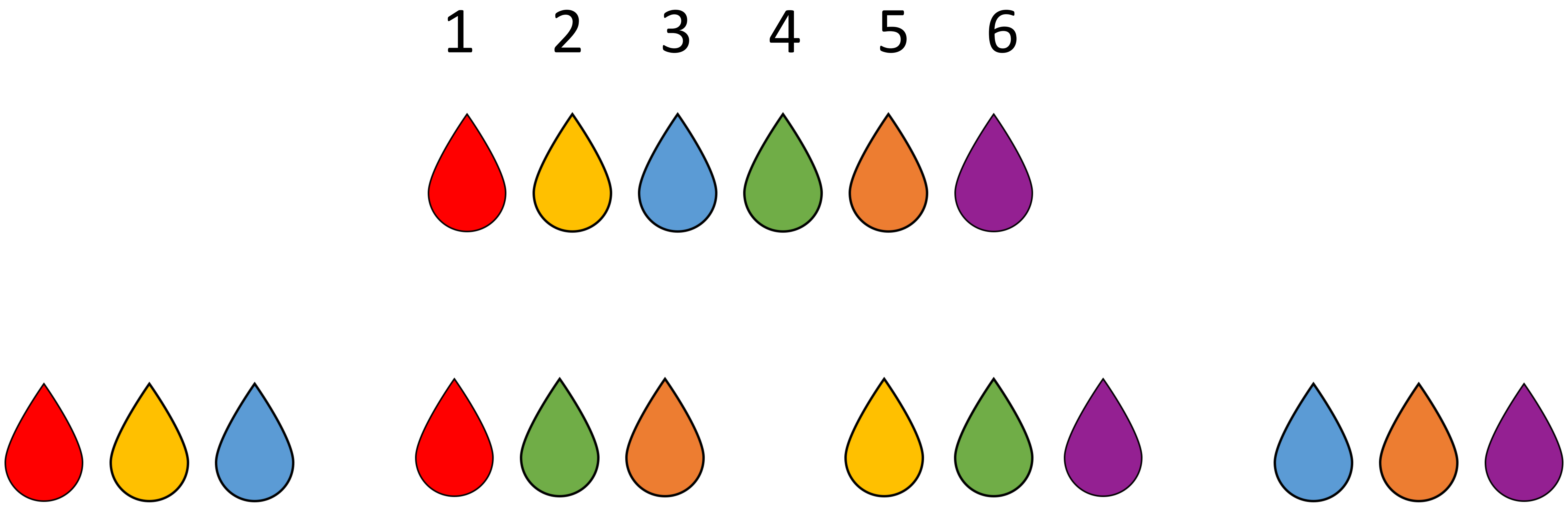
In this talk



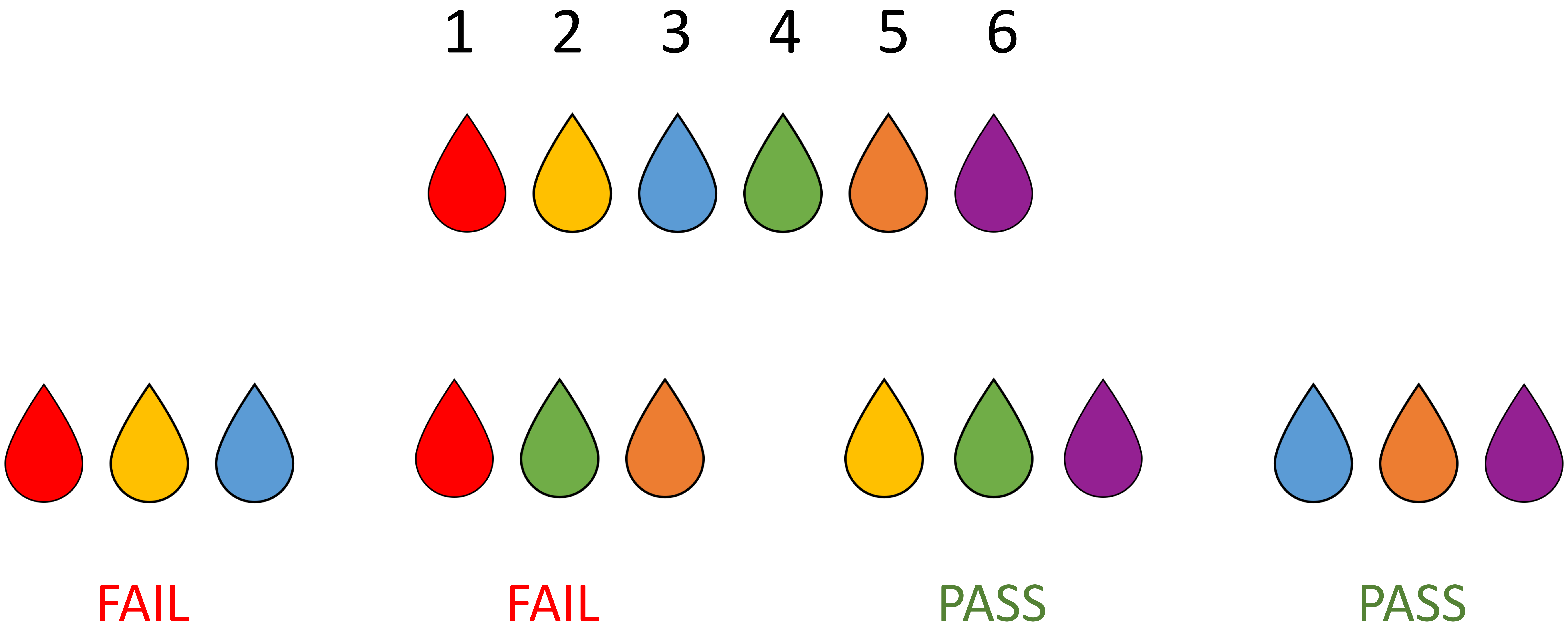
1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

- A **modification-tolerant signature scheme** using **cover-free families**
- How to locate modifications.
- How to correct modifications.
- How to guarantee privacy of redacted data.

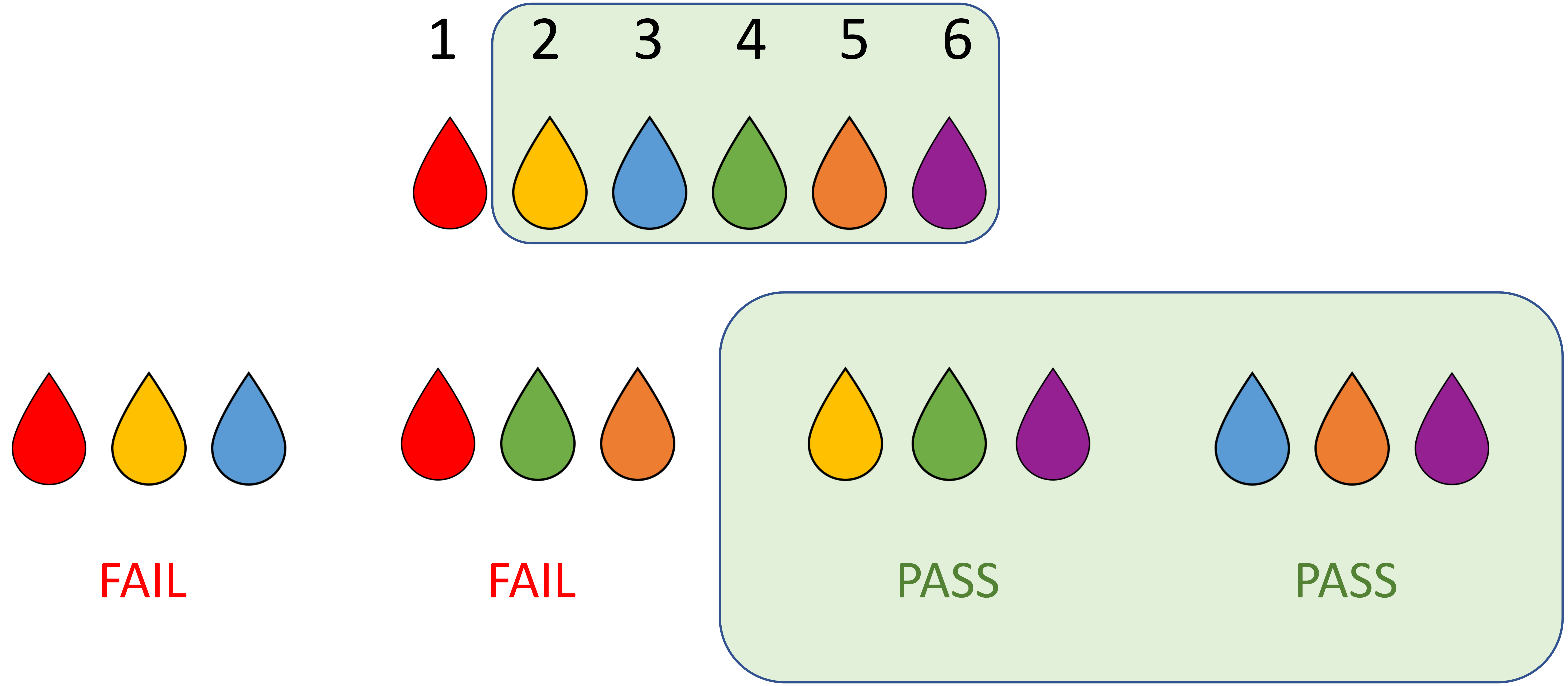
Group testing



Group testing

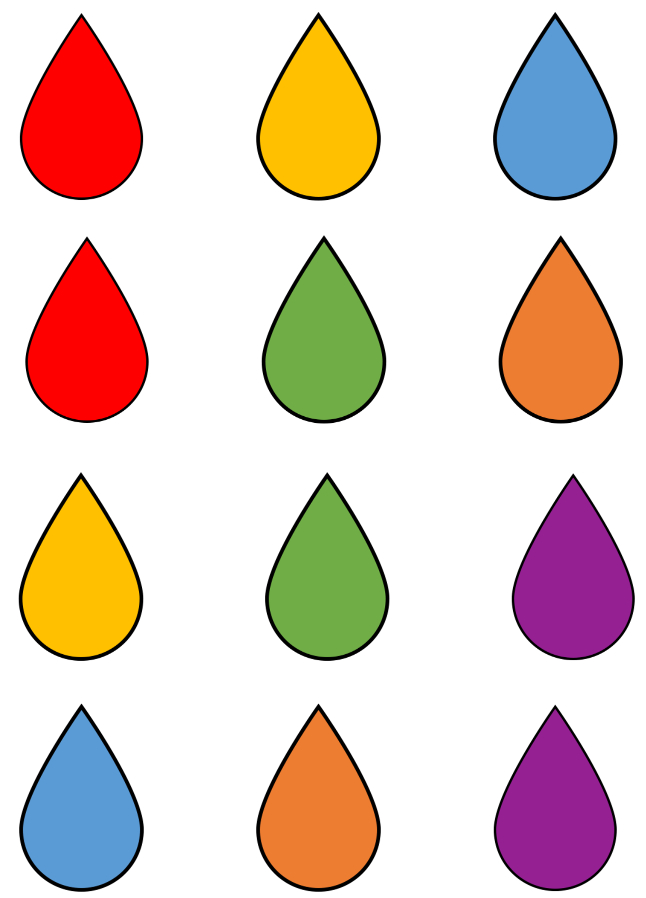


Group testing



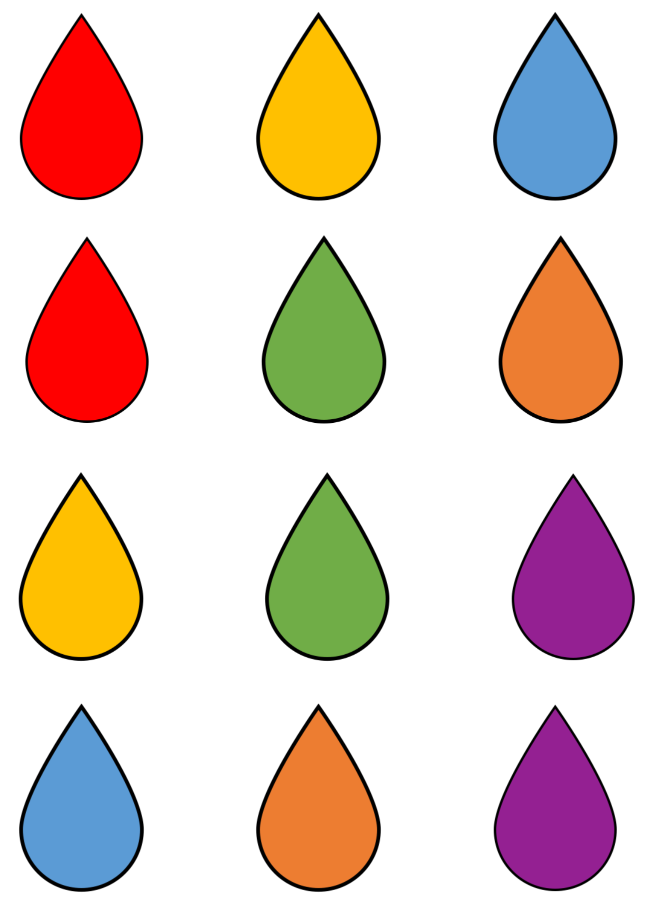
Cover-Free Families

	1	2	3	4	5	6	
Test 1	1	1	1	0	0	0	FAIL
Test 2	1	0	0	1	1	0	FAIL
Test 3	0	1	0	1	0	1	PASS
Test 4	0	0	1	0	1	1	PASS



Cover-Free Families

	1	2	3	4	5	6
Test 1	1	1	1	0	0	0
Test 2	1	0	0	1	1	0
Test 3	0	1	0	1	0	1
Test 4	0	0	1	0	1	1



***d* – cover-free family**

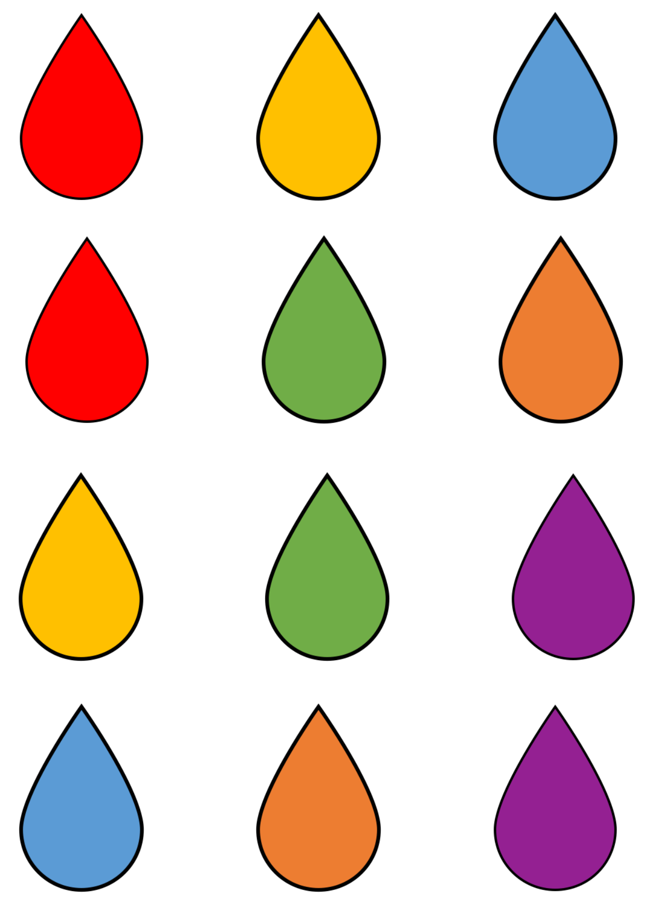
Cover-Free Families

	1	2	3	4	5	6	
Test 1	1	1	1	0	0	0	FAIL
Test 2	1	0	0	1	1	0	FAIL
Test 3	0	1	0	1	0	1	PASS
Test 4	0	0	1	0	1	1	PASS

***1* – CFF(4, 6)**

Cover-Free Families

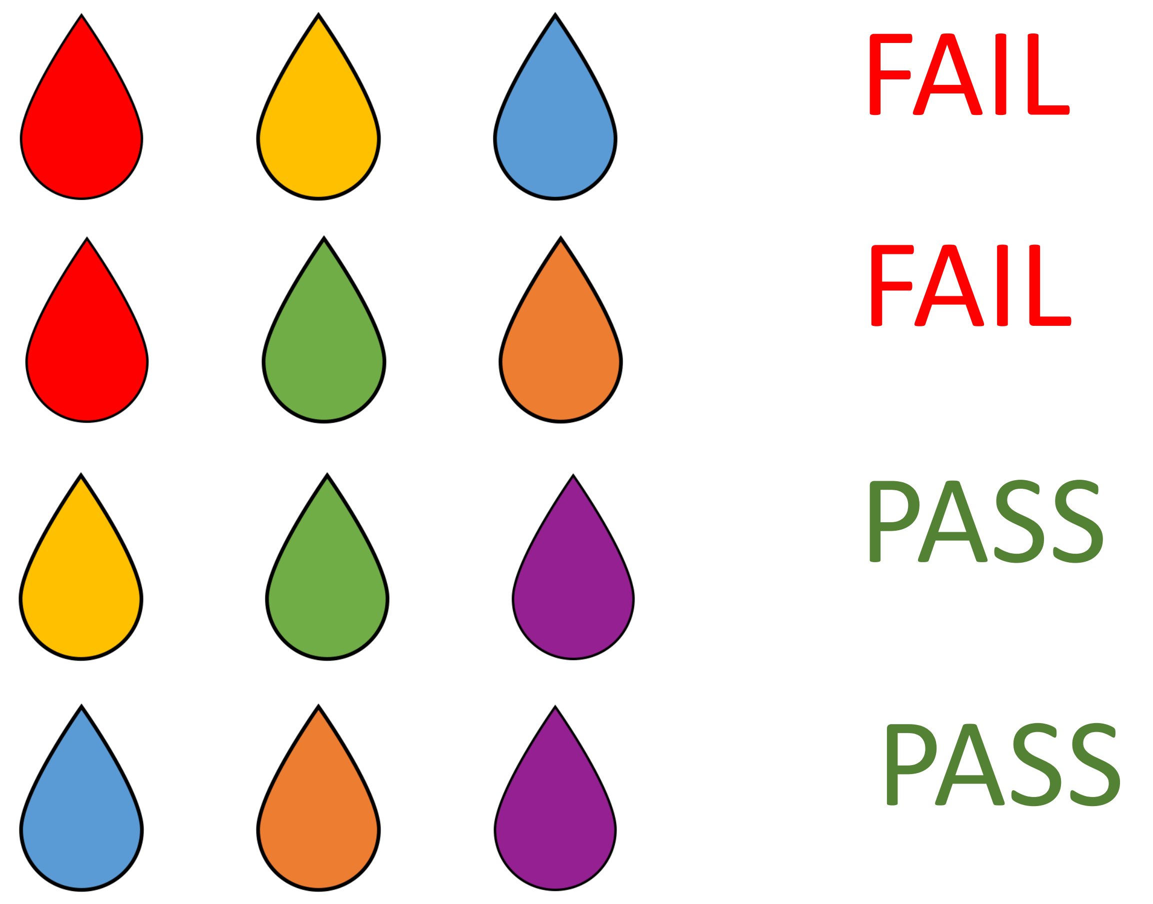
	1	2	3	4	5	6	
Test 1	1	1	1	0	0	0	FAIL
Test 2	1	0	0	1	1	0	FAIL
Test 3	0	1	0	1	0	1	PASS
Test 4	0	0	1	0	1	1	PASS



1 – CFF(4, 6)

Cover-Free Families

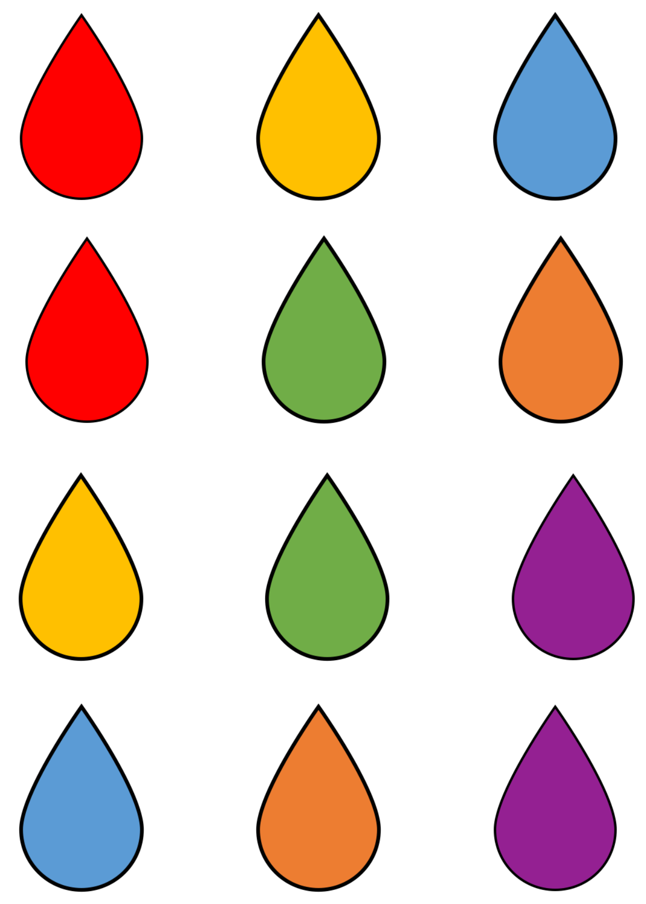
	1	2	3	4	5	6
Test 1	1	1	1	0	0	0
Test 2	1	0	0	1	1	0
Test 3	0	1	0	1	0	1
Test 4	0	0	1	0	1	1



***1* – CFF(4, 6)**

Cover-Free Families

	1	2	3	4	5	6
Test 1	1	1	1	0	0	0
Test 2	1	0	0	1	1	0
Test 3	0	1	0	1	0	1
Test 4	0	0	1	0	1	1



FAIL

FAIL

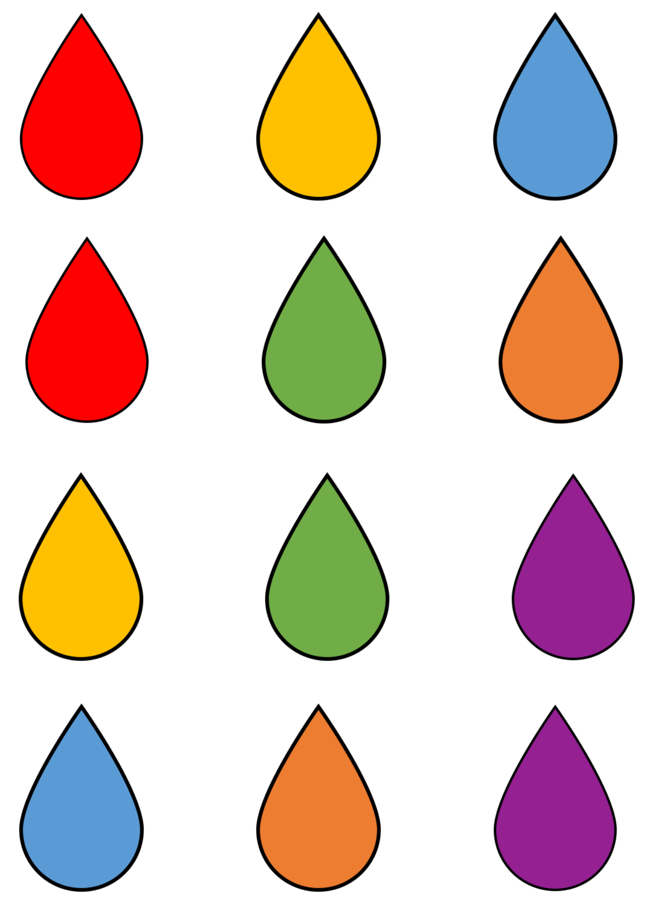
PASS

PASS

***1* – CFF(4, 6)**

Cover-Free Families

	1	2	3	4	5	6
Test 1	1	1	1	0	0	0
Test 2	1	0	0	1	1	0
Test 3	0	1	0	1	0	1
Test 4	0	0	1	0	1	1

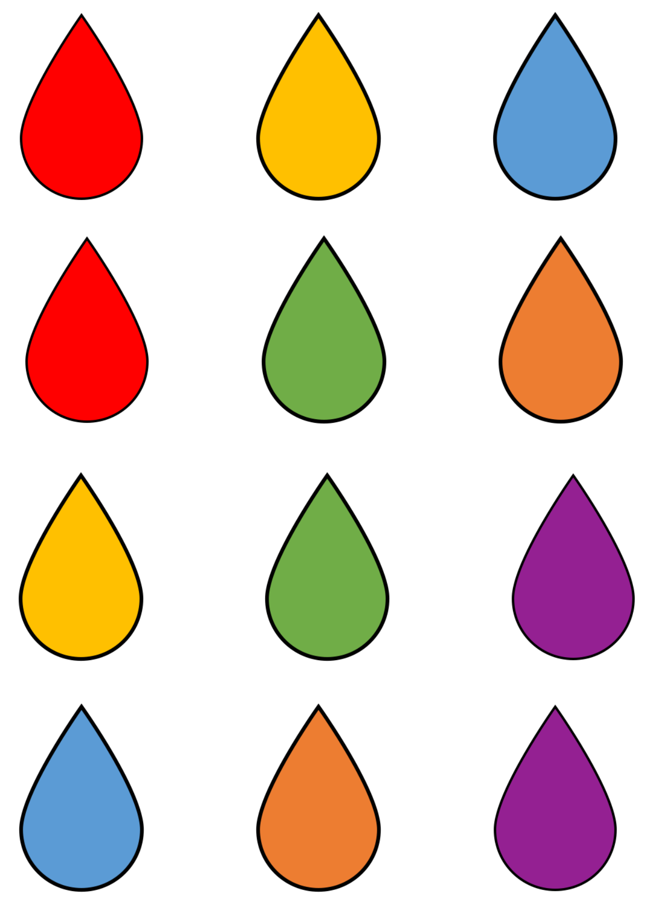


FAIL
FAIL
PASS
PASS

***1* – CFF(4, 6)**

Cover-Free Families

	1	2	3	4	5	6	
Test 1	1	1	1	0	0	0	FAIL
Test 2	1	0	0	1	1	0	FAIL
Test 3	0	1	0	1	0	1	PASS
Test 4	0	0	1	0	1	1	PASS



***1* – CFF(4, 6)**

Constructions

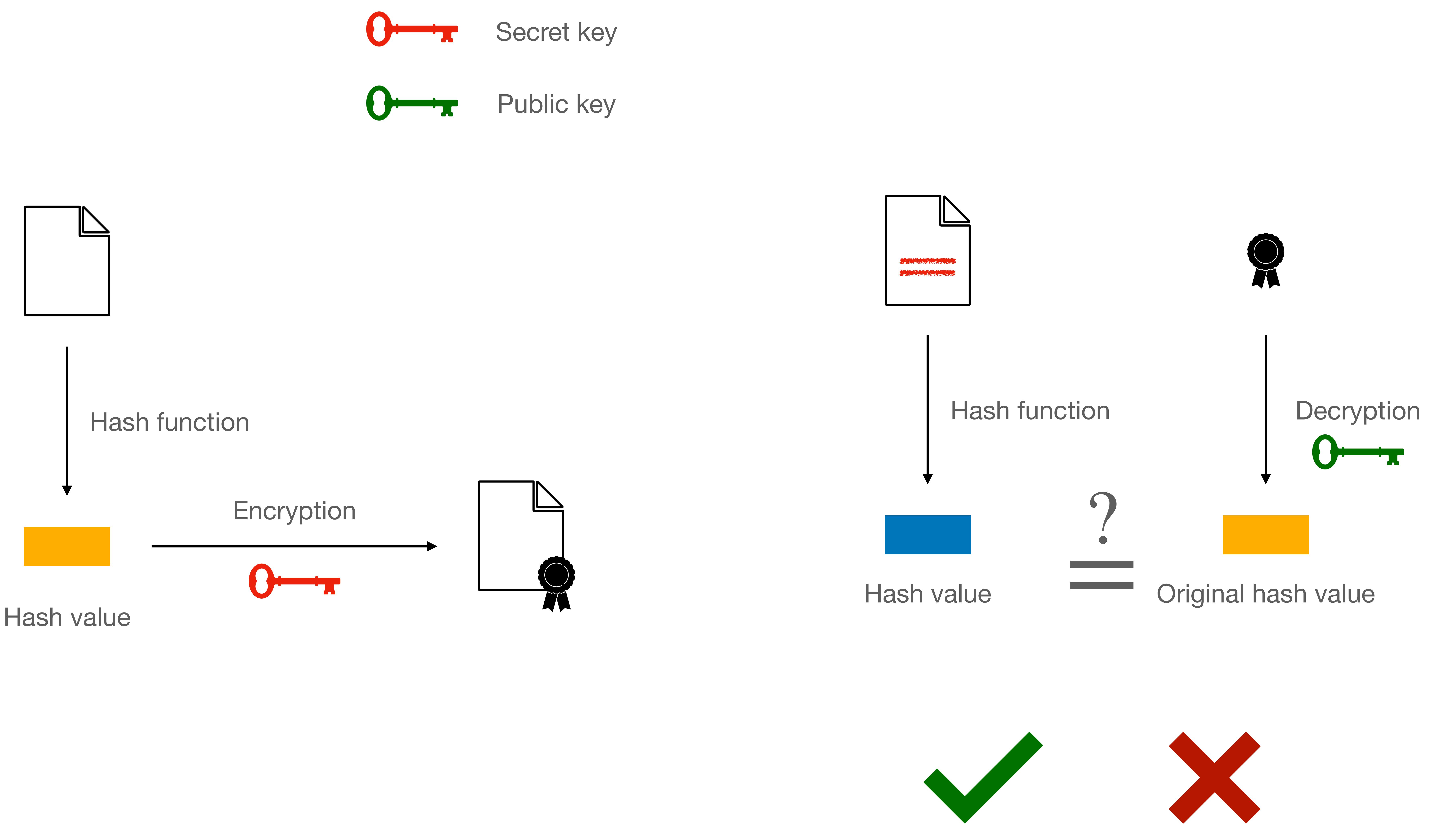
- When ***d* = 1** we can use Sperner sets, where *t* grows as $\log_2 n$ as $n \to \infty$;
- For $d \geq 2$, the best known lower bound on t for *d*-CFF(*t*, *n*) is given by

$$t \geq c \frac{d^2}{\log d} \log n$$

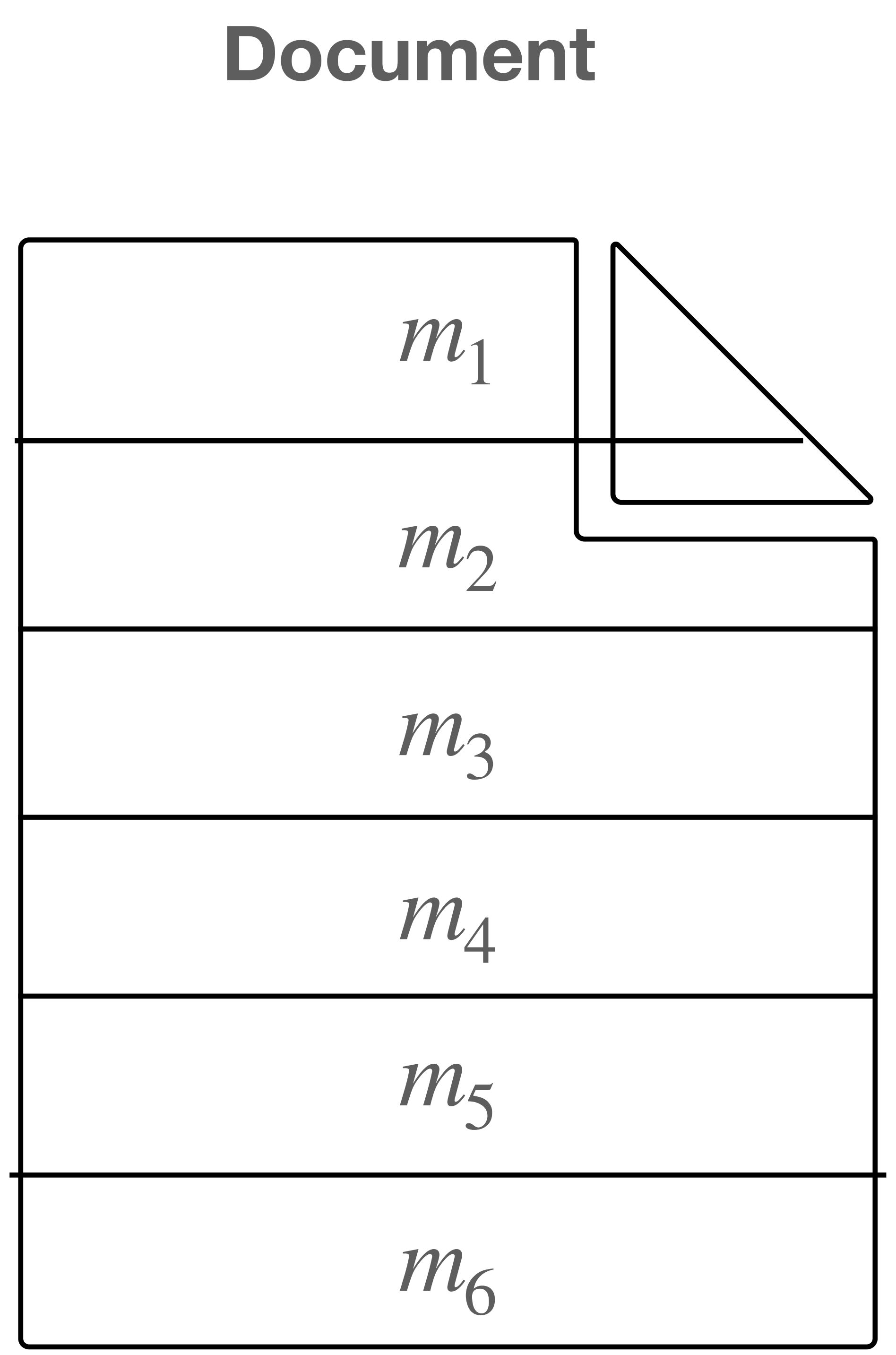
for some constant *c*;

- Constructions based on Latin squares, OAs, PHFs, CAs, Codes, and many others;
- Constructions based on polynomials over finite fields;
- Constructions based on probabilistic algorithms.

How to locate modifications



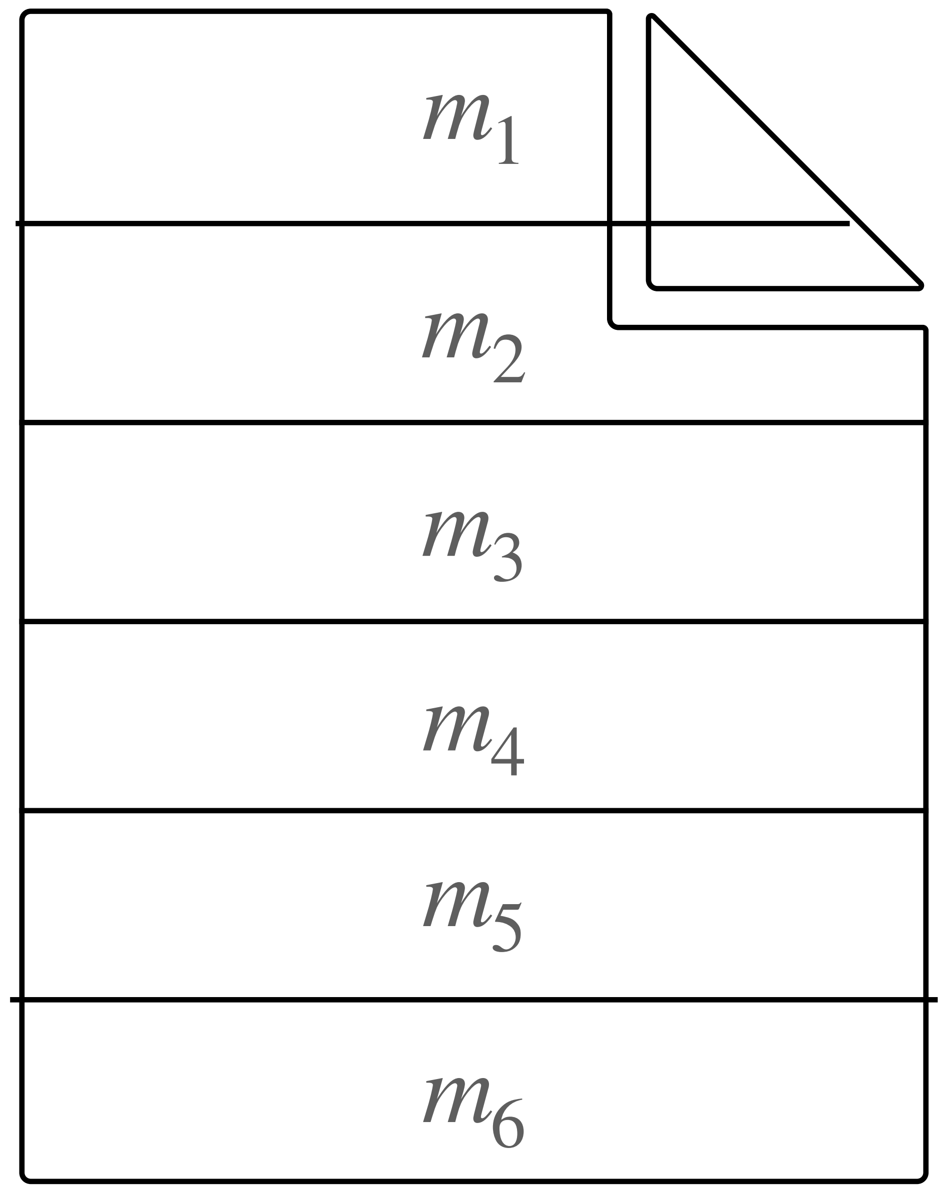
How to locate modifications



T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to locate modifications

Document



h_1 = Hash(m_1)

h_2 = Hash(m_2)

h_3 = Hash(m_3)

h_4 = Hash(m_4)

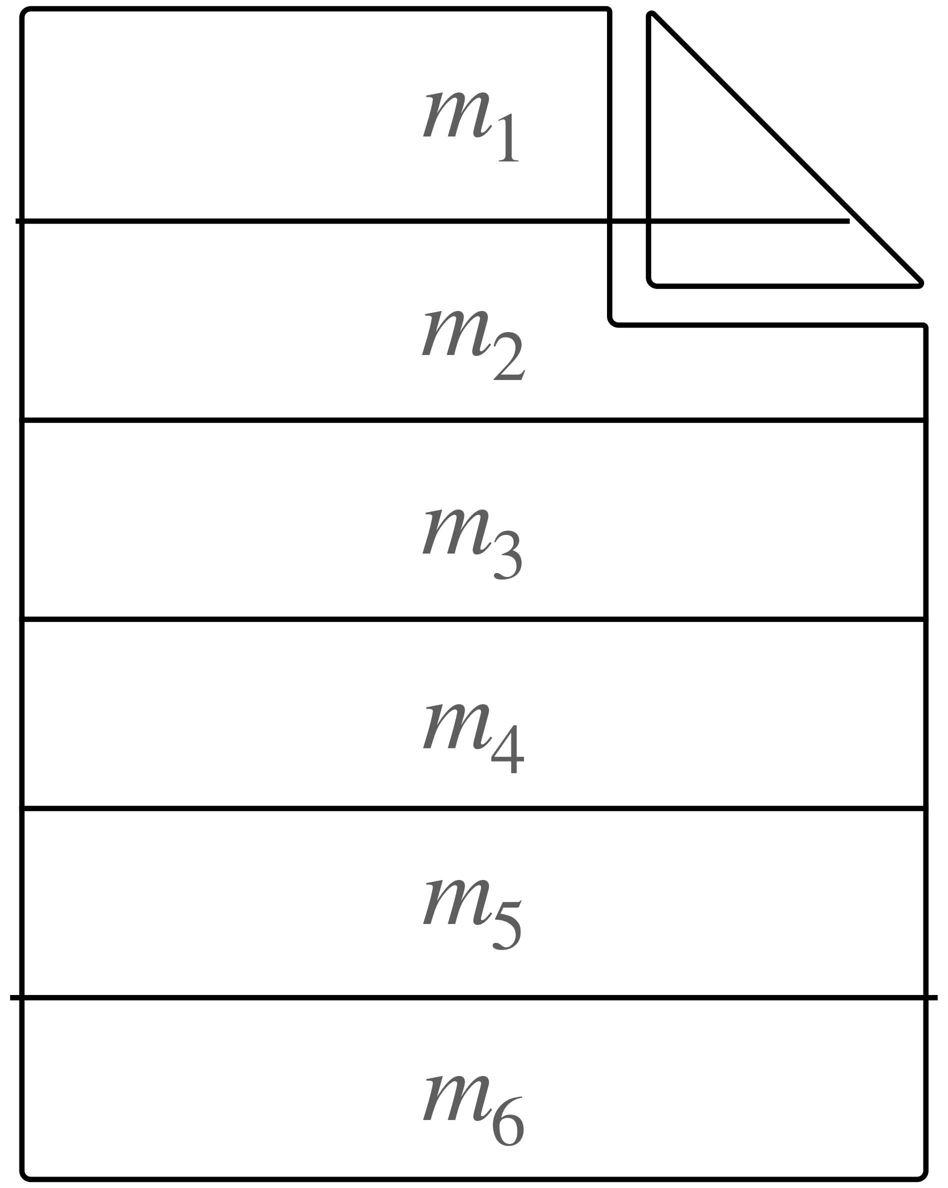
h_5 = Hash(m_5)

h_6 = Hash(m_6)

T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to locate modifications

Document



Signature

$Hash(h_1 | h_2 | h_3)$

$Hash(h_1 | h_4 | h_5)$

$Hash(h_2 | h_4 | h_6)$

$Hash(h_3 | h_5 | h_6)$

***d*-CFF**

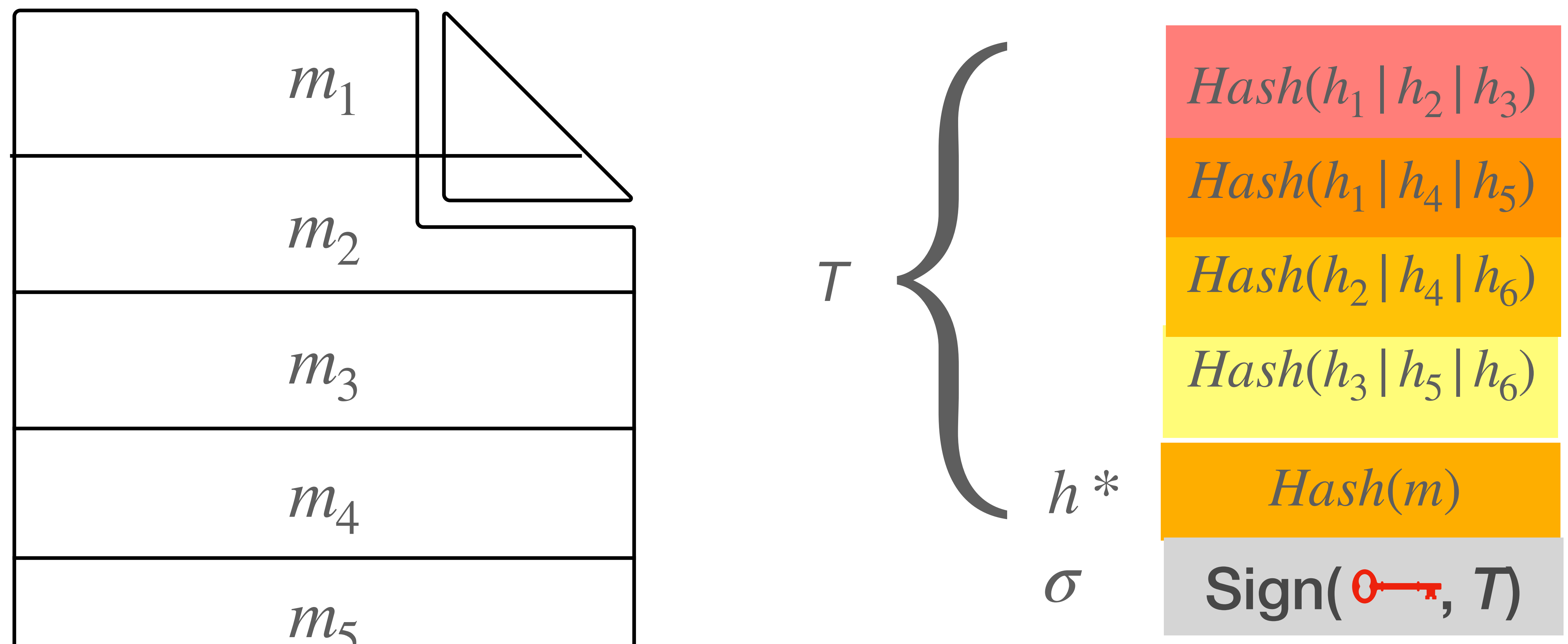
1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to locate modifications

Document

Signature



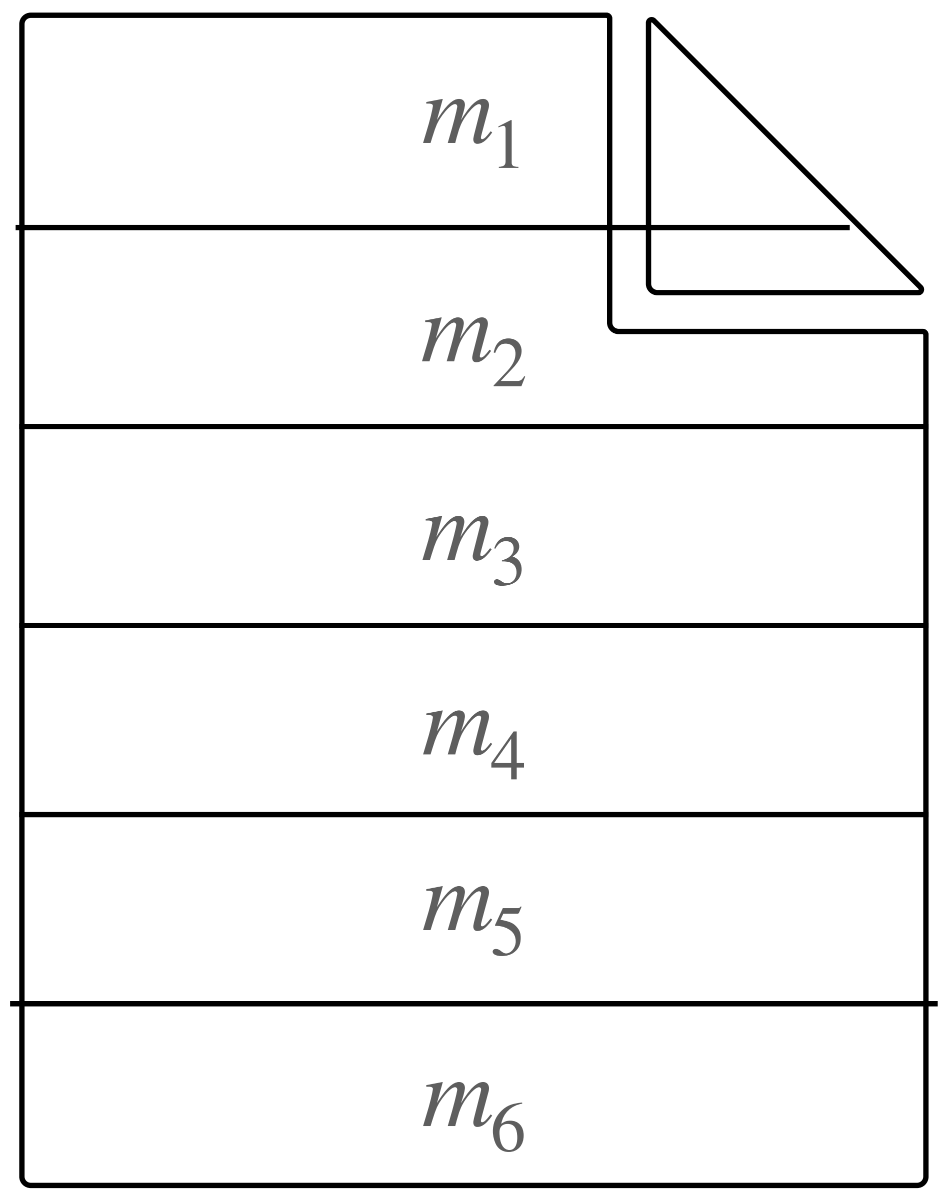
***d*-CFF**

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to **locate** modifications

Document



Signature

$$T \left\{ \begin{array}{l} Hash(h_1|h_2|h_3) \\ Hash(h_1|h_4|h_5) \\ Hash(h_2|h_4|h_6) \\ Hash(h_3|h_5|h_6) \end{array} \right.$$

$h*$ $Hash(m)$

σ Sign(, T)

d-CFF

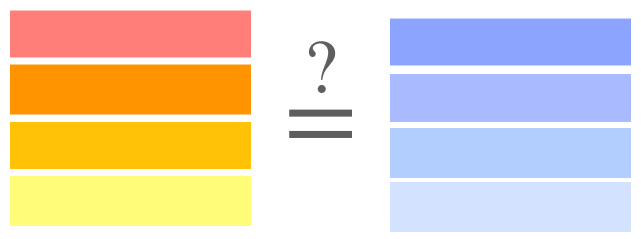
1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

Verification

1) $Verify(T, \sigma,)$ OK?

2) $h* \stackrel{?}{=} Hash(m')$

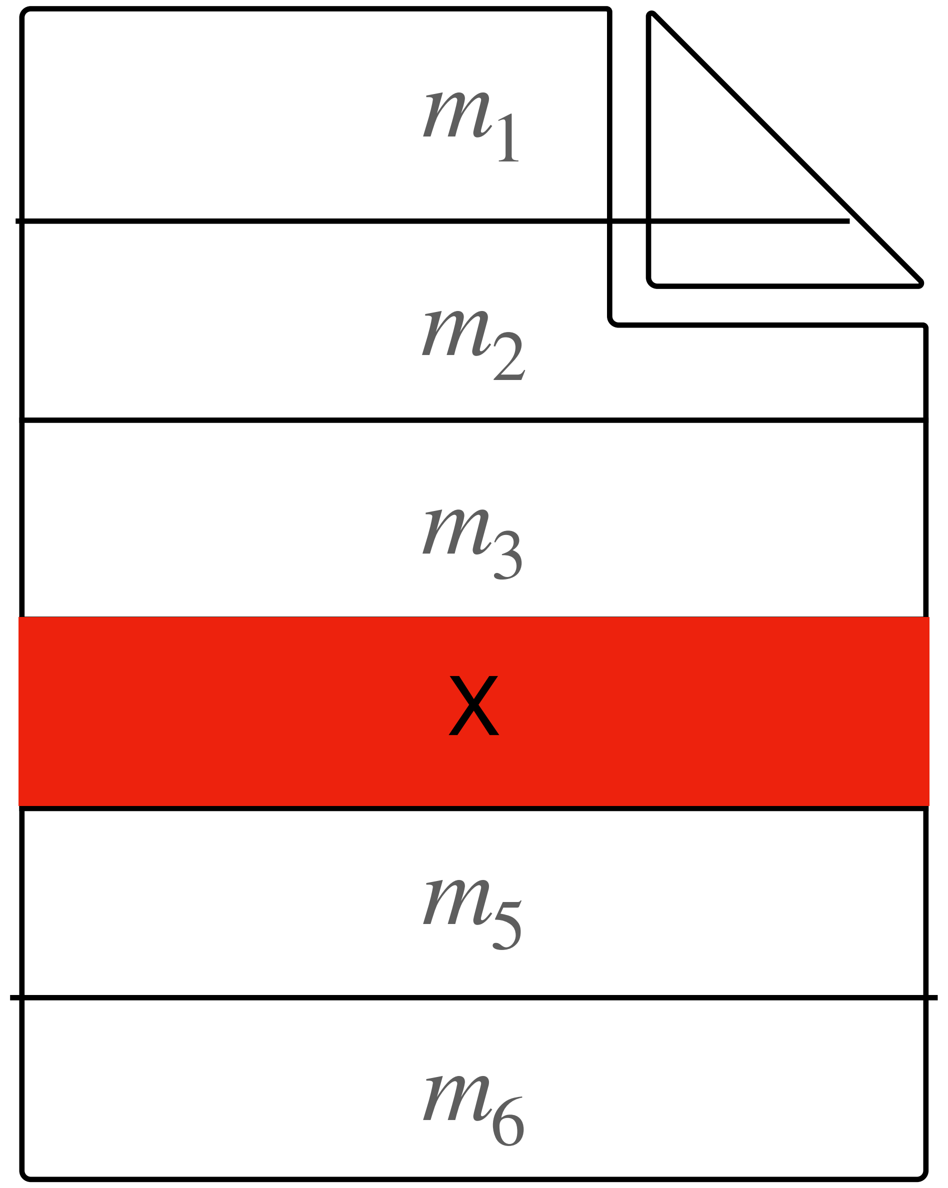
3) $Hash(h'_1|h'_2|h'_3)$
$Hash(h'_1|h'_4|h'_5)$
$Hash(h'_2|h'_4|h'_6)$
$Hash(h'_3|h'_5|h'_6)$



T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to locate modifications

Document



Signature

$$T \left\{ \begin{array}{l} Hash(h_1 \mid h_2 \mid h_3) \\ Hash(h_1 \mid h_4 \mid h_5) \\ Hash(h_2 \mid h_4 \mid h_6) \\ Hash(h_3 \mid h_5 \mid h_6) \\ h^* \quad Hash(m) \end{array} \right.$$

σ Sign(, T)

d-CFF

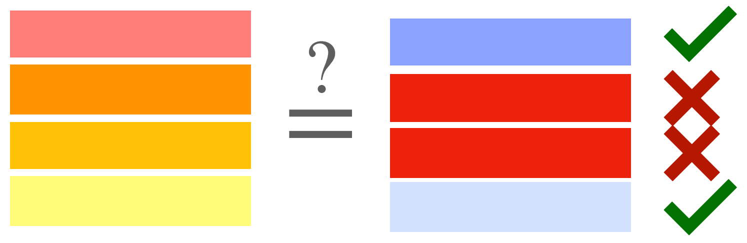
1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

Verification

1) $Verify(T, \sigma,)$ OK?

2) $h^* \stackrel{?}{=} Hash(m')$

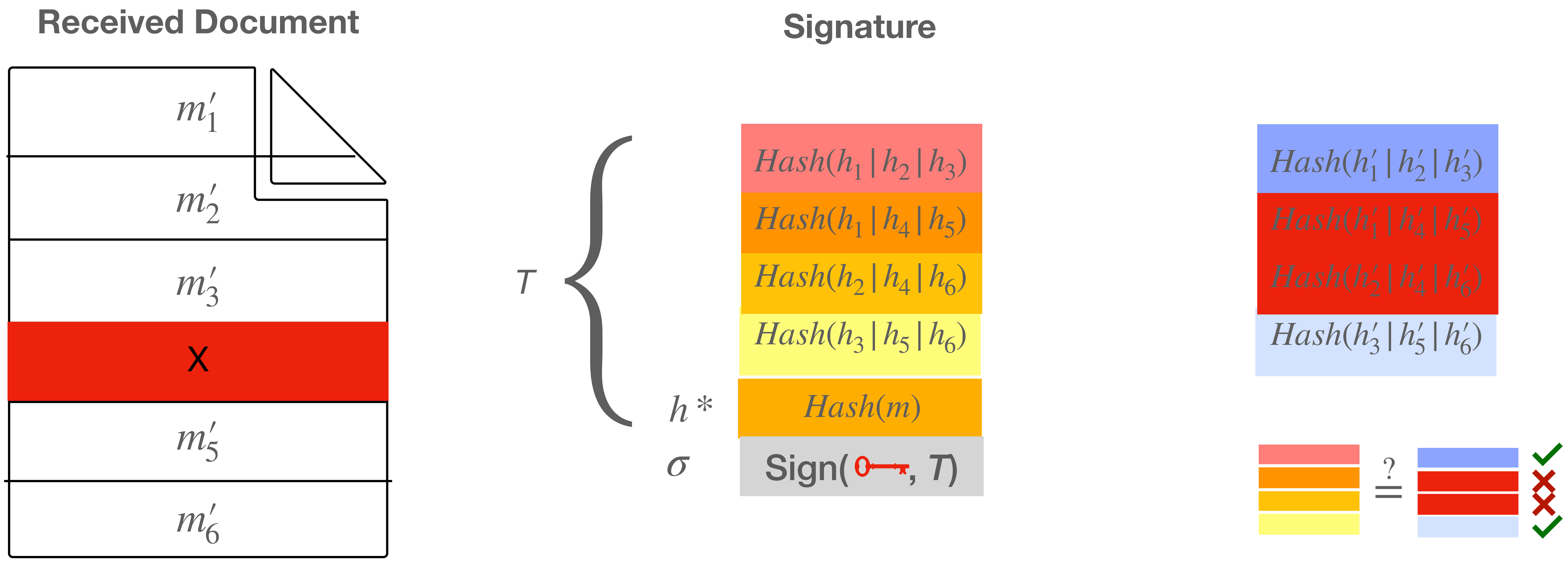
3) $Hash(h'_1 \mid h'_2 \mid h'_3)$
$Hash(h'_1 \mid h'_4 \mid h'_5)$
$Hash(h'_2 \mid h'_4 \mid h'_6)$
$Hash(h'_3 \mid h'_5 \mid h'_6)$



T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to correct modifications

For small enough blocks, we can correct modifications:



T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

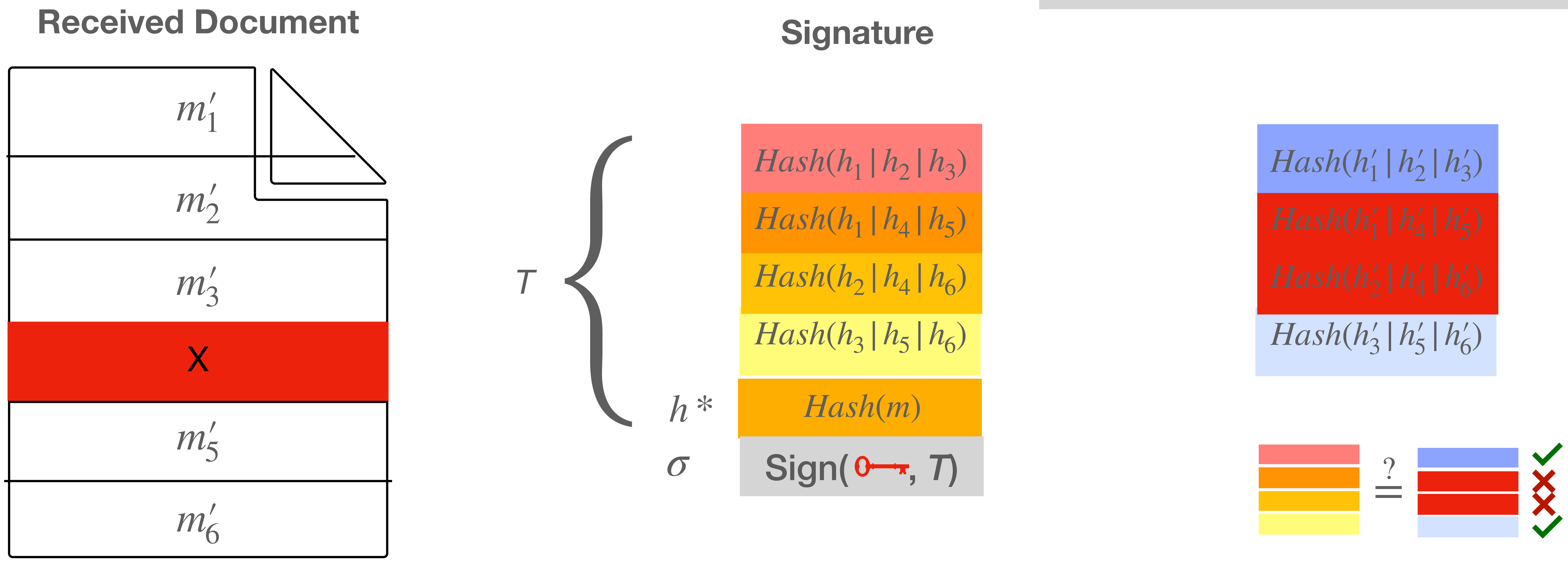
How to correct modifications

For small enough blocks, we can correct modifications:

1. Compute $h'_1 = \mathrm{Hash}(m'_1)$ and $h'_5 = \mathrm{Hash}(m'_5)$

For each possible value of m'_4:

2. Compute $h'_4 = \mathrm{Hash}(m'_4)$
3. Stop when the values below match



T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

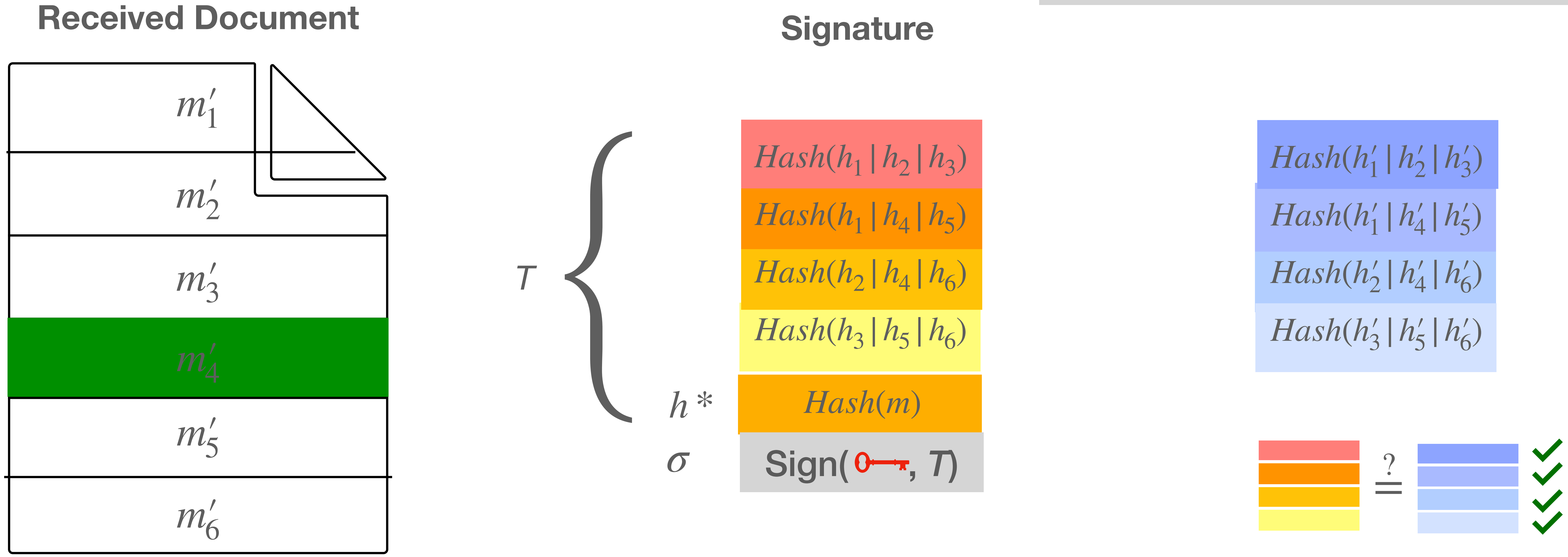
How to correct modifications

For small enough blocks, we can correct modifications:

1. Compute $h'_1 = \text{Hash}(m'_1)$ and $h'_5 = \text{Hash}(m'_5)$

For each possible value of m'_4:

2. Compute $h'_4 = \text{Hash}(m'_4)$
3. Stop when the values below match



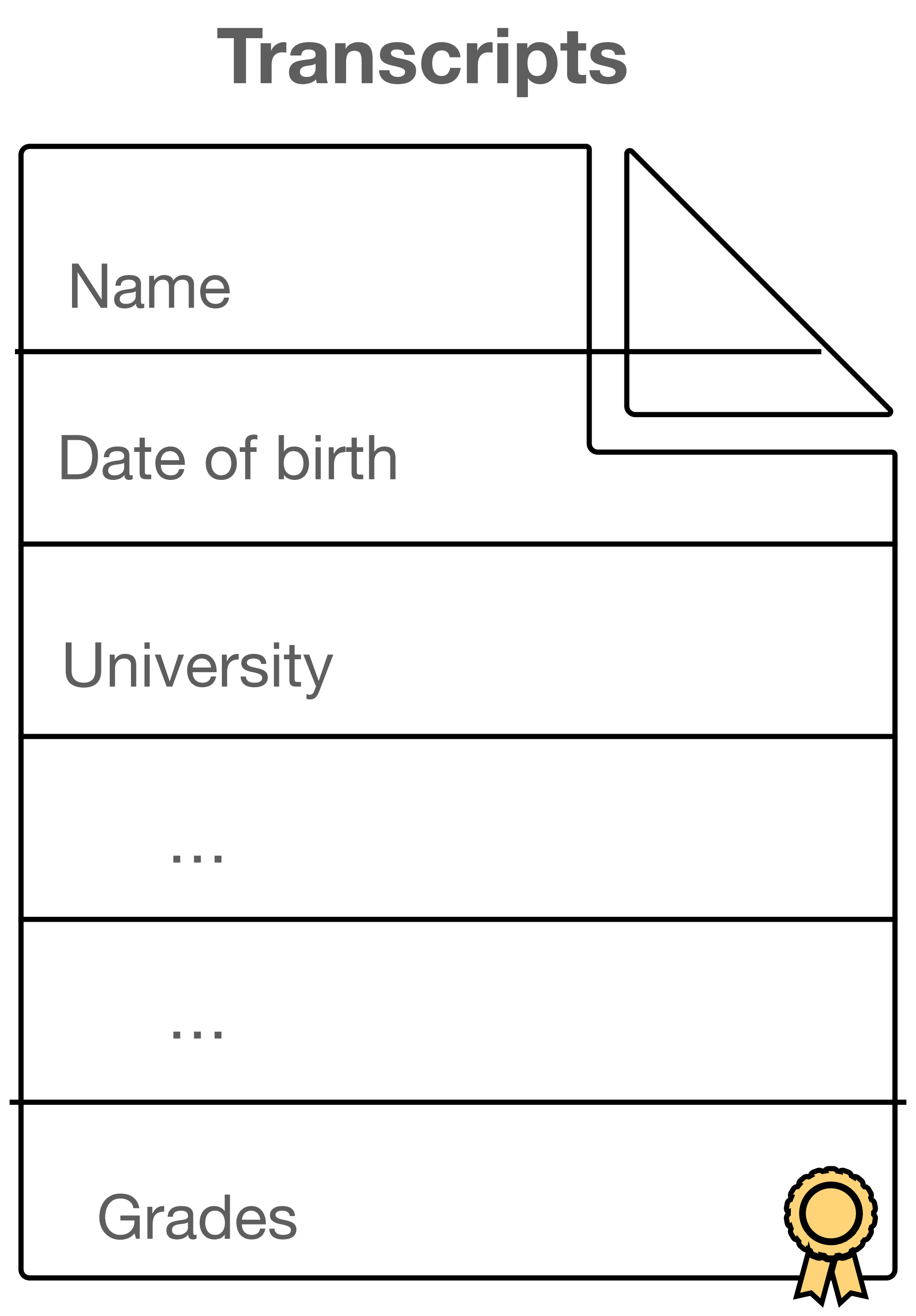
How to correct modifications

Extra details:

- We consider blocks of size at most ***s***
- Our algorithm keeps track of possible preimages of the modified blocks
- The probability of collision is very small, since ***s*** is small
  - For SHA-256 and s = 20, the probability is $\approx 3.70 \times 10^{-68}$
  - We experimentally verified that SHA-256 has no collision for s = 20
- The correction algorithm computes $O(d2^s + n)$ hash calculations
- These schemes are existentially unforgeable

How to ~~redact~~ private data

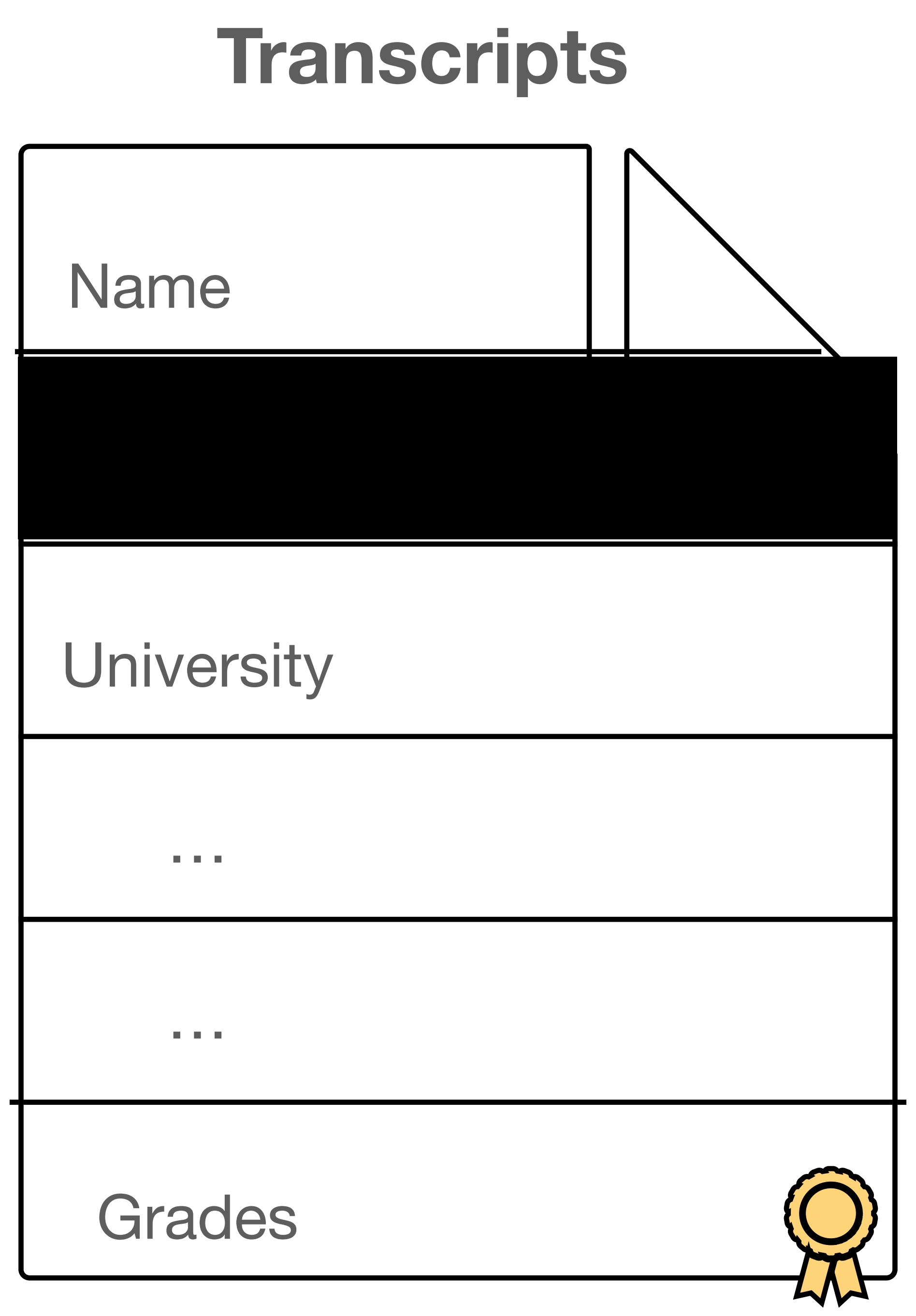
Redactable Signature



T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

How to ~~redact~~ private data

Redactable Signature



Rules:

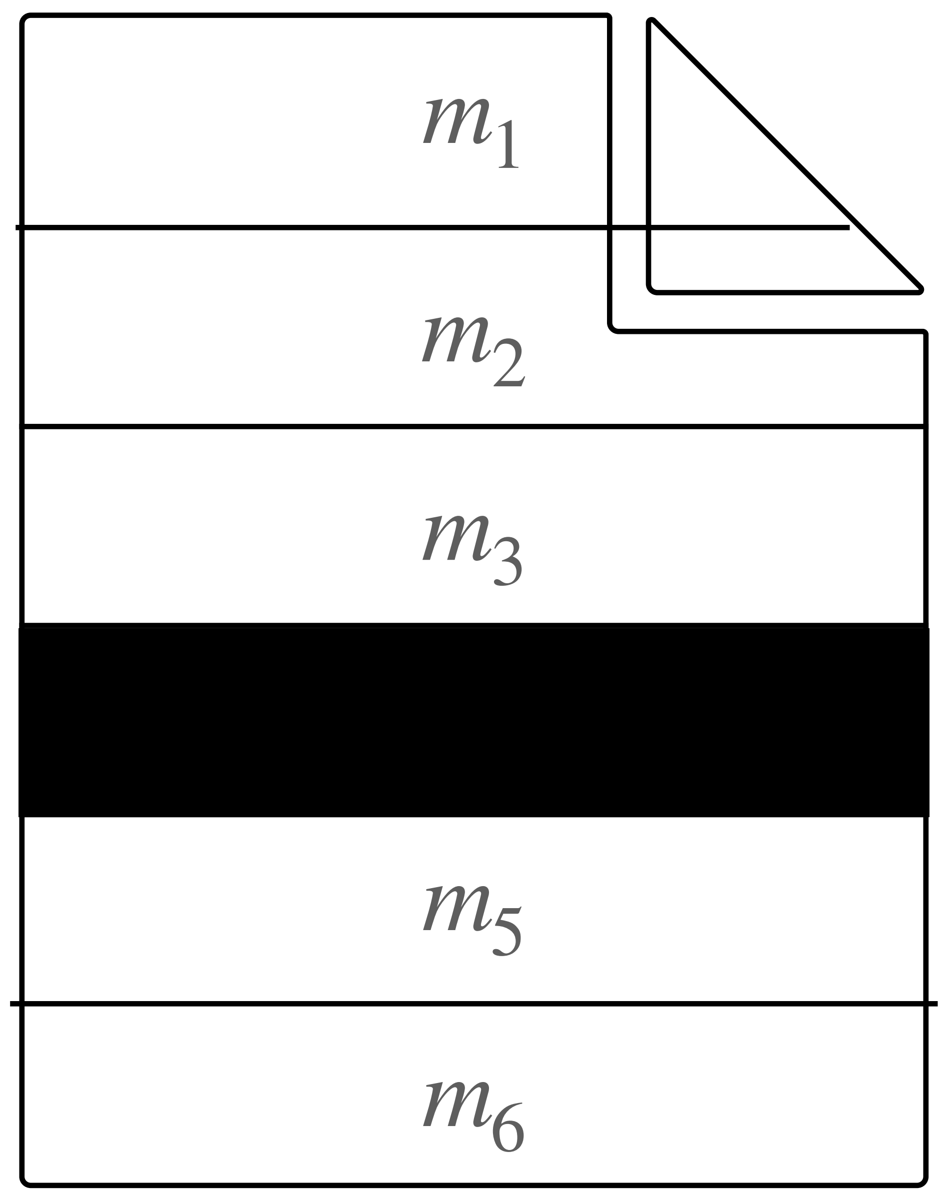
- Hide content without invalidating signature
- Should not be able to correct redacted blocks
- Should not leak information about them.

Our first two schemes are not suitable for this application

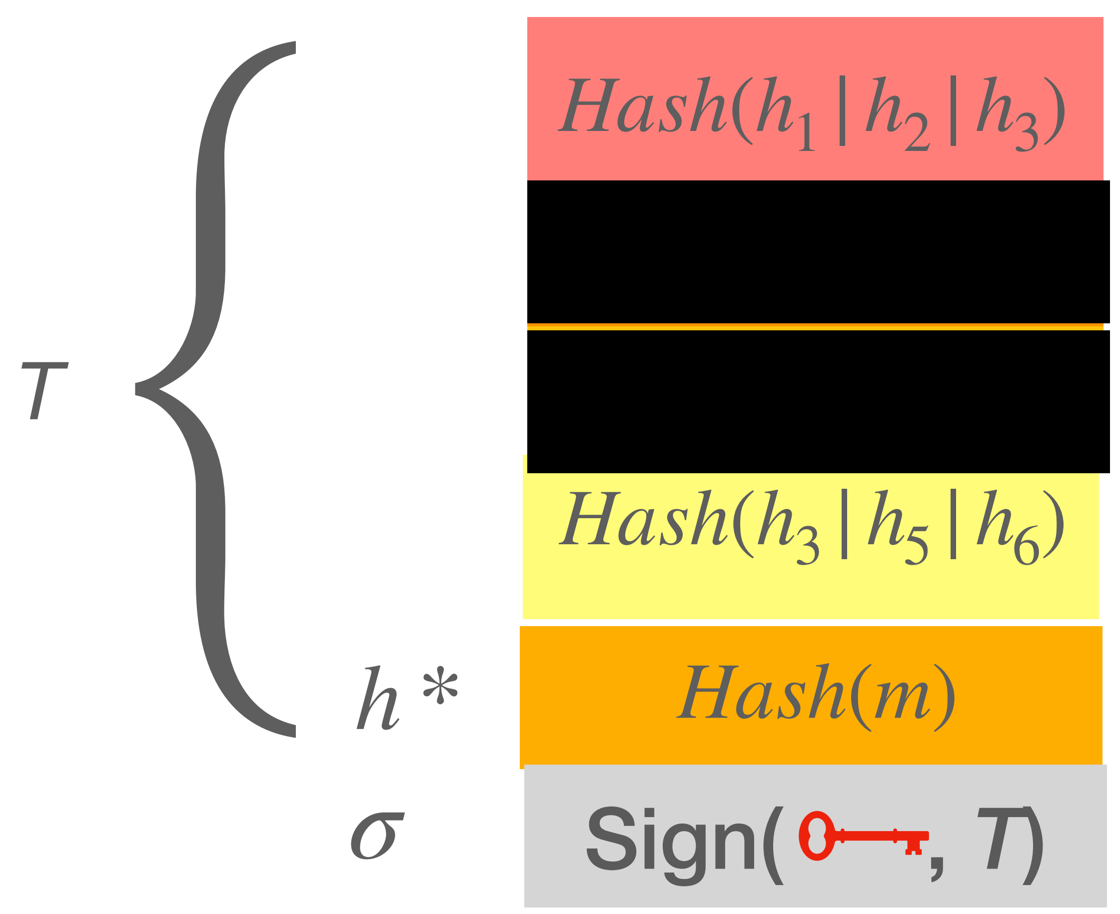
T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

How to ~~redact~~ private data

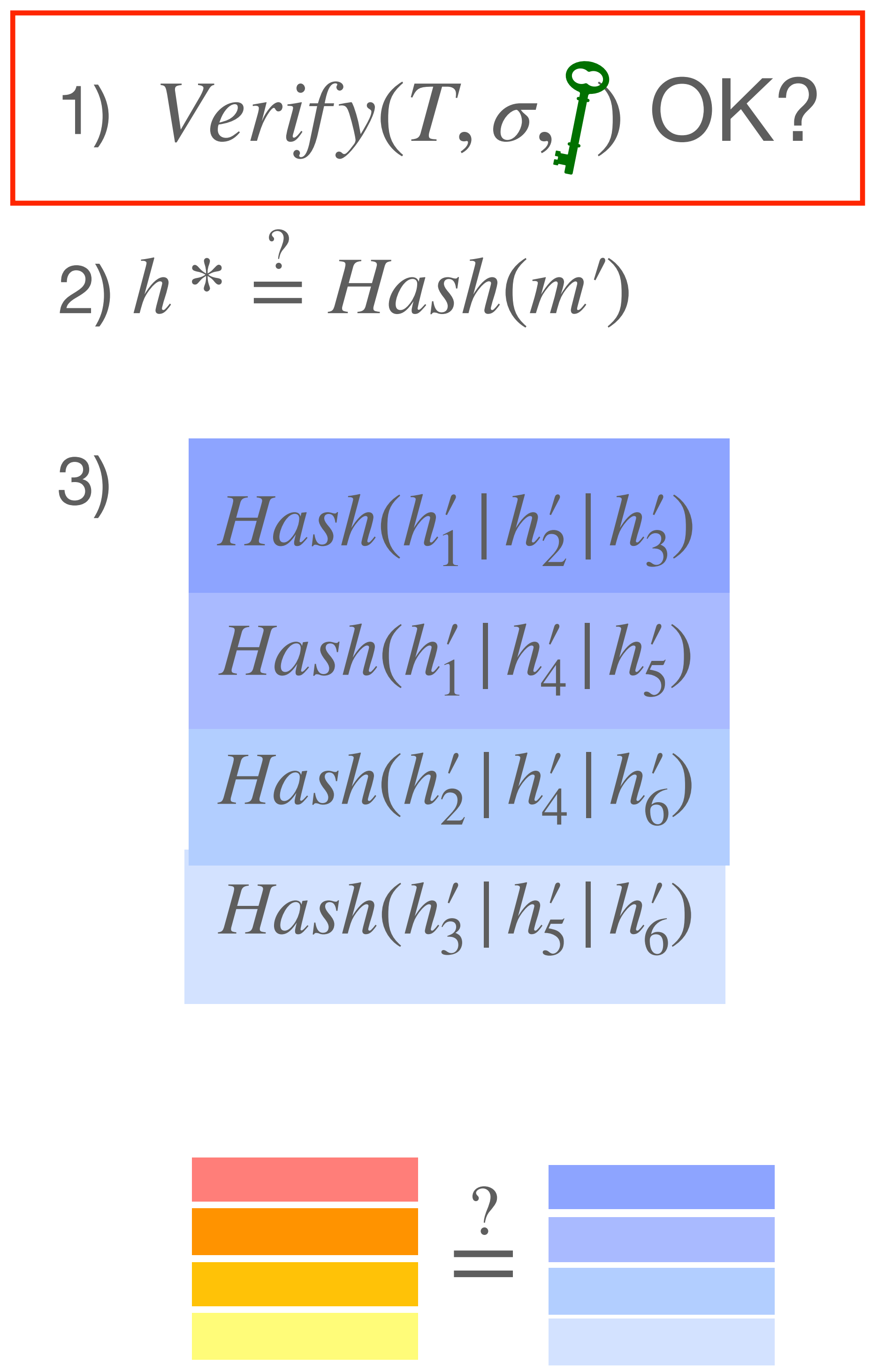
Document



Signature



Verification

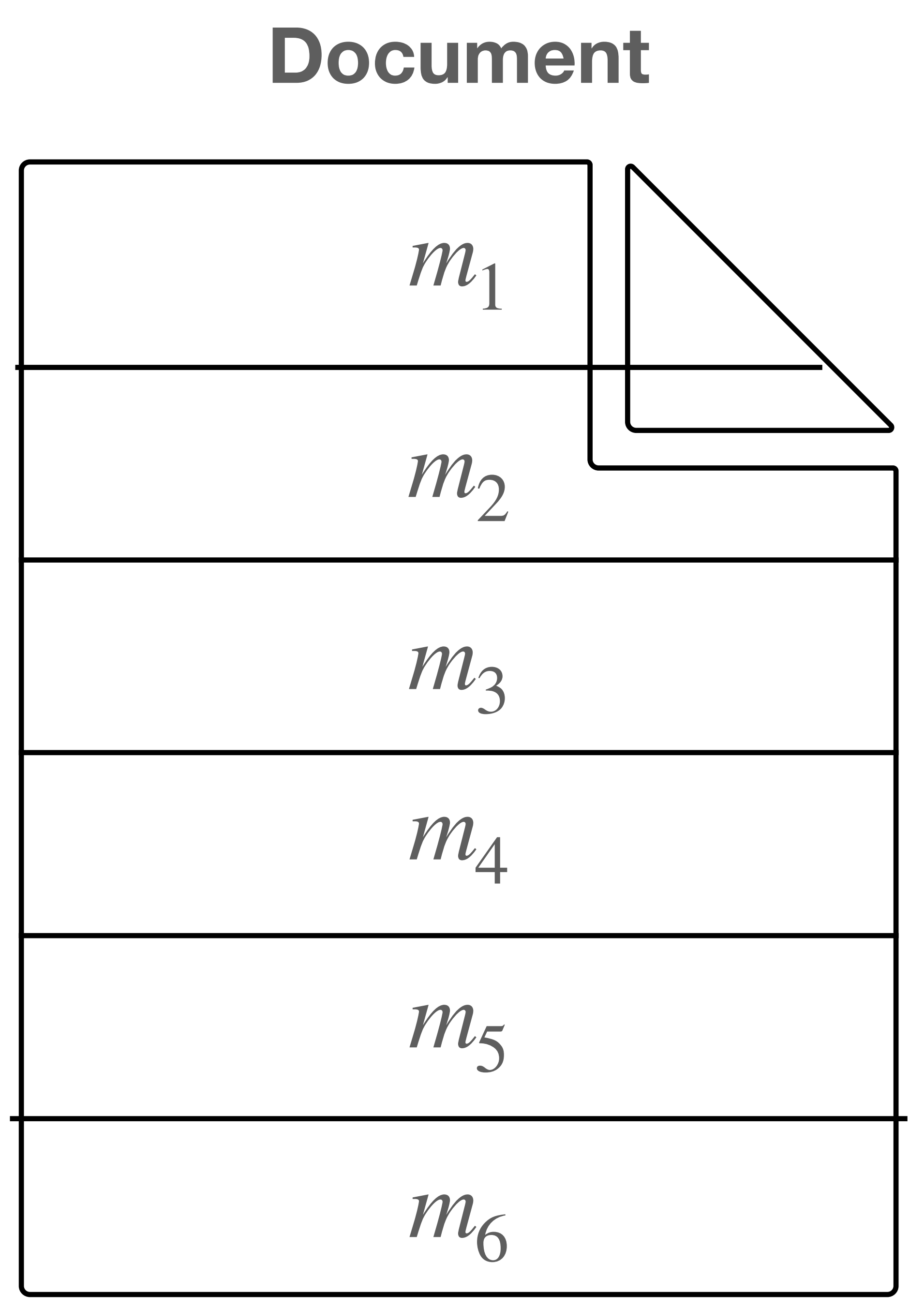


***d*-CFF**

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, R. F. Custodio, D. Panario. *Locating modifications in signed data for partial data integrity*. Information Processing Letters, 2015.

How to ~~redact~~ private data



Signature

T[1]	$Hash(h_1 \mid h_2 \mid h_3)$	$\sigma'[1]$	Sign(, T[1])
T[2]	$Hash(h_1 \mid h_4 \mid h_5)$	$\sigma'[2]$	Sign(, T[2])
T[3]	$Hash(h_2 \mid h_4 \mid h_6)$	$\sigma'[3]$	Sign(, T[3])
T[4]	$Hash(h_3 \mid h_5 \mid h_6)$	$\sigma'[4]$	Sign(, T[4])
T[5]	$Hash(m)$	$\sigma'[5]$	Sign(, T[5])

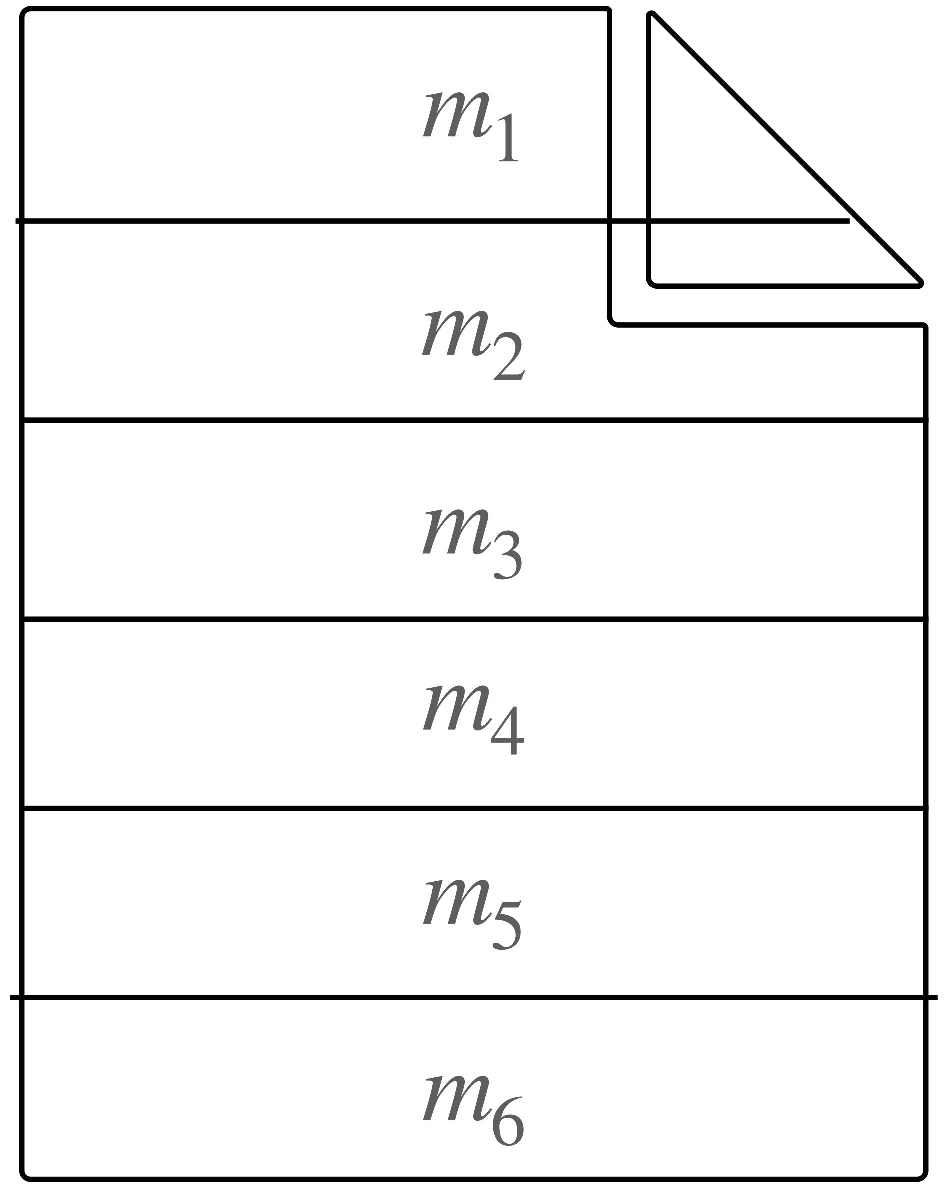
***d*-CFF**

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

How to ~~redact~~ private data

Document



Signature

$T[1]$	$Hash(h_1 \| h_2 \| h_3) \| r \| id(1,5)$	$\sigma'[1]$	Sign(, T[1])
$T[2]$	$Hash(h_1 \| h_4 \| h_5) \| r \| id(2,5)$	$\sigma'[2]$	Sign(, T[2])
$T[3]$	$Hash(h_2 \| h_4 \| h_6) \| r \| id(3,5)$	$\sigma'[3]$	Sign(, T[3])
$T[4]$	$Hash(h_3 \| h_5 \| h_6) \| r \| id(4,5)$	$\sigma'[4]$	Sign(, T[4])
$T[5]$	$Hash(m) \| r \| id(5,5)$	$\sigma'[5]$	Sign(, T[5])

$$\sigma = (\sigma', r)$$

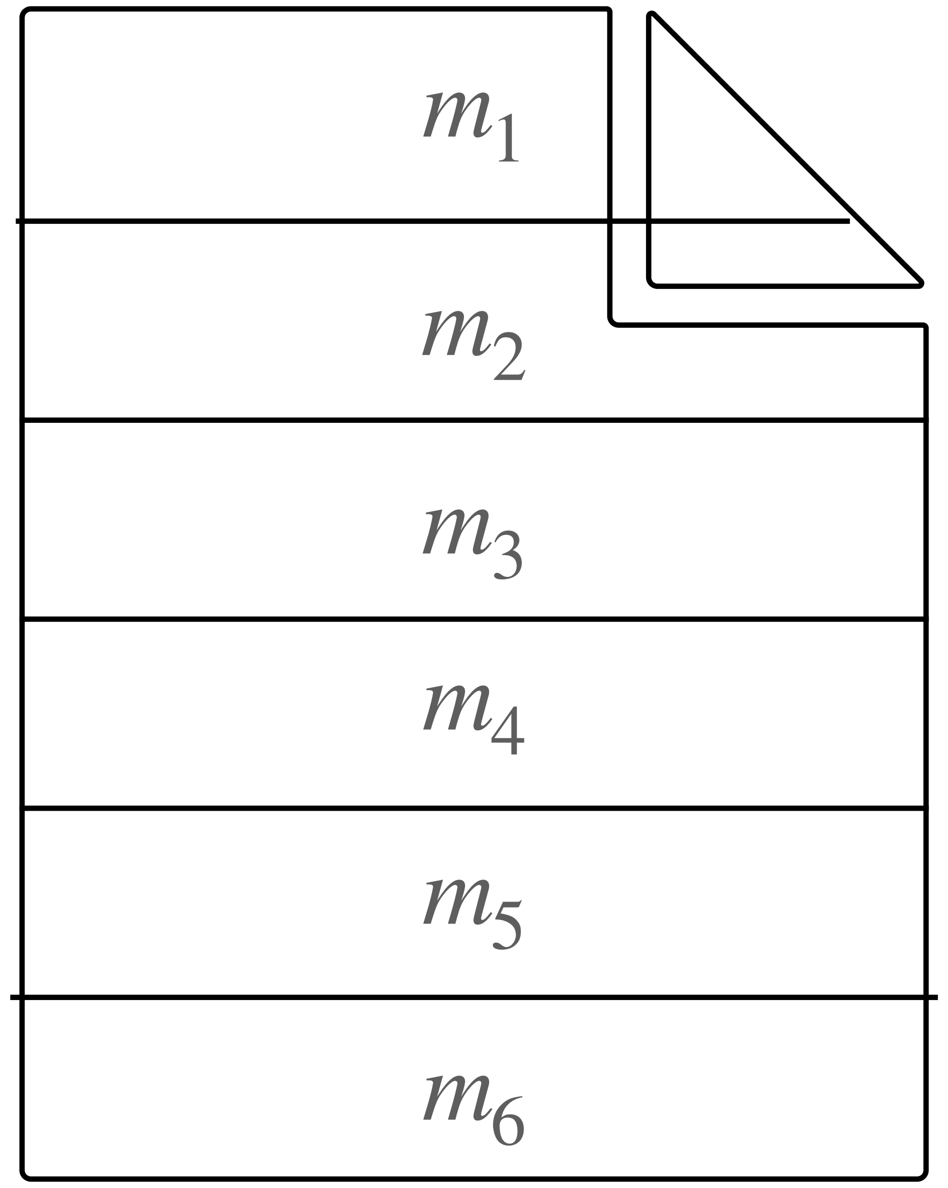
***d*-CFF**

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

How to ~~redact~~ private data

Document



T[1]	$Hash(h_1 \mid h_2 \mid h_3) \mid r \mid id(1,5)$
T[2]	$Hash(h_1 \mid h_4 \mid h_5) \mid r \mid id(2,5)$
T[3]	$Hash(h_2 \mid h_4 \mid h_6) \mid r \mid id(3,5)$
T[4]	$Hash(h_3 \mid h_5 \mid h_6) \mid r \mid id(4,5)$
T[5]	$Hash(m) \mid r \mid id(5,5)$

Signature

$\sigma'[1]$	Sign(, T[1])
$\sigma'[2]$	Sign(, T[2])
$\sigma'[3]$	Sign(, T[3])
$\sigma'[4]$	Sign(, T[4])
$\sigma'[5]$	Sign(, T[5])

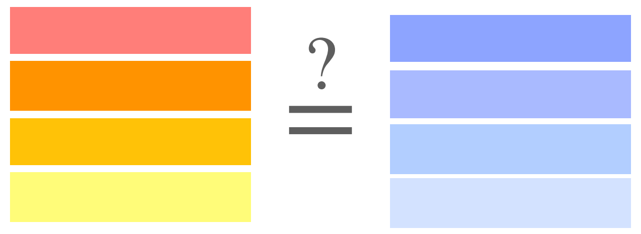
$\sigma = (\sigma', r)$

Verification

$Verify(h(m') \mid r \mid id(5,5), \sigma'[5],)$ OK?

T'[1]	$Hash(h_1 \mid h_2 \mid h_3) \mid r \mid id(1,5)$
T'[2]	$Hash(h_1 \mid h_4 \mid h_5) \mid r \mid id(2,5)$
T'[3]	$Hash(h_2 \mid h_4 \mid h_6) \mid r \mid id(3,5)$
T'[4]	$Hash(h_3 \mid h_5 \mid h_6) \mid r \mid id(4,5)$

$Verify(T'[i] \mid r \mid id(i,5), \sigma'[i],)$ OK?



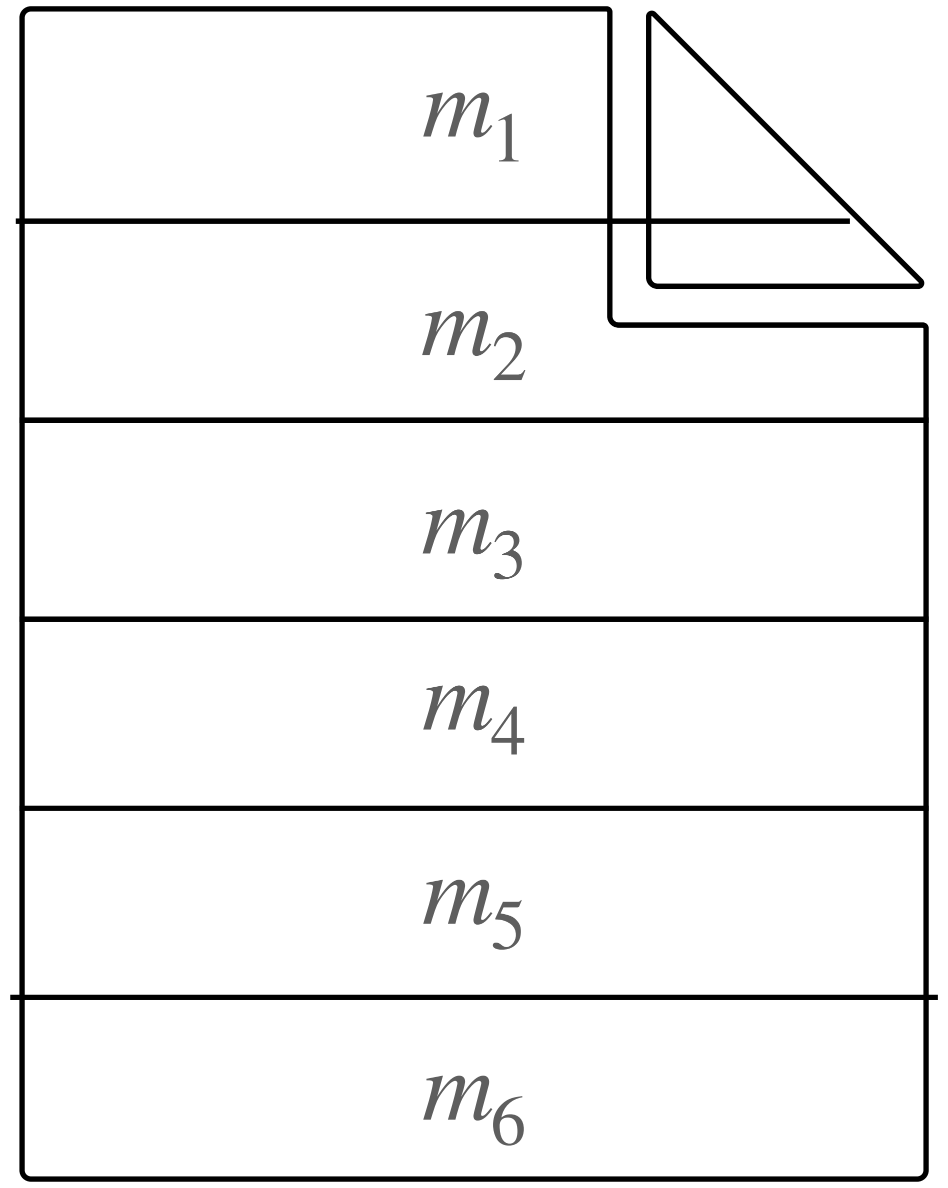
d-CFF

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

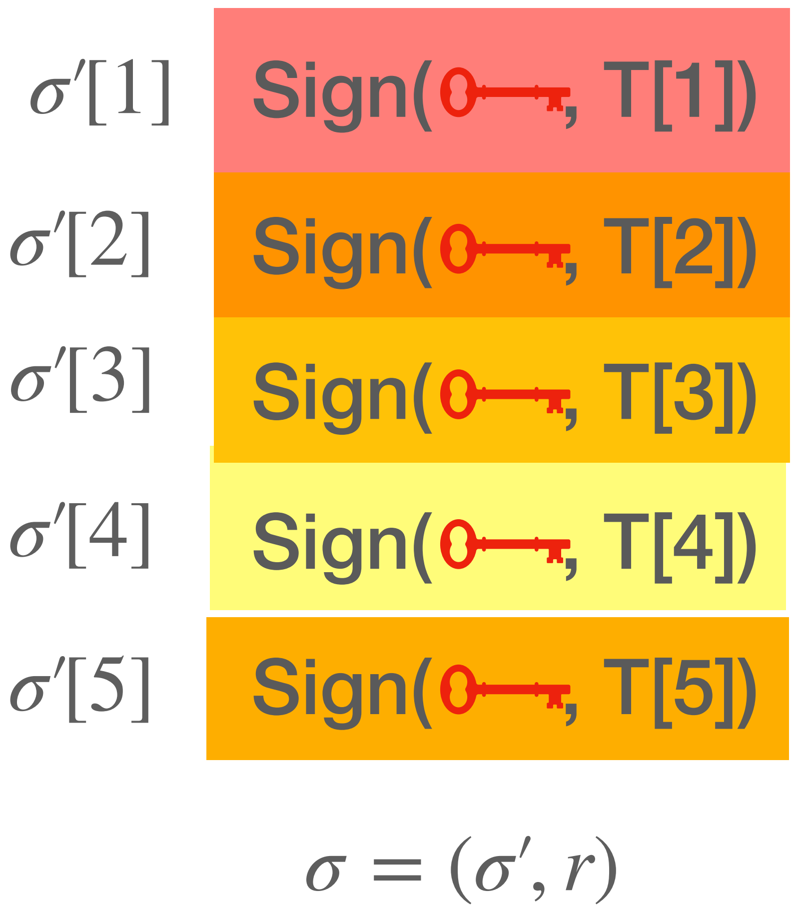
T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

How to ~~redact~~ private data

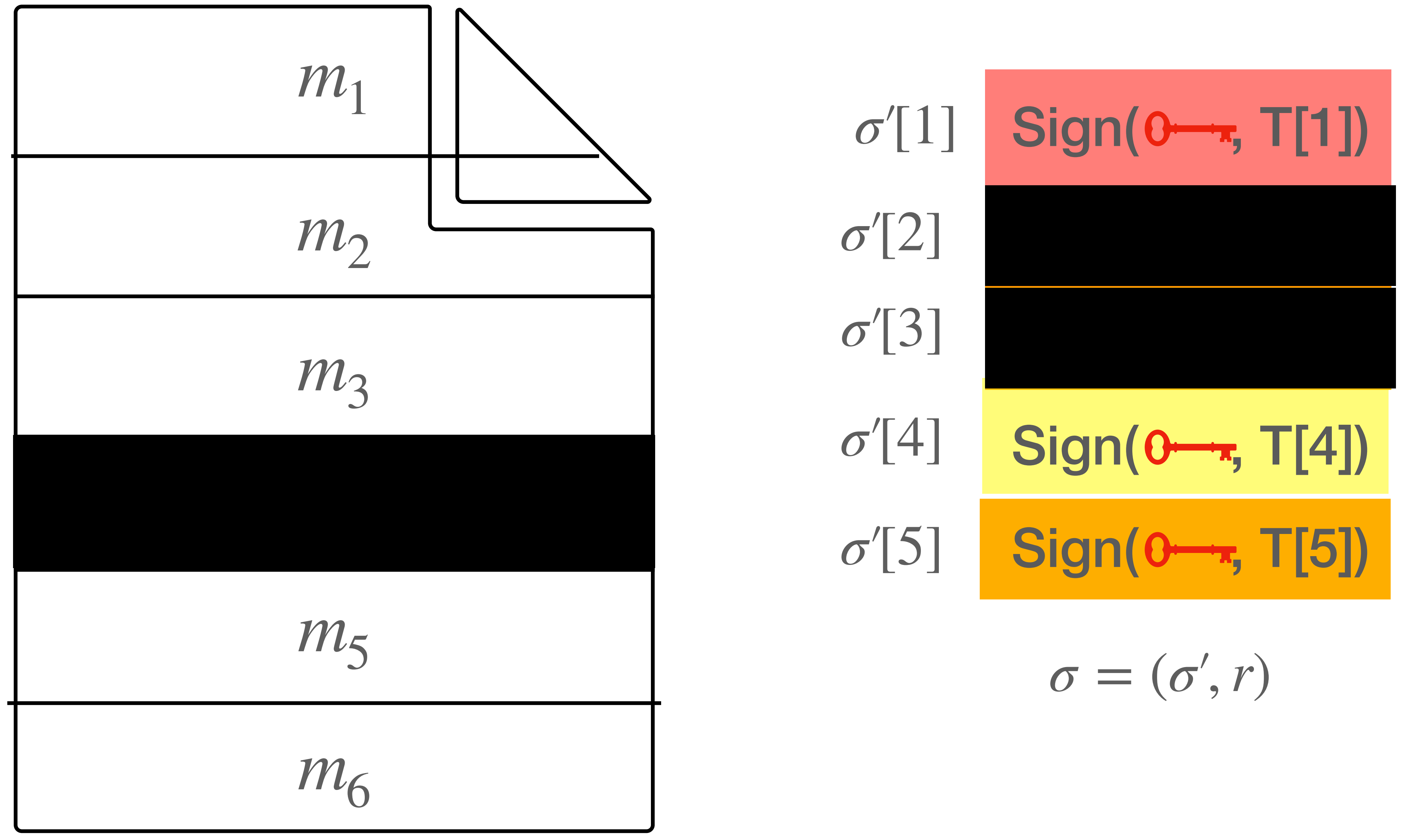
Document



Signature



Redaction

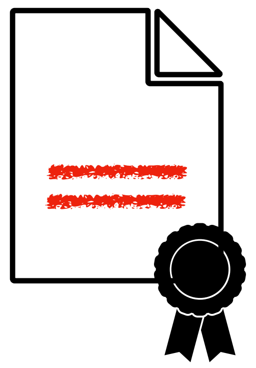


***d*-CFF**

1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

T.B. Idalino, L. Moura, C. Adams. *Modification Tolerant Signature Schemes: Location and Correction*. Progress in Cryptology - INDOCRYPT 2019

In this talk



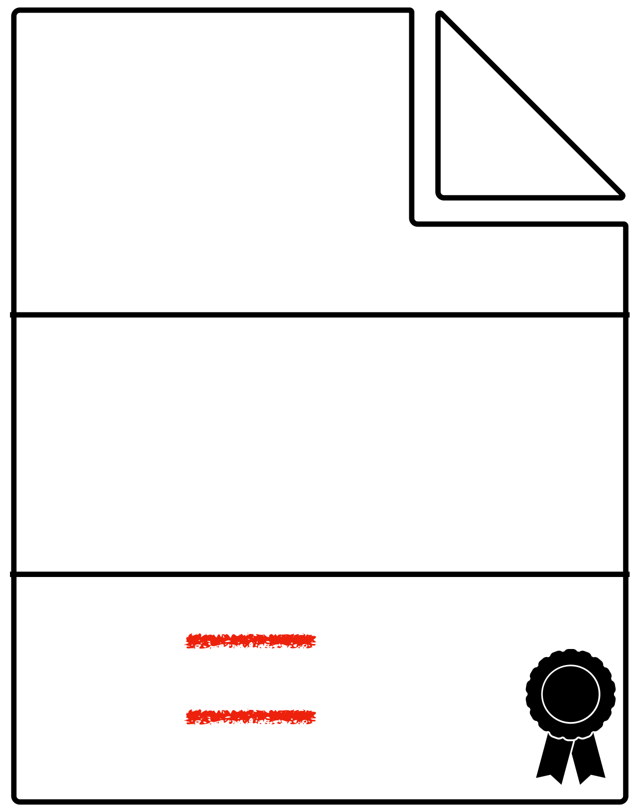
1	1	1	0	0	0
1	0	0	1	1	0
0	1	0	1	0	1
0	0	1	0	1	1

- A **modification-tolerant signature scheme** using **cover-free families**
- How to **locate** modifications.
- How to **correct** modifications.
- How to guarantee **privacy** of redacted data.

How can this be improved?

Variable CFFs

Acceptable modifications



- Traditional *d*-CFFs:
  - Any combination of *d* modifications is allowed.
- Variable CFFs:
  - We specify a set $\mathcal{S}$ with the allowed modifications (we call them $\mathcal{S}$-CFFs).

Variable CFFs

What are the benefits?

- Less requirement for coverage might give us $\mathscr{S}$-CFFs with less rows than d-CFFs.
  - This means less tests, smaller signatures, etc.
- We are working on constructions and bounds for $\mathscr{S}$-CFFs.

What else?

- Batch verification of signatures
- Aggregation of signatures *
- One-time signature schemes resistant against attacks by quantum computers
- Broadcast communication **

* T.B. Idalino, L. Moura. *Nested Cover-Free Families for Unbounded Fault-Tolerant Aggregate Signatures*. Theoretical Computer Science, 2021.
** T.B. Idalino, L. Moura. *Embedding Cover-Free Families and Cryptographical Applications.* Advances in Mathematics of Communications, 2018.

Thank you!