DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

RSA® CONFERENCE 2013
FEBRUARY 25 – MARCH 1
SAN FRANCISCO

Fair Exchange of Short Signatures without Trusted Third Party

Philippe Camacho
University of Chile

Digital Goods Economy

Enforcing Secure Transactions through a Trusted Third Party (TTP)

PayPal™

Google™ Checkout

amazon.com®

Problems with TTP

Anonymous Claims To Have Hacked 28,000 PayPal Passwords For Guy Fawkes Day

The Huffington Post | By Cavan Sieczkowski

Posted: 11/05/2012 11:15 am EST Updated: 11/05/2012 1:01 pm EST

Like 829 people like this.

Problems with TTP

PayPal

Privacy Policy

Last Update: Jul 13, 2010

Jump to section:

This policy describes the ways we collect, store, use and protect your personal information. You accept this policy and consent to such collection, storage and use when you sign up for or use our products, services or any other feature, technologies or functionalities offered by us on our website or through any other means (collectively "the PayPal Services"). We may amend this policy at any time by posting a revised version on our website. The revised version will be effective at the time we post it. In addition, if the revised version includes a substantial change, we will provide you with 30 days' prior notice by posting notice of the change on the "Policy Updates" page of our website. After this 30 days notice, you will be considered as having expressly consented to all amendments to this policy.

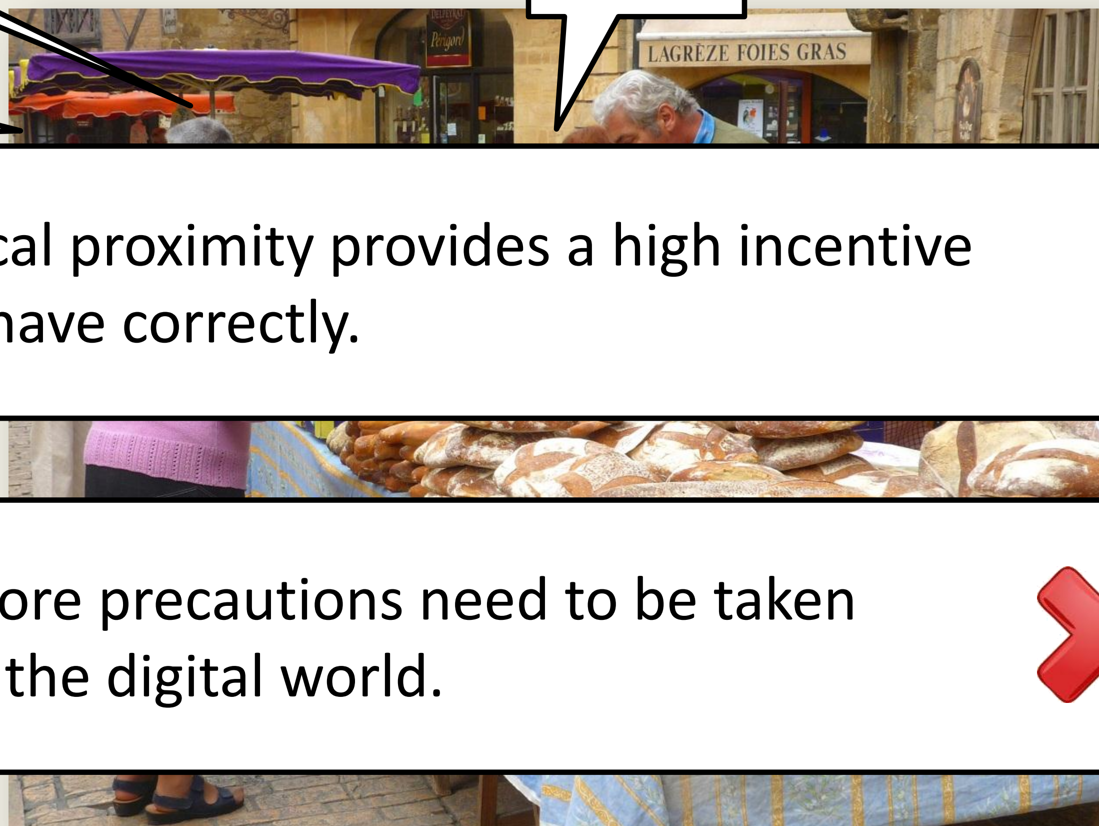
How we collect information about you

Fair Exchange in the Physical World is “easy”

Witness

Witness

Witness



Buyer

Physical proximity provides a high incentive to behave correctly.

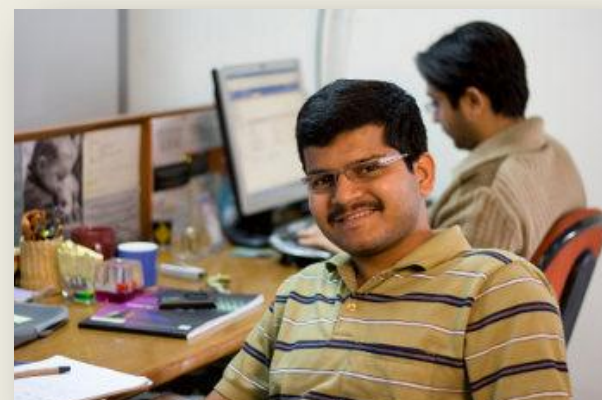
More precautions need to be taken in the digital world.

Modeling Transactions with Digital Signatures

The problem: Who starts first?
Impossibility Result **[Cleve86]**

Software License

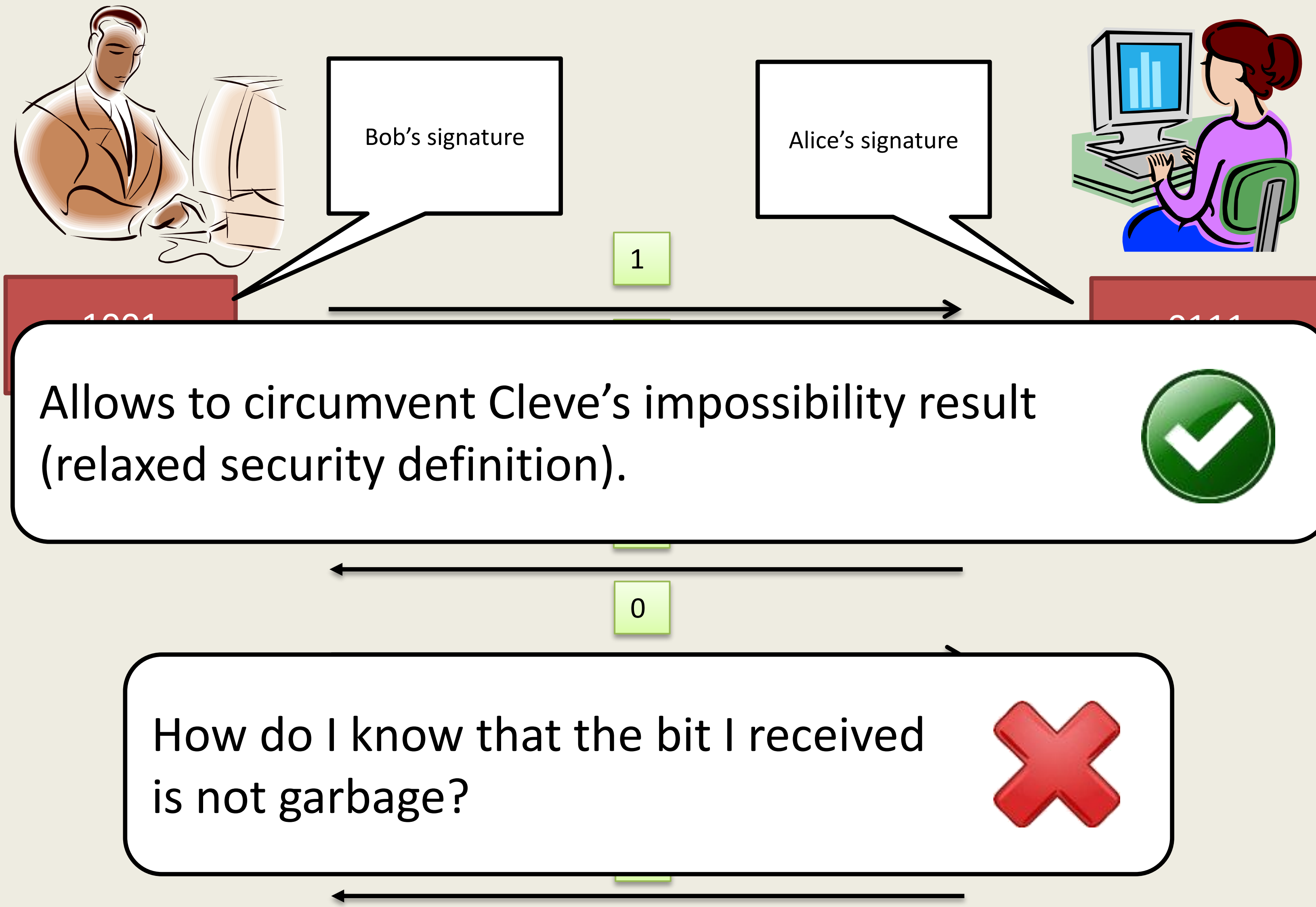
Buyer



Seller

Digital Check

Gradual Release of a Secret



Allows to circumvent Cleve's impossibility result (relaxed security definition).

How do I know that the bit I received is not garbage?

Our Construction

- Fair Exchange of Digital Signatures
- Boneh-Boyen [BB04] Short Signatures
- No TTP
- Practical

Contributions

- Formal definition of *Partial Fairness*

- Efficiency

	κ: Security Parameter	$\boldsymbol{\kappa = 160}$
# Rounds	$\kappa + 1$	161
Communication	$16\kappa^2 + 12\kappa$ bits	≈ 52 kB
# Crypto operations per participant	$\approx 30\kappa$	≈ 4800

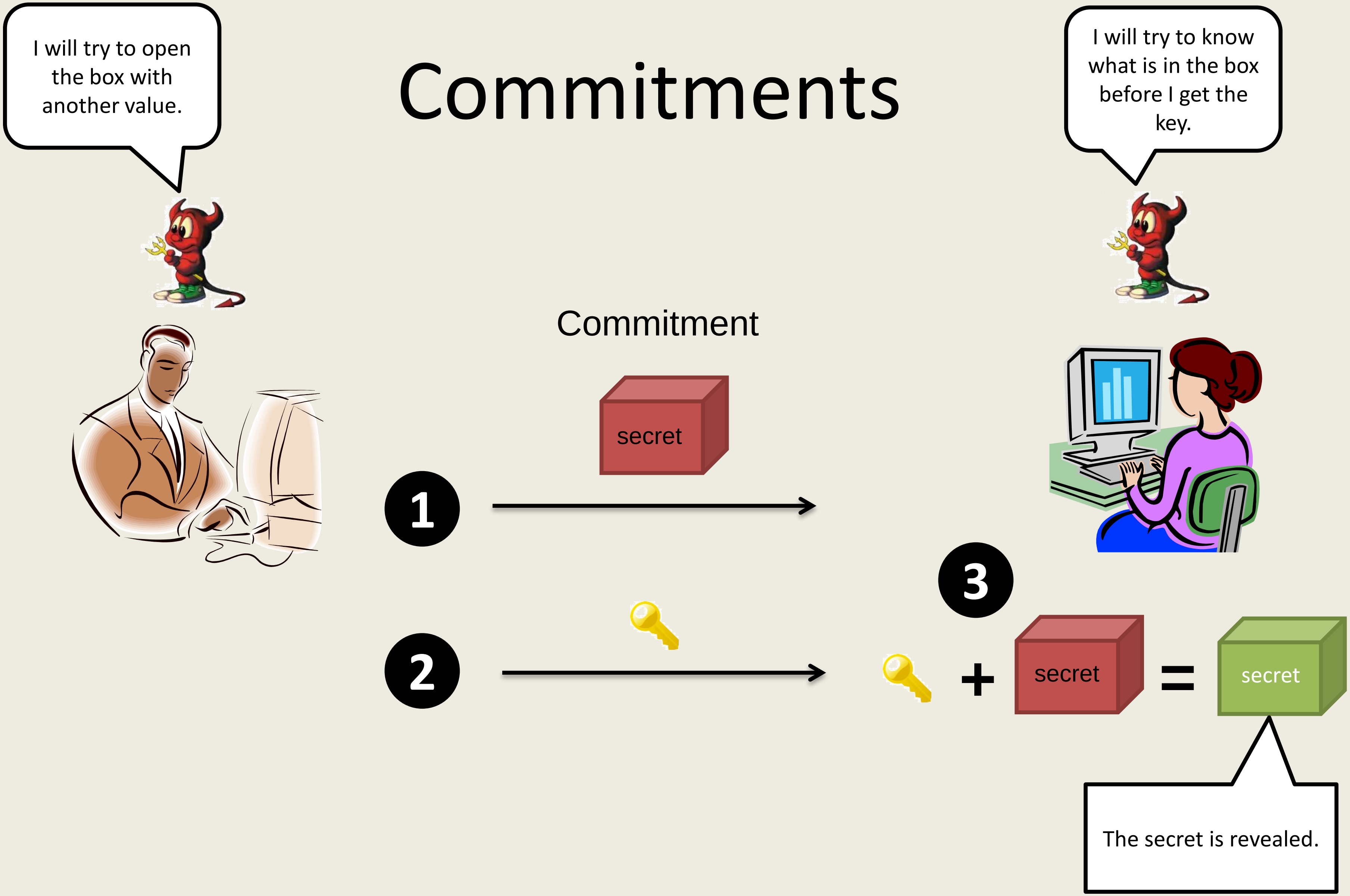
- First protocol for Boneh-Boyen signatures

Contributions

- NIZK argument to prove that a commitment encodes a **bit vector**.

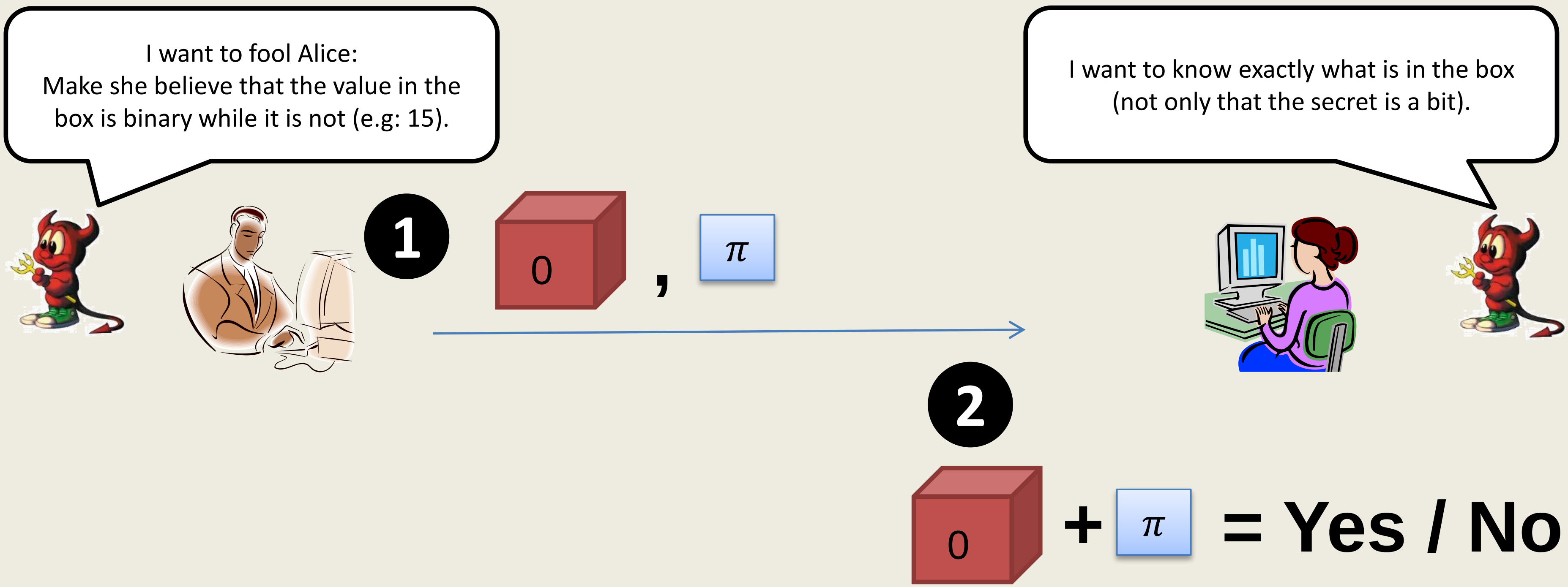
- NIZK argument to prove a commitment to a **bit vector** is the **binary expansion of the discrete logarithm** θ of $D = g^{\theta}$.

Commitments



Non-Interactive
Zero-Knowledge Proofs

Prove something about the secret in the box
without opening the box.



Abstract Protocol

Phase	$\mathcal{P}_A(\mathsf{CRS}, m_A, m_B)$		$\mathcal{P}_B(\mathsf{CRS}, m_A, m_B)$
KeyGen	$(sk_A, pk_A) \leftarrow \mathsf{FEKeyGen}(1^\kappa)$		
	pk_A	$\longrightarrow$	
			$(sk_B, pk_B) \leftarrow \mathsf{FEKeyGen}(1^\kappa)$
		$\longleftarrow$	pk_B
Encrypt Signature	$(\theta_A, \vec{r}_A, \gamma_A) \leftarrow \mathsf{EncSigGen}(\mathsf{CRS}, sk_A, m_A)$		
	γ_A	$\longrightarrow$	
			$(\theta_B, \vec{r}_B, \gamma_B) \leftarrow \mathsf{EncSigGen}(\mathsf{CRS}, sk_B, m_B)$
		$\longleftarrow$	γ_B
Verify Encrypted Signature	$v \leftarrow \mathsf{EncSigCheck}(\mathsf{CRS}, pk_B, m_B, \gamma_B)$		
	if $v = 0$ **then ABORT**		
			$v \leftarrow \mathsf{EncSigCheck}(\mathsf{CRS}, pk_A, m_A, \gamma_A)$
			if $v = 0$ **then ABORT**
Release Bits	**for** $i = 1$ **to** κ:		
	$\mathsf{open}_{A,i} \leftarrow \mathsf{KeyBitProofGen}(\mathsf{CRS}, \vec{r}_A, \theta_A, i)$		
	$\mathsf{open}_{A,i}$	$\longrightarrow$	
			$\mathsf{open}_{B,i} \leftarrow \mathsf{KeyBitProofGen}(\mathsf{CRS}, \vec{r}_B, \theta_B, i)$
		$\longleftarrow$	$\mathsf{open}_{B,i}$
	$v_i \leftarrow \mathsf{KeyBitCheck}(\mathsf{CRS}, \mathsf{open}_{B,i}, i)$		
	if $v_i = 0$ **then ABORT**		
			$v_i \leftarrow \mathsf{KeyBitCheck}(\mathsf{CRS}, \mathsf{open}_{A,i}, i)$
			if $v_i = 0$ **then ABORT**
	end for		
Recover Signature	$\sigma_{m_B} \leftarrow \mathsf{EncSigDecrypt}(\gamma_B, \theta_B)$		
			$\sigma_{m_A} \leftarrow \mathsf{EncSigDecrypt}(\gamma_A, \theta_A)$

Setup

Partial Fairness

$O_{Sign}(sk_B, \cdot)$

Not queried to

m_A, m_B, pk_A

(sk_B, pk_B)

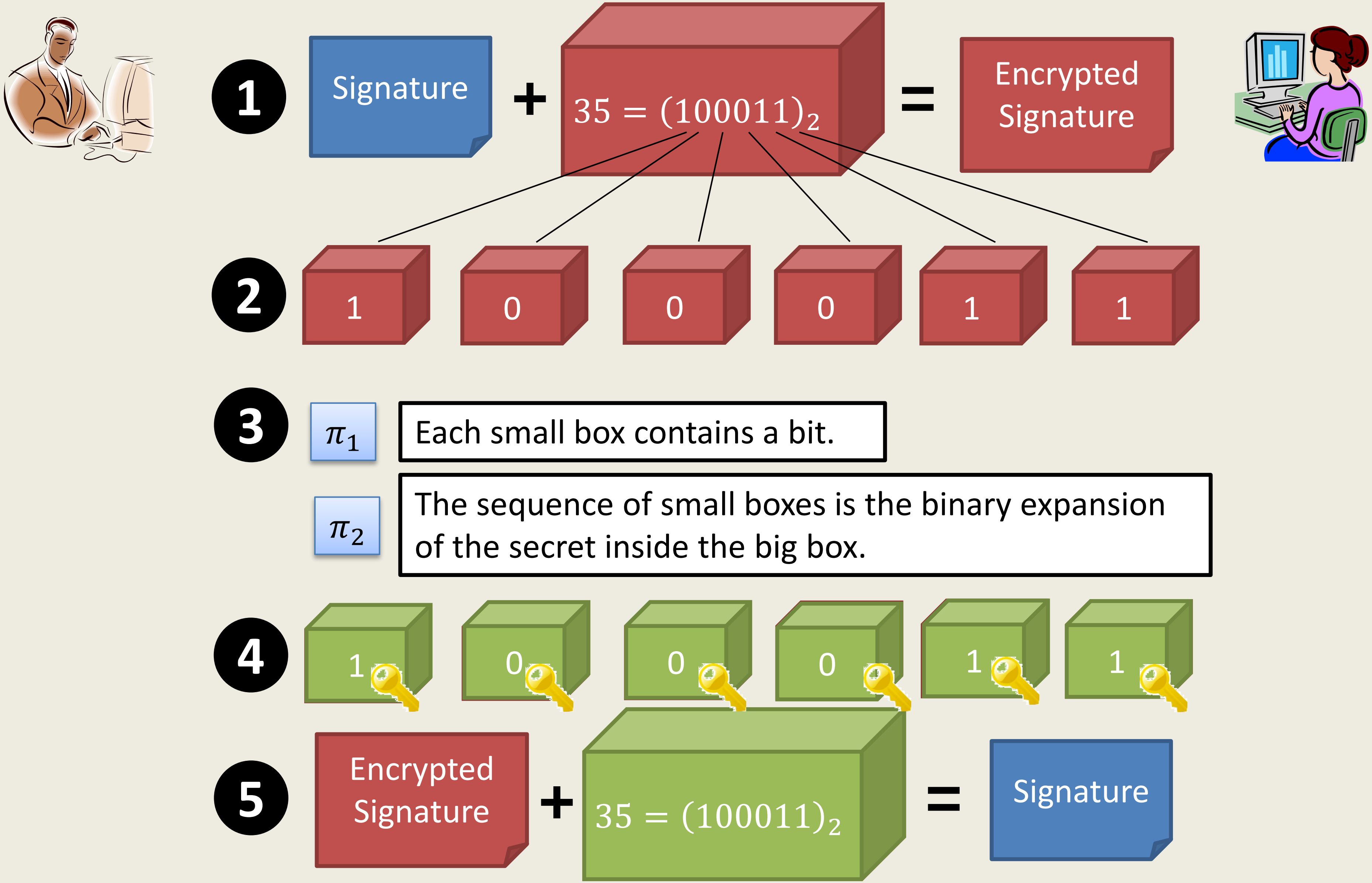
$$\frac{\Pr[\,\mathsf{SVf}(pk_B, m_B, \sigma_A) = \mathsf{valid}\,]}{\Pr[\,\mathsf{SVf}(pk_A, m_A, \sigma_B) = \mathsf{valid}\,]} \leq Q(\kappa)$$

σ_B on m_A

Bet according to
partially released secret

σ_A on m_B

Protocol


1
Signature
+
$35 = (100011)_2$
=
Encrypted Signature
2
1
0
0
0
1
1
3
π_1
Each small box contains a bit.
π_2
The sequence of small boxes is the binary expansion of the secret inside the big box.
4
1
0
0
0
1
1
5
Encrypted Signature
+
$35 = (100011)_2$
=
Signature

Bilinear maps

- $(p, e, G, G_T, g) \leftarrow BMGen(1^k)$

- $|G| = |G_T| = p$
- $e: G \times G \rightarrow G_T$
- $e(g^a, g^b) = e(g,g)^{ab}$
- $e(g,g)$ generates G_T

Assumptions

- Given $(g, g^s, g^{s^2}, g^{s^3}, \cdots, g^{s^q})$ it's hard to compute
  - $g^{\frac{1}{s}}$ (q- Diffie-Hellman Inversion)
  - $e(g, g,)^{\frac{1}{s}}$ (q-Bilinear Diffie-Hellman Inversion)
  - $(c, g^{\frac{1}{s+c}})$ (q-Strong Diffie-Hellman)
  - $g^{s^{q+i}}$ for $1 \leq i \leq q$
    ($q + i$ Diffie-Hellman Exponent)

Assumptions

- **Proposition:** $q - BDHI \Rightarrow \ q + i \ - DHE$
- Our protocol is secure under
  - $q - SDH$
  - $q - BDHI$

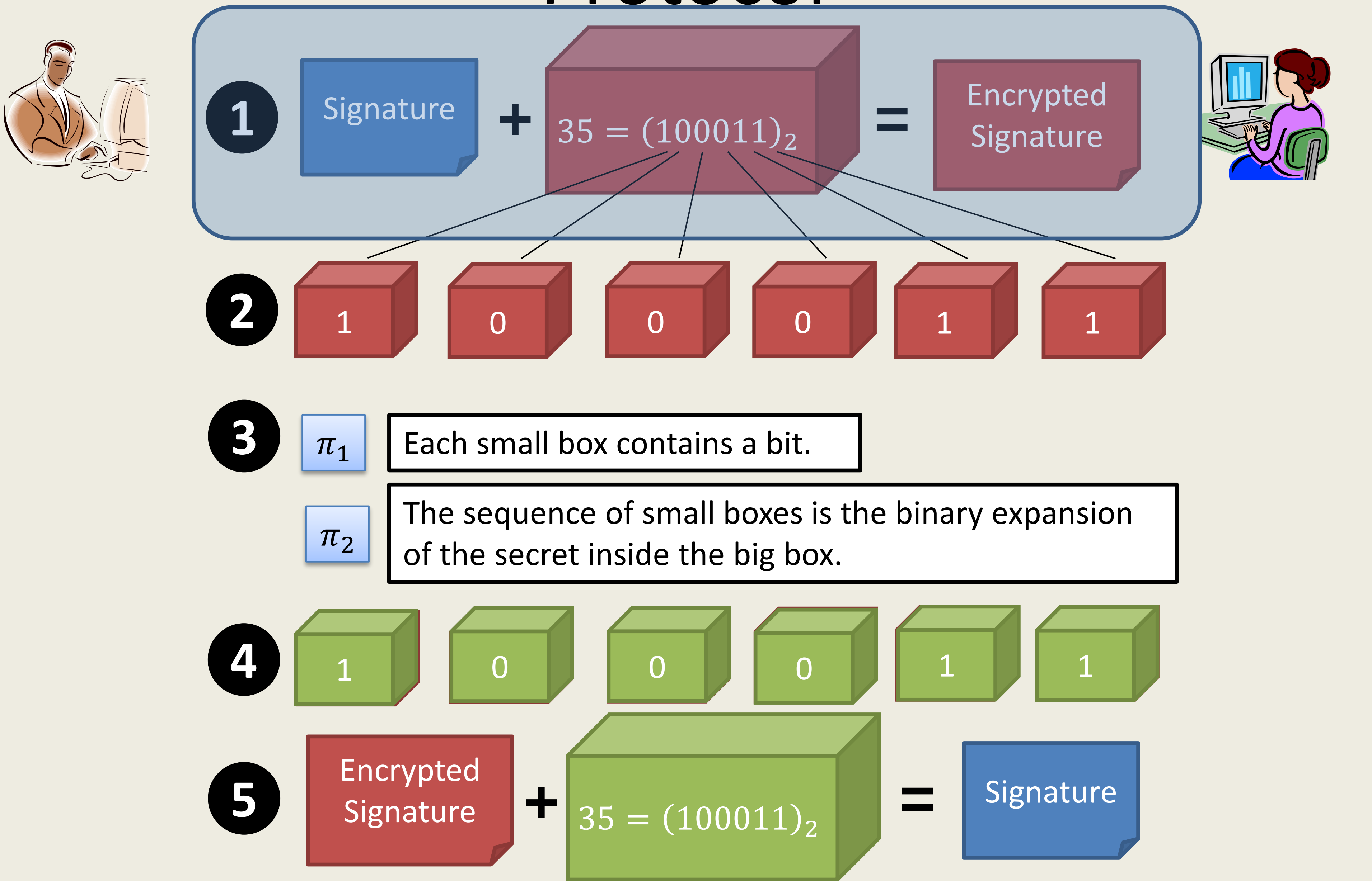
Short Signatures w/o Random Oracle [BB04]

- $\boldsymbol{KeyGen(1^k)}$
  1. $x, y \in Z_p$
  2. $u = g^x, v = g^y$
  3. $pk = (u, v), sk = (x, y)$
  4. return (sk, pk)

- $\boldsymbol{SSign(sk, m)}$
  1. $r \in Z_p$
  2. return $\sigma = (g^{\frac{1}{x+m+yr}}, r) = (\sigma_r, r)$

- $\boldsymbol{SVf(pk, m, \sigma)}$
  1. Check that $\boldsymbol{e(\sigma_r, ug^m v^r)} = e(g^{\frac{1}{x+m+yr}}, g^{x+m+yr}) = \boldsymbol{e(g, g)}$

Protocol


1
Signature
+
35 = (100011)2
=
Encrypted Signature
2
1
0
0
0
1
1
3
π1
Each small box contains a bit.
π2
The sequence of small boxes is the binary expansion of the secret inside the big box.
4
1
0
0
0
1
1
5
Encrypted Signature
+
35 = (100011)2
=
Signature

The Encrypted Signature

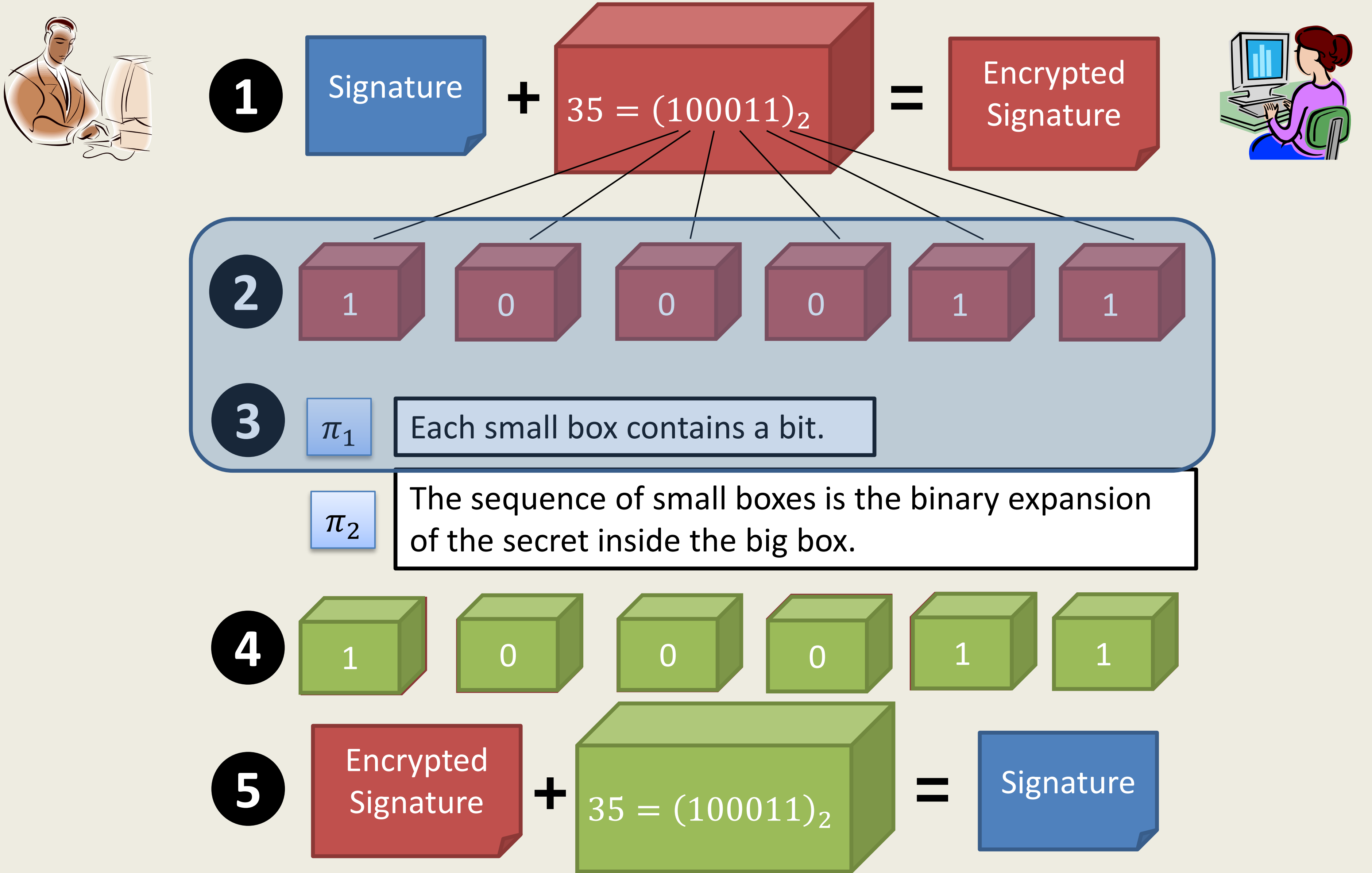
- Computing
  - $\theta \leftarrow Z_p$
  - $D = g^{\theta}$
  - $\boldsymbol{\sigma} = (\boldsymbol{g}^{\frac{\theta}{x+m+yr}}, \boldsymbol{r})$

Secret key / “blinding” factor

Boneh-Boyen signature “blinded” by θ

- Checking
  - Given (D, σ, pk, m) parse σ and pk as
    - $\sigma = (\sigma_\theta, r)$
    - $pk = (g, u = g^x, v = g^y)$
  - $\boldsymbol{e}(\boldsymbol{\sigma_\theta}, \boldsymbol{u}\boldsymbol{g}^{\boldsymbol{m}}\boldsymbol{v}^{\boldsymbol{r}}) = e(g^{\frac{\theta}{x+m+yr}}, g^{x+m+yr}) = \boldsymbol{e}(\boldsymbol{D}, \boldsymbol{g})$

Protocol



1
Signature
+
35 = (100011)₂
=
Encrypted Signature
2
1
0
0
0
1
1
3
π1
Each small box contains a bit.
π2
The sequence of small boxes is the binary expansion of the secret inside the big box.
4
1
0
0
0
1
1
5
Encrypted Signature
+
35 = (100011)₂
=
Signature

NIZK argument 1

- $CRS = (g, g^{s}, g^{s^2}, g^{s^3}, \cdots, g^{s^q}) = (g_0, g_1, g_2, g_3, \ldots, g_q\ )$
- **Statement**

  Let $C\ = (C_1, C_2, \ldots, C_q)$

  The prover knows $(r_i, b_i) \in\ (Z_p \times\ \{0,1\})$ such that $\boldsymbol{C_i = g^{r_i} g_i^{b_i}}$
- **Argument**
  - $A_i\ =\ g_{q-i}^{r_i}\, g_q^{b_i}$
  - B_i such that $e(A_i, C_i g_i^{-1}) = e(B_i, g)$
  - Return (A_i, B_i) for each $i \in [1..q]$
- **Verification**
  - $e(A_i, g) = e(C_i, g_{q-i})$
  - $e(A_i, C_i g_i^{-1}) = e(B_i, g)$

Shift C_i by $q - i$ positions to the right.

Force the product $\boldsymbol{b_i(b_i - 1)}$ to be computed in the exponent.

NIZK argument 1

- **Theorem:**

  The argument is perfectly complete, computationally sound under the $q+i$ - DHE assumption and perfectly zero-knowledge.

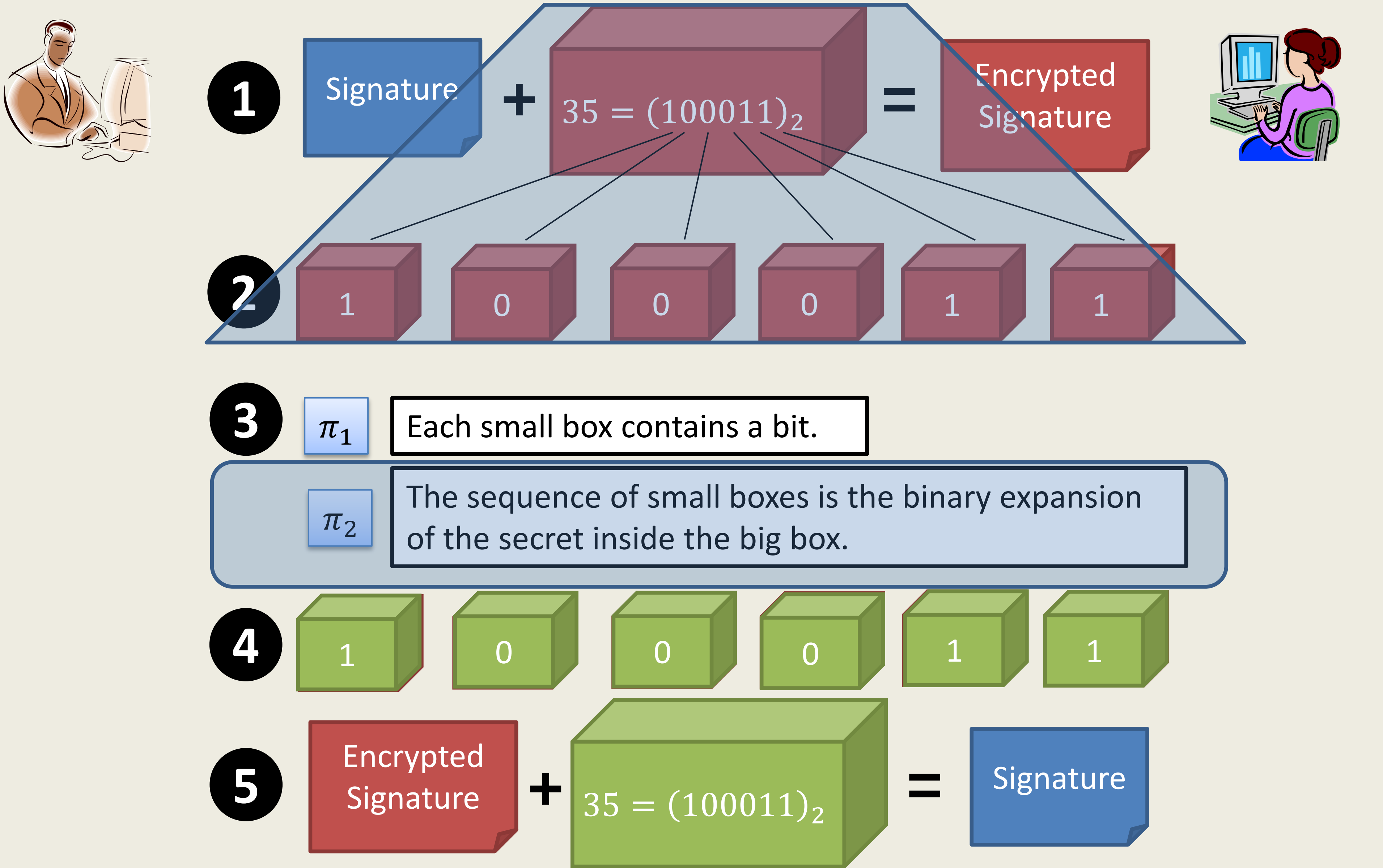
Proof (sketch).

$$e(A_i, C_i g_i^{-1}) = e(g_{q-i}^{r_i} g_q^{b_i}, g^{r_i} g_i^{b_i - 1})$$

$$= e\Big(\underbrace{g_{q-i}^{r_i^2} g_q^{r_i(2b_i-1)}}_{B_i} \boldsymbol{g_{q+i}^{b_i(b_i-1)}}, g\Big) = e(B_i, g)$$

If $b_i \notin \{0,1\}$, the adversary breaks the $q+i-\mathrm{DHE}$ assumption.

Protocol


1
Signature
+
35 = (100011)2
=
Encrypted Signature
2
1
0
0
0
1
1
3
π1
Each small box contains a bit.
π2
The sequence of small boxes is the binary expansion of the secret inside the big box.
4
1
0
0
0
1
1
5
Encrypted Signature
+
35 = (100011)2
=
Signature

NIZK argument 2

- $CRS = \left(g, g^{s}, g^{s^2}, g^{s^3}, \cdots, g^{s^q}\right) = \left(g_0, g_1, g_2, g_3, \dots, g_q\right)$

- We set $q = \kappa$ (security parameter)

- **Statement**
  - The prover knows $(r_i, b_i) \in (Z_p \times \{0,1\})$ and θ such that $C_i = g^{r_i} g_i^{b_i}$, $D = g^{\theta}$ and

$$\theta = \sum_{i=1}^{\kappa} b_i 2^{i-1}$$

NIZK argument 2

- **Verification:** Input (C …
  - Parse $\pi = (r', U, V)$
  - Check that $e(\frac{\prod_{i=1}^{k} C_i}{g^{r'}}, g) = e(U, g_1)$
  - Check that $e(\frac{U}{D}, g) = e(V, g_1 g^{-2})$

$$\prod_{i=1}^{k} C_i = \prod_{i=1}^{k} g^{r_i} g_i^{b_i} \Leftrightarrow [r', b_1, b_2, \ldots, b_\kappa]$$

$$U = (\prod_{i=1}^{k} g_i^{b_i})^{1/s} = \prod_{i=1}^{k} g_{i-1}^{b_i} \Leftrightarrow [b_1, b_2, \ldots, b_\kappa]$$

$$r' = \sum_i r_i$$

θ

$U \Leftrightarrow P(s)$ (i.e. $U = g^{P(s)}$)

$V \Leftrightarrow W(s)$ s.t. $P(s) - P(2) = W(s)(s-2)$

NIZK argument 2

- **Theorem:**

  The argument is perfectly complete, computationally sound under the $q - SDH$ assumption and perfectly zero-knowledge.

Protocol

1 Signature + $35 = (100011)_2$ = Encrypted Signature

2 1 0 0 0 1 1

3 π_1 Each small box contains a bit.

π_2 The sequence of small boxes is the binary expansion of the secret inside the big box.

4 1 0 0 0 1 1

5 Encrypted Signature + $35 = (100011)_2$ = Signature

Recovering the Signature

- All the bits b_i are revealed
- Compute $\theta = \sum_{i=1}^{\kappa} b_i \, 2^{i-1}$
- We have $\sigma = \left(g^{\frac{\cancel{\theta}}{x+m+yr}}, r\right) = (\sigma_\theta, r)$
- Compute $\boldsymbol{\sigma} = (\sigma_\theta{}^{\mathbf{1}/\boldsymbol{\theta}}, \boldsymbol{r})$

Proofs of Knowledge

- Discrete logarithm θ of
  - $D = g^{\theta}$
- r_i, b_i such that
  - $C_i = g^{r_i} g_i^{b_i}$

Needed in order to simulate the adversary despite it aborts early.

Simultaneous Hardness of Bits for Discrete Logarithm

Holds in the generic group model **[Schnorr98]**

An adversary cannot distinguish between a **random sequence** of $\boldsymbol{\kappa - l}$ bits and the **first** $\boldsymbol{\kappa - l}$ bits **of** $\boldsymbol{\theta}$ given $\boldsymbol{g^{\theta}}$.

$$Adv^{SHDL}(\mathcal{A},\kappa) = \left|\Pr\left[\begin{array}{c}\theta \xleftarrow{R} \mathbb{Z}_p : \\ 1 \leftarrow \mathcal{A}(g^{\theta}, \theta[1..\kappa - l])\end{array}\right] - \Pr\left[\begin{array}{c}\theta, \alpha \xleftarrow{R} \mathbb{Z}_p : \\ 1 \leftarrow \mathcal{A}(g^{\theta}, \alpha[1..\kappa - l])\end{array}\right]\right|$$

$l = \omega(\log \kappa)$

Conclusion

- Fair exchange protocol for short signatures [BB04] without TTP
- Practical
- Two new NIZK arguments

Thank you!

Partial Fairness

- A randomized protocol for signing contracts [EGL85]
- Gradual release of a secret [BCDB87]
- Practically and Provably secure release of a secret and exchange of signatures [Damgard95]
- Resource Fairness and Composability of Cryptographic protocols [GMPY06]

Only contract signing

RSA, Rabin, ElGamal signatures

"Time-line" assumptions, Generic construction

- **Theorem:**

  The protocol is partially fair under the $\kappa - SDH$ and the $\kappa - BDHI$ assumption.

Proof (Sketch)

- Type I
  - Does not forge values but aborts «early»
  - => He has to break the signature scheme

  - **Careful:**
    What happens if A detects he is simulated?
    - The simulator will try to break the SHDL assumption
    - If few bits remain, it does not win, everything is OK!

Proof (Sketch)

- Type II
  - Forge values
  - The simulator can extract all values computed by adversary and break the soundness of the NIZK arguments or binding property of commitment scheme.