



IBM Research, Zurich

Anonymous Credentials on a Standard Java Card

Thomas Gross

Patrik Bichsel, Jan Camenisch, Victor Shoup

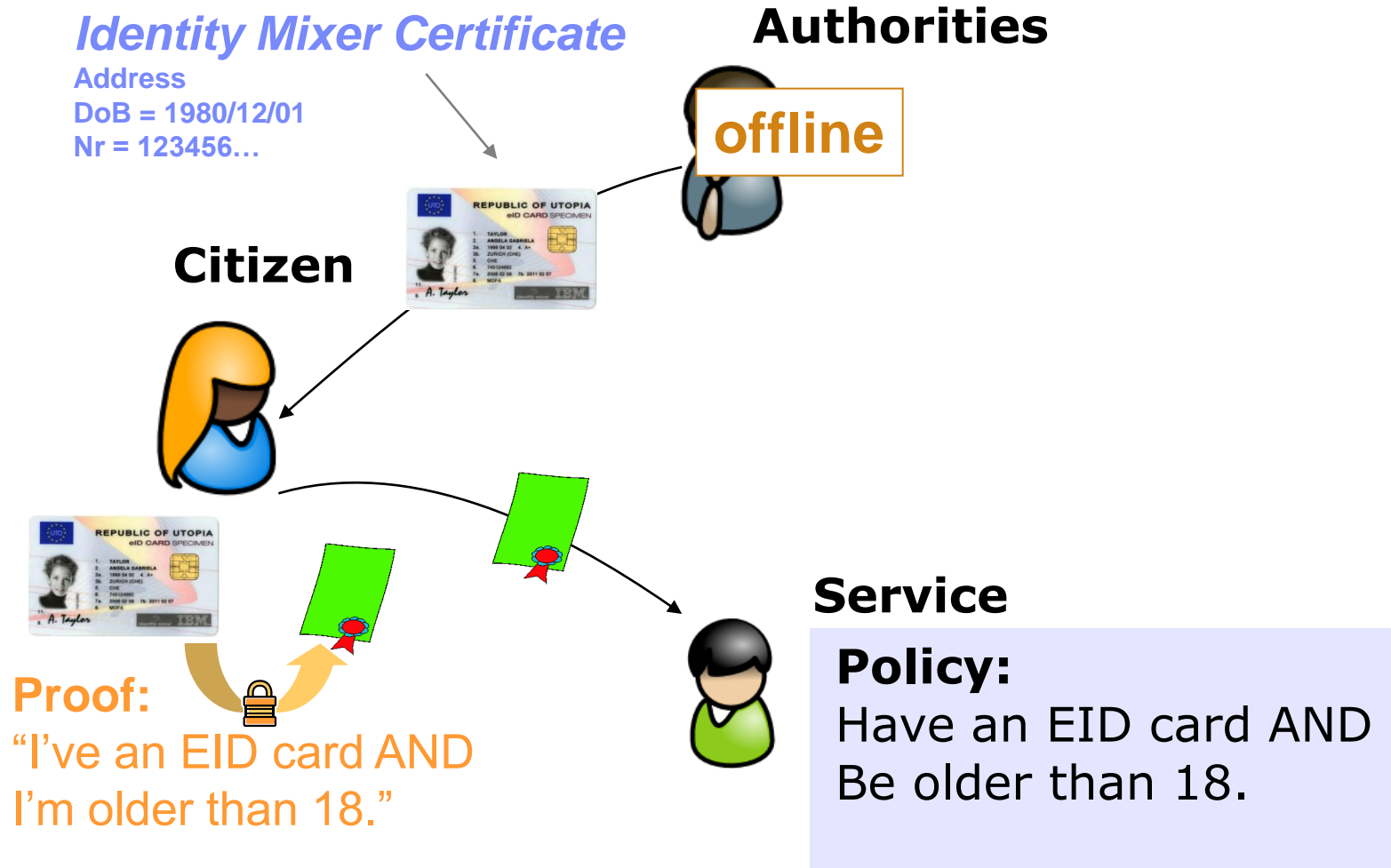
IBM's BlueZ Group for Strong Authentication

joint work with
supported by

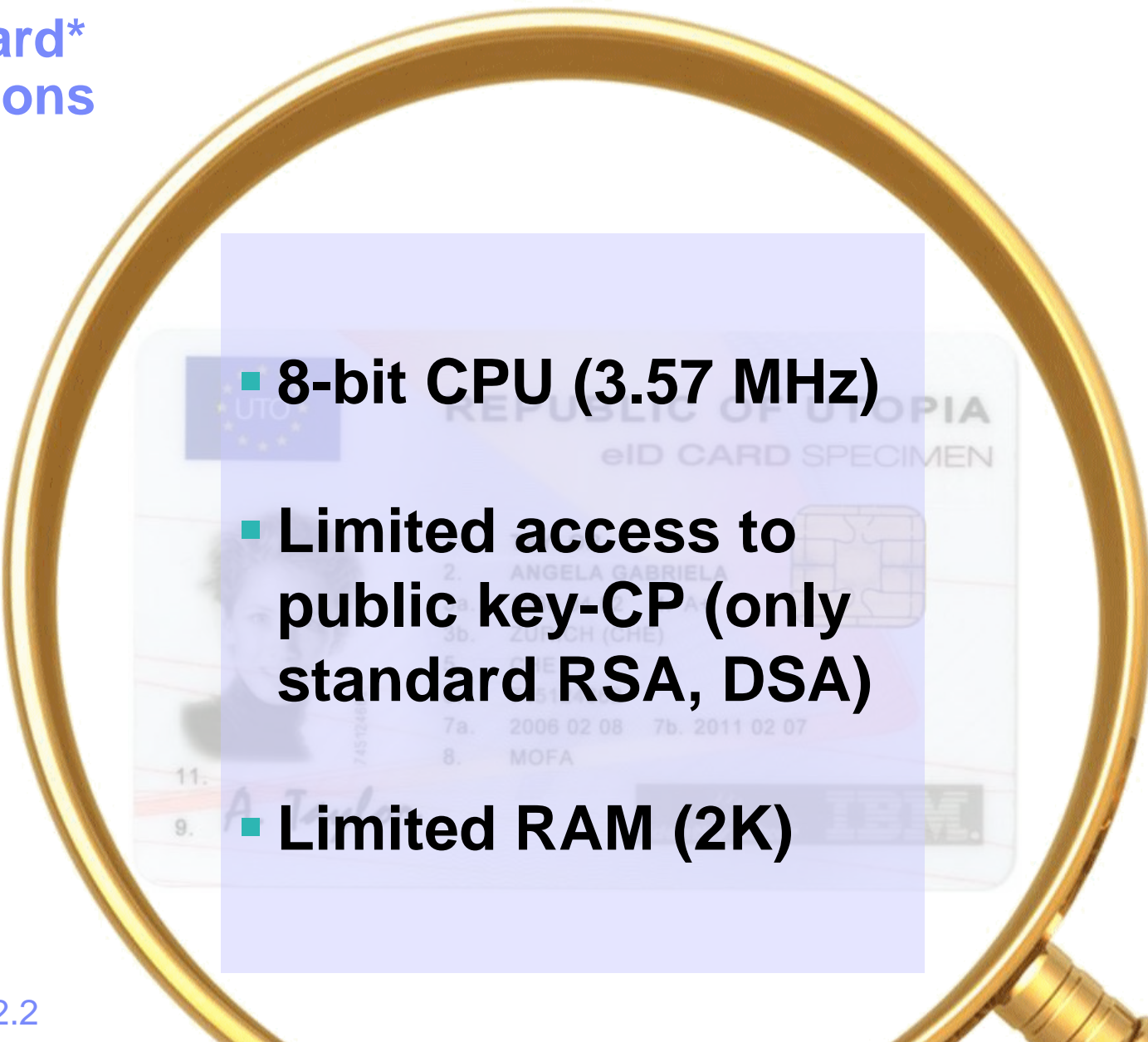
Overview

- **Introduction**
- **Caménisch-Lysyanskaya Signatures**
- **Problem Statement**
- **Key Ideas**
- **Results**

Example: Age Proof with Strong Privacy



Java Card* Limitations

- 
- **8-bit CPU (3.57 MHz)**
 - **Limited access to public key-CP (only standard RSA, DSA)**
 - **Limited RAM (2K)**

*: JCOP 41/v2.2

Basis: Camenisch-Lysyanskaya Signatures

[Camenisch & Lysyanskaya '01]

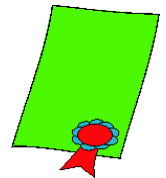
Public key of signer: RSA modulus n and $a_i, b, d \in \mathbb{Q}\mathbb{R}_n$

Secret key: factors of n

Signature of L attributes $m_1, \dots, m_L \in \{0,1\}^\ell$: (c, e, s)

For random prime $e > 2^\ell$ and integer $s \approx n$, compute c such that

$$d = a_1^{m_1} \cdot \dots \cdot a_L^{m_L} \cdot b^s \cdot c^e \pmod{n}$$



Theorem: Signature scheme is secure against adaptively chosen message attacks under SRSA assumption.

[SRSA: Barić & Pfitzmann '97 and Fujisaki & Okamoto '97]

Basis: Camenisch-Lysyanskaya Signatures

[Camenisch & Lysyanskaya '01]

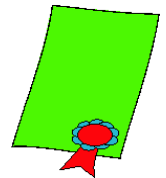
Public key of signer: RSA modulus n and $a_i, b, d \in QR_n$

Secret key: factors of n

Signature of L attributes $m_1, \dots, m_L \in \{0,1\}^\ell$: (c, e, s)

For random prime $e > 2^\ell$ and integer $s \approx n$, compute c such that

$$d = a_1^{m_1} \cdot \dots \cdot a_L^{m_L} \cdot b^s \cdot c^e \pmod{n}$$



Theorem: Signature scheme is secure against adaptively chosen message attacks under SRSA assumption.

[SRSA: Barić & Pfitzmann '97 and Fujisaki & Okamoto '97]

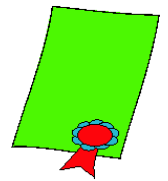
Basis: Camenisch-Lysyanskaya Signatures

[Camenisch & Lysyanskaya '01]

Signature of L attributes $m_1, \dots, m_L \in \{0,1\}^{\ell} : (c, e, s)$

For random prime $e > 2^{\ell}$ and integer $s \approx n$, compute c such that

$$d = a_1^{m_1} \cdot \dots \cdot a_L^{m_L} \cdot b^s \cdot c^e \pmod{n}$$



Abstractly requires computation of:

$$A_1^{x_1} \cdot \dots \cdot A_i^{x_i} \cdot \dots \cdot A_L^{x_L} \pmod{n}$$

where x_i correspond to attributes in the certificates
and potentially $|x_i| > |n|$

Problem Statement

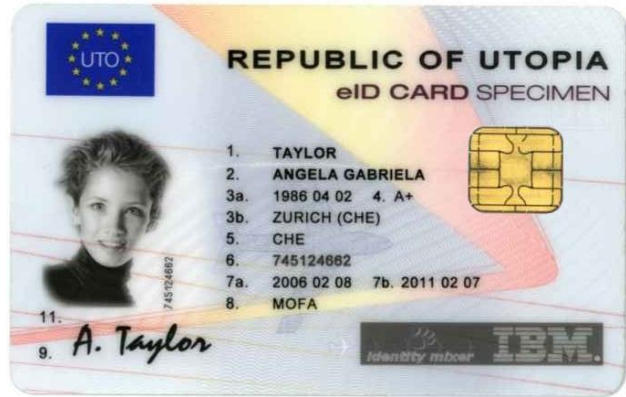
[Independent result:
Sterckx, ~~Palasch, De Rieck, Verduyn, Waele '09]~~

Run anonymous credential system autonomously and securely on a standard off-the-shelf Java Card.

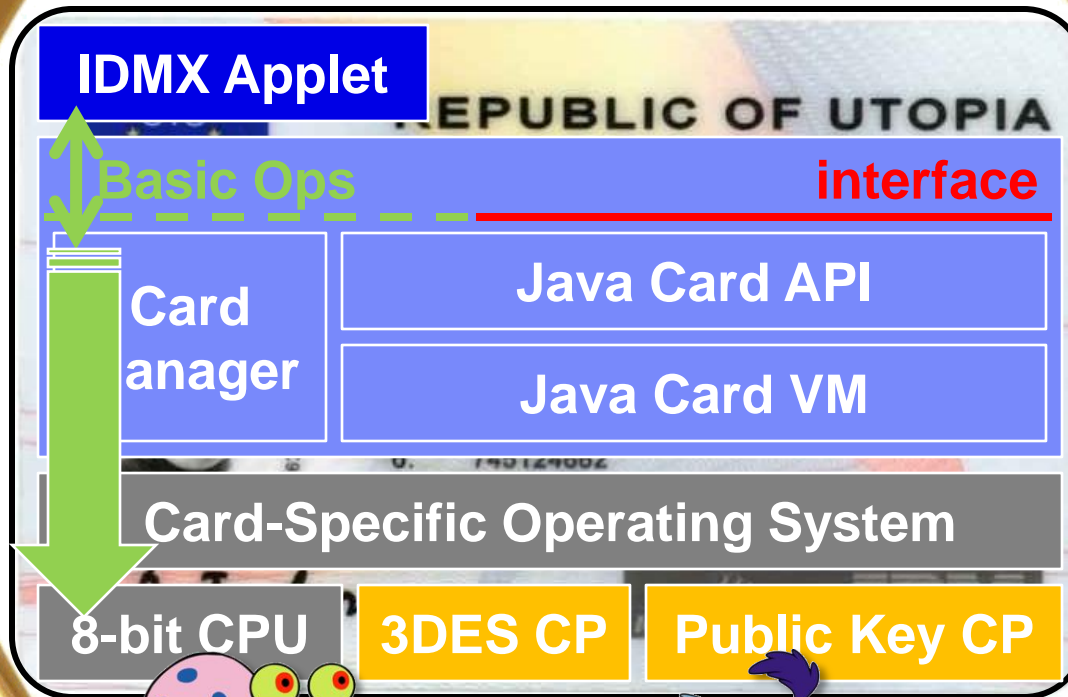
Security
CL-Signatures
Realistic keys

Autonomy
All data on card
Malicious terminal

Efficiency
Proof in seconds



Java Card Structure

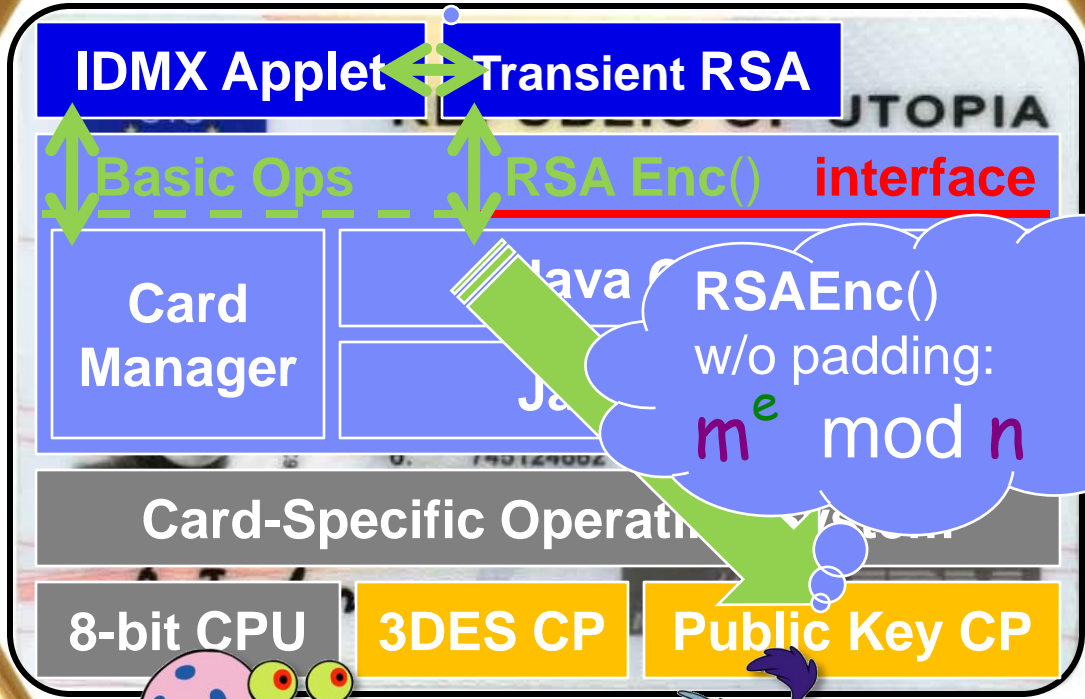


Source: Prof. Wolfgang Reif – chip cards

Java Card Structure

modExp() →
 RSAEnc()
 ☹ in EEPROM

modExp() →
 adapt key in RAM ☺
 RSAEnc()



RSAEnc()
 w/o padding:
 $m^e \text{ mod } n$



Source: Prof. Wolfgang Reif – chip cards

(Ab-)Using Standard RSA Interface

- Recall RSA Encryption: $m^e \bmod n$ (Limited size of e)
- ModExp()** with Big Exponents → Split exponents:

$$\begin{aligned}
 A_1^{x1} A_2^{x2} &= A_1^{x11 + x12 \cdot 2k} A_2^{x21 + x22 \cdot 2k} \bmod n \\
 &= A_1^{x11} (A_1^{2k})^{x12} A_2^{x21} (A_2^{2k})^{x22} \bmod n \\
 &= A_1^{x11} A_1'^{x12} A_2^{x21} A_2'^{x22} \bmod n
 \end{aligned}$$

- ModMultiply()**: RSA interface can only do exponentiation
→ Reduce multiply to **modExp()** by binomial formula:

$$A * B = ((A+B)^2 - A^2 - B^2)/2 \bmod n$$

Execution Times Full Proof (Including Communication)

Modulus	1280 bit	1536 bit	1984 bit
Precomputation	5203 ms	7828 ms	13250 ms
Compute A'	2125 ms	2906 ms	5000 ms
Compute T1	3078 ms	4922 ms	8250 ms
Policy-dependent	2234 ms	2625 ms	3298 ms
Compute 1 Response	562 ms	656 ms	828 ms
Total	7437 ms	10453 ms	16548 ms

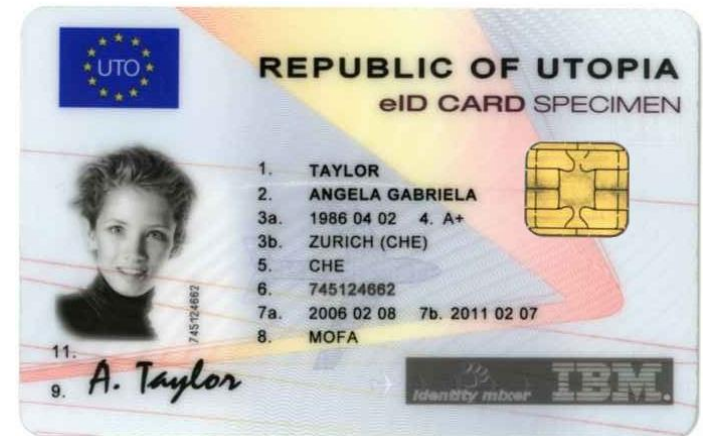
Results

- **Anonymous credential system on standard Java Card**
 - JCOP 41/v2.2
 - Future: Java Card 3.0 standard

- **Attributes:** Focus on proof of possession
 - rely on hardware tamper resistance for statement, and
 - detect / revoke broken cards.

- **Autonomous:** secure in face of untrusted terminal

- **Efficient:** 10 sec (at 1536 bits)
 - 7.5 sec pre-computation / 2.5 sec on-line



I'm happy to answer questions...

Identity Mixer Community Site:

idemix.wordpress.com

- See what's going on...
- Look at the spec...
- Download the library...

BACKUP

Detailed Performance Analysis: Modulus 1536 bit

Amortized Estimates over 1000 Ops, Upper Bound on Parameter Length, Percent Rounded Down

Function	Time	Ops	Percent
Multiplication	4'653 ms	9 Ops	39 %
↳ Addition	2988 ms	36 Ops	25 %
↳ ModSquare	243 ms	27 Ops	2 %
ModExp	4'308 ms	10 Ops	36 %
SRNG	1'088 ms	16 Ops	9 %
TRNG	815 ms	1 Op	6 %
Addition	581 ms	7 Ops	4 %
Digest	220 ms	10 Ops	1 %
Total	11'665 ms		

Recall: The Strong RSA Assumption

Flexible RSA Problem: Given RSA modulus n and $z \in QR_n$ find integers e and u such that


$$u^e = z \pmod{n}$$


(Recall: $QR_n = \{x : \text{exist } y \text{ s.t. } y^2 = x \pmod{n}\}$)

- Introduced by Barić & Pfitzmann '97 and Fujisaki & Okamoto '97
- Hard in generic algorithm model [Damgård & Koprowski '01]

Signature Scheme based on the SRSA I

[Camenisch & Lysyanskaya '02]

Public key of signer: RSA modulus n and $a_i, b, d \in QR_n$ 

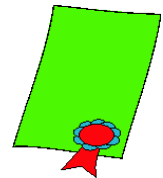
Secret key: factors of n 

To sign k messages $m_1, \dots, m_k \in \{0,1\}^\ell$:

- choose random prime $e > 2^\ell$ and integer $s \approx n$
- compute c such that

$$d = a_1^{m_1} \cdot \dots \cdot a_k^{m_k} b^s c^e \pmod{n}$$

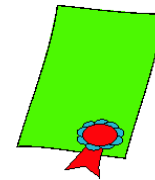
- signature is (c, e, s)



Signature Scheme based on the SRSA II

A signature (c, e, s) on messages m_1, \dots, m_k is valid iff:

- $m_1, \dots, m_k \in \{0,1\}^\ell$:
- $e > 2^\ell$
- $d = a_1^{m_1} \cdot \dots \cdot a_k^{m_k} b^s c^e \pmod n$

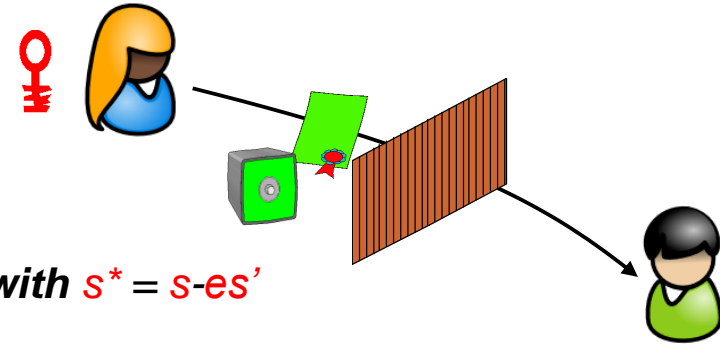


Theorem: Signature scheme is secure against adaptively chosen message attacks under SRSA assumption.

Proof of Knowledge of a Signature

Observe:

Let $c' = c b^{s'}$ mod n with random s'
 then $d = c'^e a_1^{m1} \cdot \dots \cdot a_k^{mk} b^{s^*}$ (mod n), with $s^* = s - es'$
 i.e., (c', e, s^*) is also a valid signature!



Therefore, to prove knowledge of signature on some m

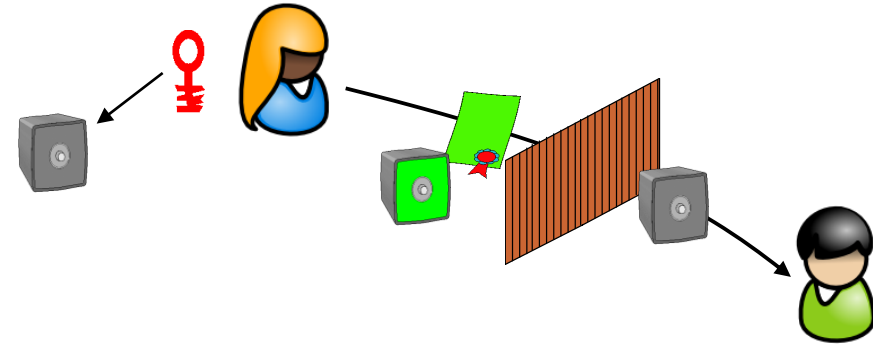
- provide c'
- PK $\{(e, m1, \dots, mk, s) : d := c'^e a_1^{m1} \cdot \dots \cdot a_k^{mk} b^s$
 $\wedge mi \in \{0,1\}^t \wedge e \in 2^{\ell+1} \pm \{0,1\}^t \}$

Proof of Knowledge of a Signature

Using second Commitment

assume second group n, a, b, n

2nd commitment $C = a_1^{sk} b^{s^*}$



To prove knowledge of signature on some m
provide c'

$PK\{(e, m_1, \dots, m_k, s, s^*)\} :$

$$C = a_1^{m_1} b^{s^*} \wedge d := \{c'^e a_1^{m_1} \cdot \dots \cdot a_k^{m_k} b^s\}$$