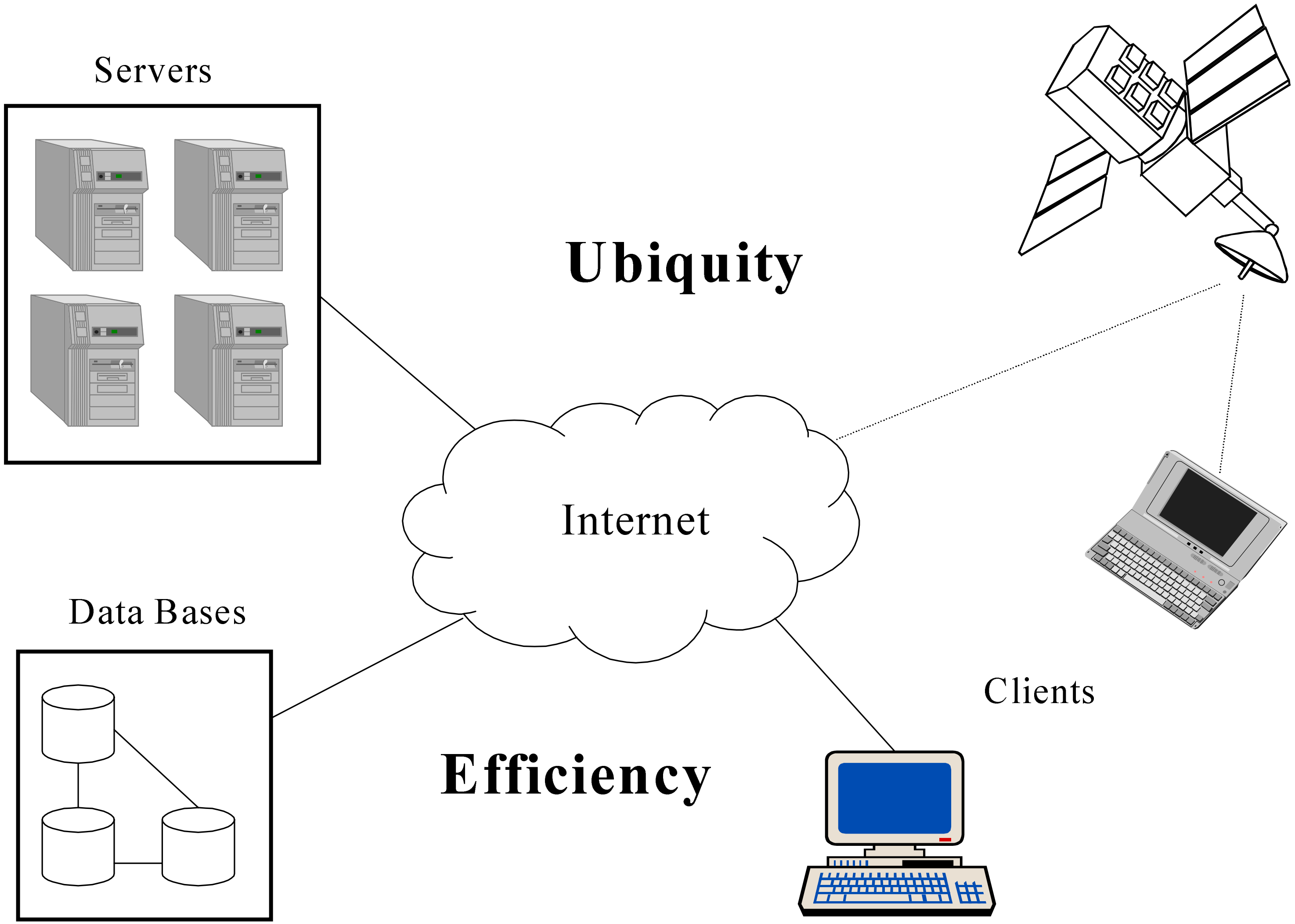
User Efficient Blind Signatures and Its Applications in Digital Cash and Electronic Voting

雷欽隆
Chin-Laung Lei

Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.

Outlines

Introduction


Servers
Ubiquity
Internet
Data Bases
Clients
Efficiency

Features of Internet Services:

- **Efficiency**: Faster than traditional services
- **Ubiquity**: Users can obtain services anywhere.
- **Flexibility**: Clients can request services anytime.
- **Openness**: Popularization
- Examples: Digital cash and electronic voting services

Some Challenges to Internet Services:

- Robust **security** mechanisms and protocols
  - Hackers and viruses
  - Privacy and policy considerations
- **Efficiency**
  - A lot of extra computations must be performed by users.
  - Limited power of devices such as mobile units or smart cards

Goals:

1. Design efficient blind signature schemes to reduce the computation overheads of users especially for digital cash and electronic voting.

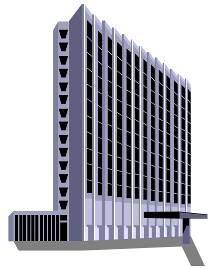
2. Develop flexible digital cash services for different types of transactions

3. Construct practical voting services to strengthen the security of electronic elections.

Preliminary

Digital Signatures

Signer



❶ message

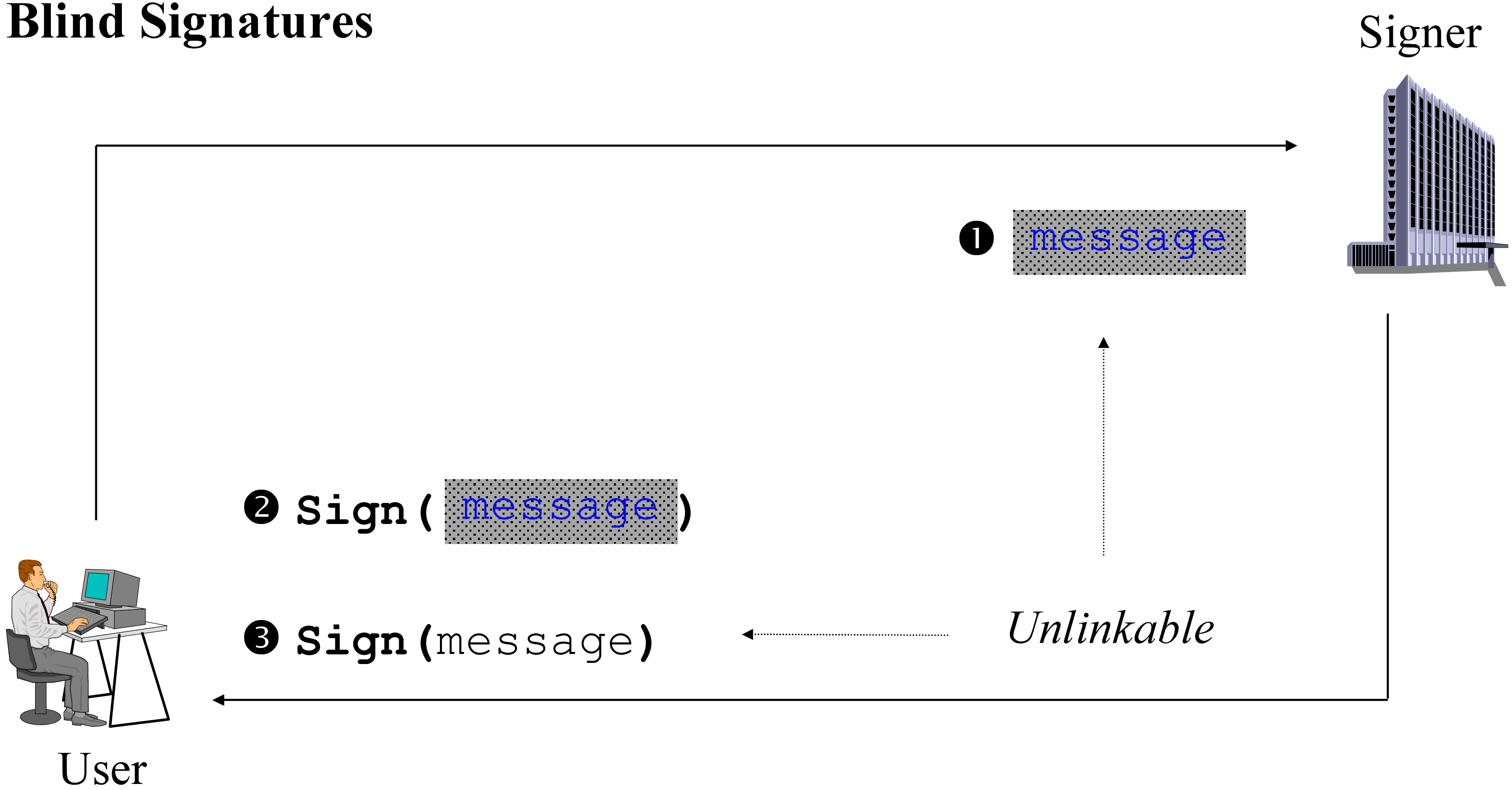
Linkable

❷ **Sign**(message)

User

★**Authentication**

Blind Signatures

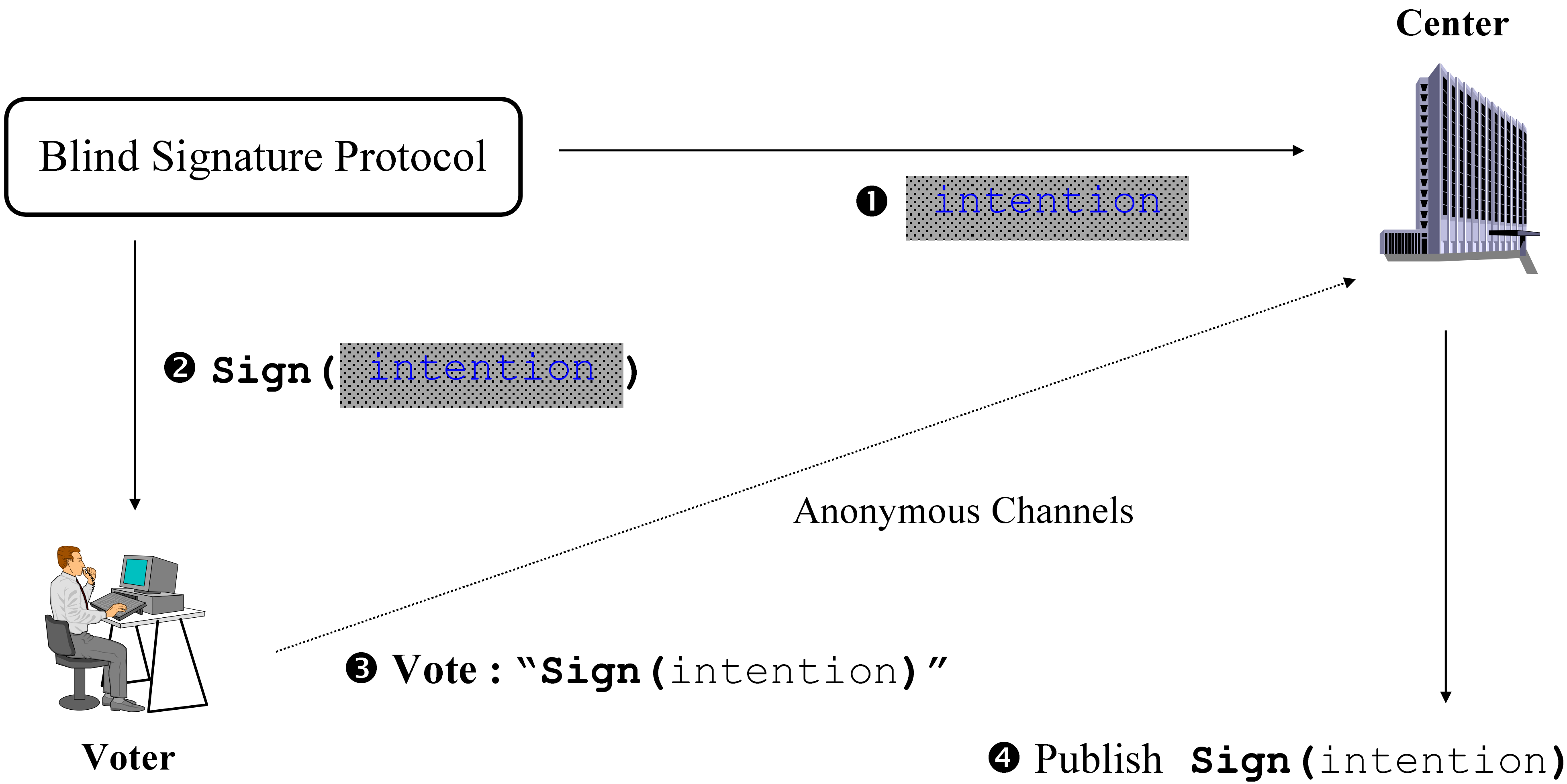


★**Privacy**

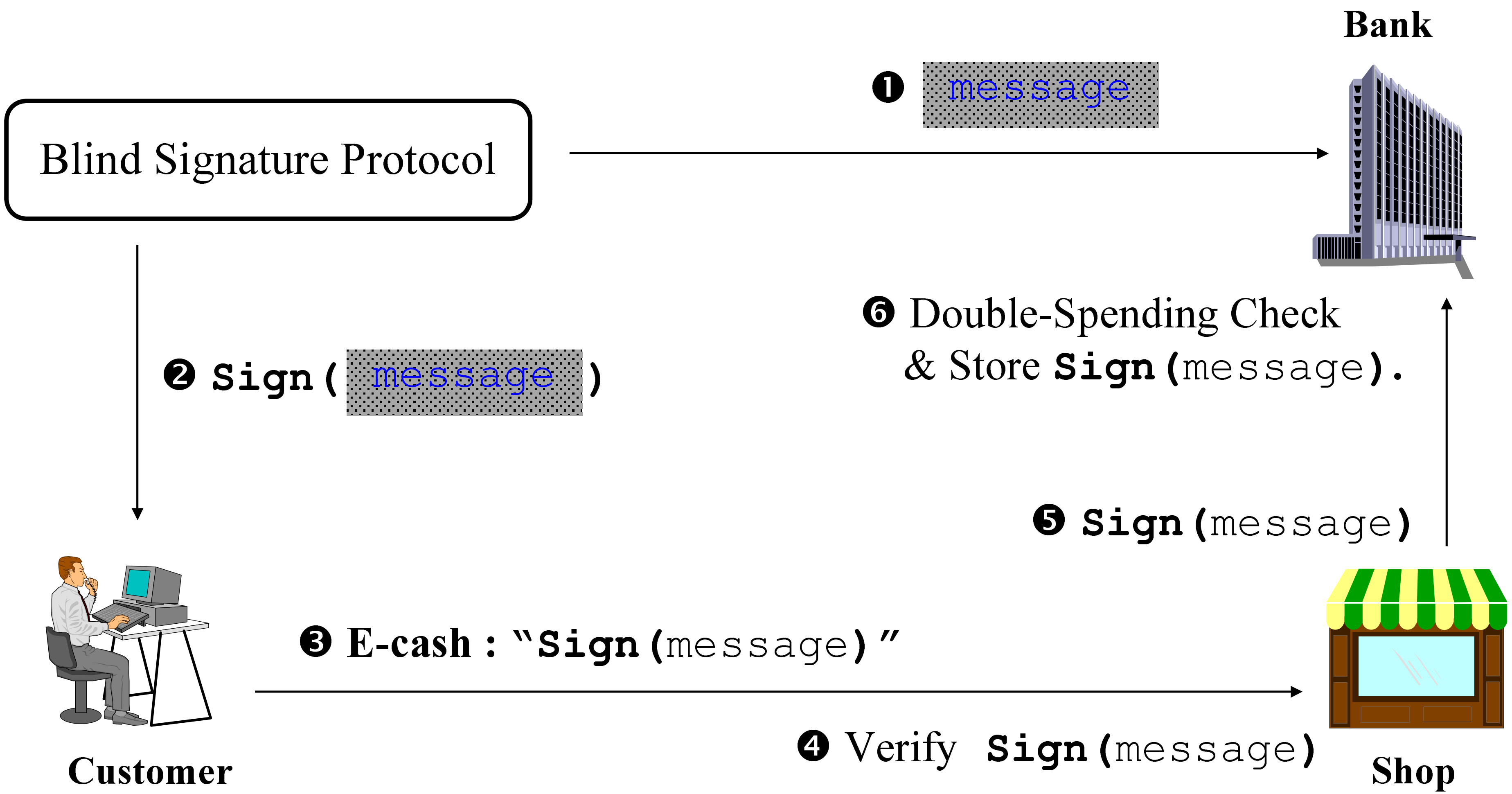
Blind Signatures

- Blind Signatures $\rightarrow$ Unforgeability + Unlinkability
- Anonymous Electronic Voting
- Untraceable Digital Cash

Electronic Voting

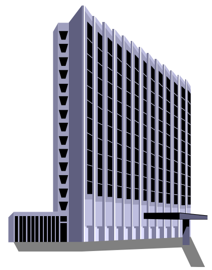


Digital Cash



Partially Blind Signatures

Signer



❶ message

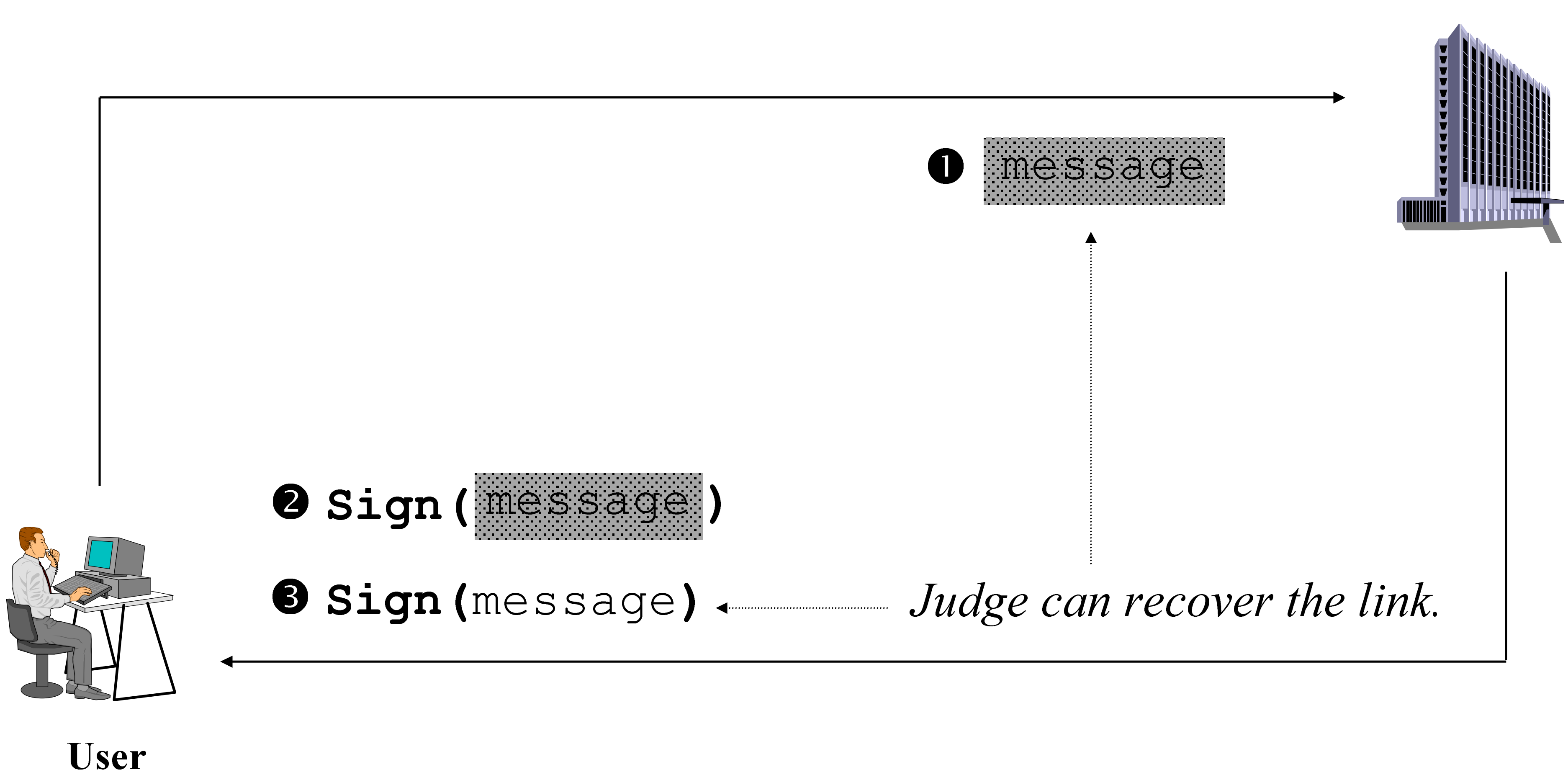
❷ **Sign(**message **;** ***A*****)**

❸ **Sign(**message**;** ***A*****)** ← *Unlinkable under the same* ***A***

User

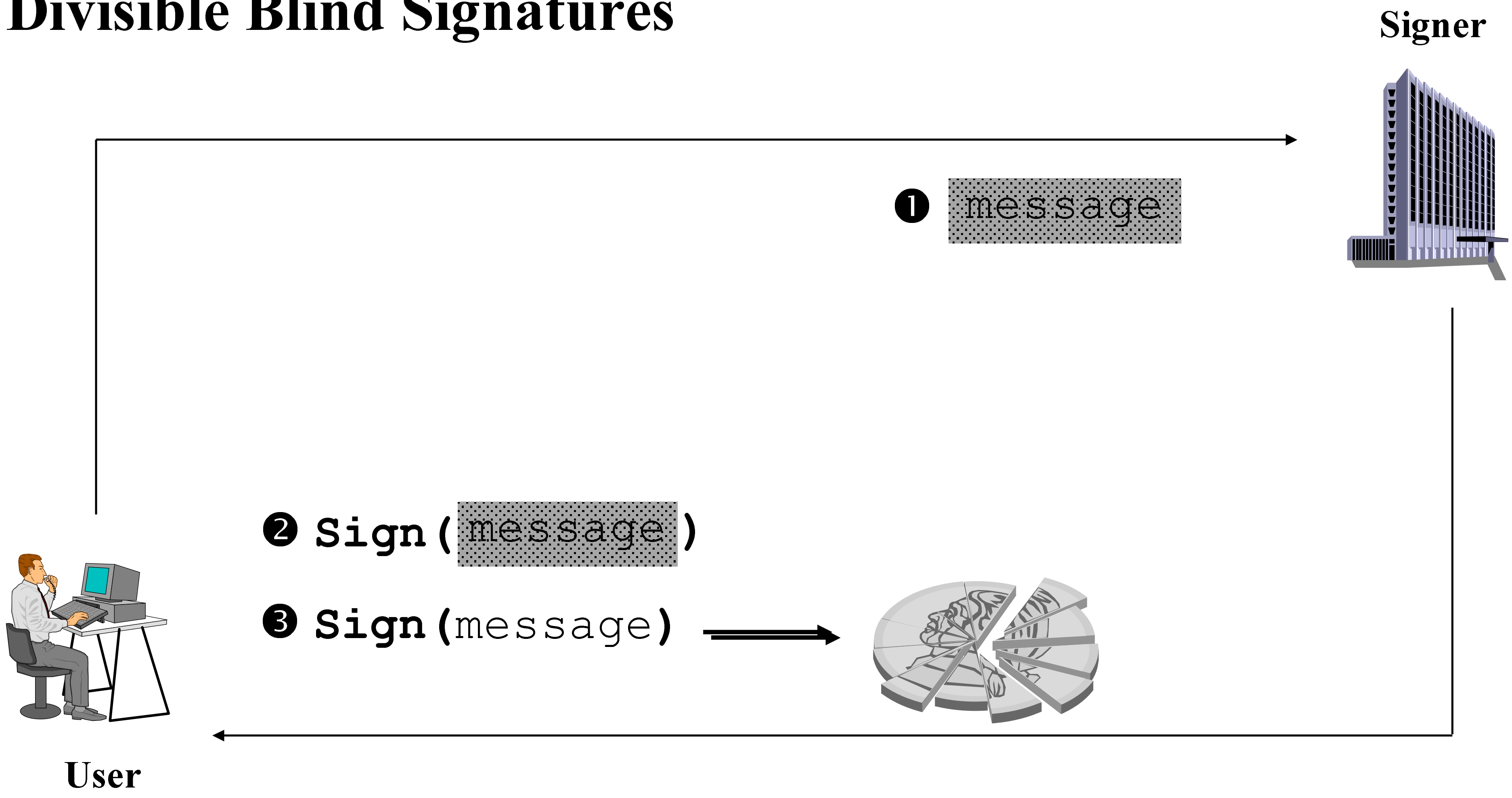
★**Prevent the bank's database from growing unlimitedly .**

Fair Blind Signatures



★**Cope with the misuse problem of unlinkability.**

Divisible Blind Signatures



★**Reduce the storage of digital cash.**

User Efficient Blind Signatures

■ A Typical Blind Signature Scheme *X*

M is the underlying set of messages.

R is a finite set of random integers.

$S_X : M \rightarrow M$ is the signing function kept secret by the signer.

$V_X : S_X(M) \times M \rightarrow \{\text{true, false}\}$ is the verification formula.

$B_X : M \times R \rightarrow M$ is the blinding function.

$U_X : S_X(M) \times R \rightarrow S_X(M)$ is the unblinding function, and

$\forall\, m \in M$ and $r \in R$, $U_X(S_X(B_X(m, r)), r) = S_X(m)$.

User | **Signer**

$m \in M$

$r \in R$

Blinding: $B_X(m, r)$ →

Blind Signing: $t = S_X(B_X(m, r))$

← t

Unblinding: $U_X(t, r) = S_X(m)$

Signature: $(S_X(m), m)$

Verifying: $V_X(S_X(m), m) \stackrel{?}{=} \text{True}$

■ The Proposed Blind Signature Scheme

- The first blind signature scheme based on Quadratic Residues.
- If $x^2 = y \pmod{n}$, then y is a quadratic residue (QR) in Z_n^* and x is a square root of y.
- If $n = p_1 p_2$ and p_1, p_2 are distinct large primes, then, given y and n, it is intractable to compute x without p_1 or p_2.

User		**Signer**
$m \in Z_n$ $u, v \in_R Z_n$		
	$\alpha = H(m)(u^2+v^2) \bmod n$ →	
		$x \in_R Z_n$
	← x	
$b \in_R Z_n$ $\delta = b^2 \bmod n$		
	$\beta = \delta(u-vx) \bmod n$ →	
		$\lambda = \beta^{-1} \bmod n$ $t^4 \equiv_n \alpha(x^2+1)\lambda^2$
	← (t, λ)	
$c = \delta\lambda(ux+v) \bmod n$ $s = bt \bmod n$ Verify (c, m, s) : $s^4 \stackrel{?}{\equiv}_n H(m)(c^2+1)$		

Discussions:

- Since b, u, and v are randomly chosen by the user, the signer cannot link the signature-message triple (c, m, s) to the instance of the signature protocol producing that triple. **(Unlinkability)**

- As p, q are kept secret by the signer and ***H*** **is one-way**, it is computationally infeasible for an intruder to forge a valid signature. **(Unforgeability)**

- The user only requires to perform 10 multiplications to obtain a valid signature-message triple, and only 4 multiplications is needed to verify a signature-message triple. **(Efficiency)**

Property Comparisons:

	Our Scheme [30, 33]	Camenisch [10]	Chaum [12]	Ferguson [39]	Pointcheval [62]	Pointcheval [63]
Foundation	**QR**	**DL/DL**	**RSA**	**RSA**	**DL/RSA**	**QR/QR**
Unlinkability	○	○/○	○	○	○/○	○/○
Randomization	○	○/○	×	○	○/○	○/○
Message Recovery	○	× /○	○	×	× /×	× /×

Comparisons of Computation Overheads for Users:

	Our Scheme [30, **33**]	Camenisch [10]	Chaum [12]	Ferguson [39]	Pointcheval [62]	Pointcheval [63]
Exponentiations	**0**	**4**	**2**	**4**	**6**	**3**
Inverses	**0**	**2**	**1**	**1**	**0**	**0**
Hashings	**2**	**0**	**2**	**2**	**2**	**2**
Multiplications	**14**	**6**	**2**	**3**	**5**	**2*k***
Reduced by:		**> 99%**	**> 99%**	**> 99%**	**> 99%**	**> 99%**

■ The Proposed Partially Blind Signature Scheme

- The first partially blind signature scheme based on QR.
- The signer ensures that every signature issued by him contains the information ***a*** he desires, such as the expiration date of an e-cash or the identity of an election.
- The property of partial blindness makes it possible for the bank to minimize its database which keeps the spent e-cash.

User		**Signer**
$m, a \in Z_n$; $u, v \in_R Z_n$ $\alpha = H(m)(u^2+v^2) \bmod n$		
	(a, α) →	
		Verify a. $x \in_R Z_n$
	← x	
$b \in_R Z_n$ $\delta = b^4 \bmod n$		
	$\beta = \delta(u-vx) \bmod n$ →	
		$\lambda = \beta^{-1} \bmod n$ $t^8 \equiv_n H(a)(\alpha(x^2+1))^3\lambda^6$
	← (t, λ)	
$c = \delta\lambda(ux+v) \bmod n$ $s = b^3 t \bmod n$ Verify (a, c, m, s) : $s^8 \overset{?}{\equiv}_n H(a)(H(m)(c^2+1))^3$		

Discussions:

- The signer cannot link the signature-message 4-tuple (a, c, m, s) to the instance of the signature protocol producing that 4-tuple under the same a. **(Unlinkability under the same embedded information)**

- Computing $(H(a)^{3^{-1} \bmod \phi(n)} \bmod n)$ is infeasible without p or q where $\phi(n) = (p-1)(q-1)$. Furthermore, we can select p and q with $3|(p-1)$ or $3|(q-1)$ such that $(3^{-1} \bmod \phi(n))$ does not exist.

- The user only performs 12 multiplications to obtain a valid signature and 8 multiplications to verify a signature, respectively.

Property Comparisons:

	Ours [30, 33]	Abe [1]	Camenisch [10]	Chaum [12]	Ferguson [39]	Pointcheval [62]	Pointcheval [63]
Foundation	**QR**	**RSA**	**DL/DL**	**RSA**	**RSA**	**DL/RSA**	**QR/QR**
Unlinkability	○	○	○/○	○	○	○/○	○/○
Randomization	○	×	○/○	×	○	○/○	○/○
Message Recovery	○	○	× /○	○	×	× /×	× /×
Partial Blindness	○	○	× /×	×	×	× /×	× /×

Comparisons of Computation Overheads for Users:

	Ours [30, **33**]	**Abe** [1]	**Camenisch** [10]	**Chaum** [12]	**Ferguson** [39]	**Pointcheval** [62]	**Pointcheval** [63]
Exponentiations	**0**	**2**	**4**	**2**	**4**	**6**	**3**
Inverses	**0**	**1**	**2**	**1**	**1**	**0**	**0**
Hashings	**3**	**4**	**0**	**2**	**2**	**2**	**2**
Multiplications	**20**	**4**	**6**	**2**	**3**	**5**	**2*k***
Reduced by:		**> 99%**	**> 99%**	**> 99%**	**> 99%**	**> 99%**	**> 99%**

■ The Proposed Fair Blind Signature Scheme

- The unlinkability property of blind signatures may be misused by criminals, such as to launder money or to safely get a ransom.
- In a fair blind signature scheme, the judge can make signatures linkable when necessary.
- The proposed scheme is the first fair blind signature scheme based on QR, and comparing with the existing schemes, our method largely reduces the computation overheads of users.

User		**Judge**		**Signer**
		β, γ, b, z : random		
		$u = H(\beta)$		
	$\Longleftarrow (b, u, v, z, S(z))$	$v = H(\gamma)$		
m : message				
$\alpha \equiv_n H(m)(u^2+v^2)$		$(\alpha, z, S(z)) \longrightarrow$		Verify $S(z)$.
				δ : random
			$\longleftarrow (x, z, S(z))$	$x = H(\delta)$
		$\lambda \equiv_n b^2(u-vx)$	$\lambda \longrightarrow$	
				$e \equiv_n \lambda^{-1}$
		$\longleftarrow (e, t, x)$		$t^4 \equiv_n \alpha(x^2+1)e^2$
$c = b^2 e(ux+v) \bmod n$				
$s = bt \bmod n$				
Verify (c, m, s) : $s^4 \overset{?}{\equiv}_n H(m)(c^2+1)$				

Discussions:

- Given a triple (c, m, s), the judge can reveal (β, γ, b, z) to the signer where $c \equiv_n (H(\beta)x+H(\gamma))(H(\beta)-H(\gamma)x)^{-1}$, so that the signer can link (c, m, s) to the identifier z. (**Linkage Recovery**)

- If the judge does not reveal necessary information to the signer, the unlinkability of signatures is preserved.

- The user only performs 14 multiplications to obtain a valid signature and 4 multiplications to verify a signature-message triple, respectively.

Property Comparisons:

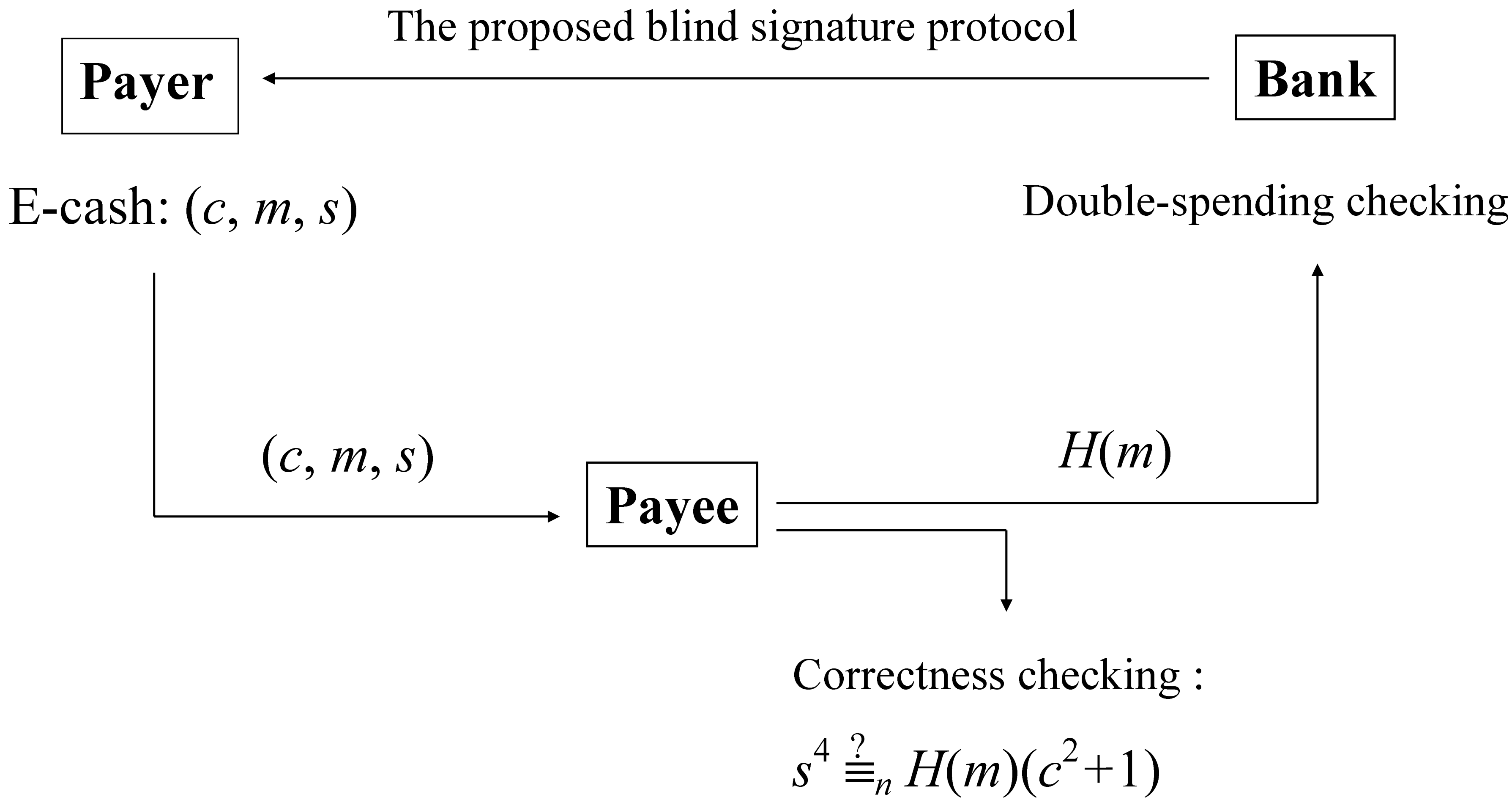
	Ours [30, 33]	Camenisch [10]	Chaum [12]	Ferguson [39]	Pointcheval [62]	Pointcheval [63]	Stadler [78]
Foundation	**QR**	**DL/DL**	**RSA**	**RSA**	**DL/RSA**	**QR/QR**	**RSA/DL/DL**
Unlinkability	○	○/○	○	○	○/○	○/○	○/○/○
Randomization	○	○/○	×	○	○/○	○/○	× /○/○
Message Recovery	○	× /○	○	×	× /×	× /×	× /× /×
Fairness	○	× /×	×	×	× /×	× /×	○/○/○

Comparisons of Computation Overheads for Users:

	Ours [30, **33**]	Camenisch [10]	Chaum [12]	Ferguson [39]	Pointcheval [62]	Pointcheval [63]	Stadler [78]
Exponentiations	**0**	**4**	**2**	**4**	**6**	**10**	**3**
Inverses	**0**	**2**	**1**	**1**	**0**	**1**	**0**
Hashings	**2**	**0**	**2**	**2**	**2**	**2**	**2**
Multiplications	**18**	**6**	**2**	**3**	**5**	**6**	**2k**
Reduced by:		**> 99%**	**> 99%**	**> 99%**	**> 99%**	**> 99%**	**> 99%**

Untraceable Digital Cash

■ User Efficient Untraceable Digital Cash



■ Divisible Digital Cash

Initialization :

H_1 , H_2 , H_3 , ..., H_w are one-way hash functions.

$$H_{i:j}(m) = \begin{cases} H_i(H_{i+1}(H_{i+2}(\ldots(H_j(m))))), & \text{if } i \le j \\ m & \text{otherwise.} \end{cases}$$

User		**Bank**
$m \in Z_n$ $u, v \in_R Z_n$		Publish w.
	$\boldsymbol{\alpha = H(m)_{1:w}(u^2+v^2)}$ →	
	← x	$x \in_R Z_n$
$b \in_R Z_n$ $\delta = b^2 \bmod n$		
	$\beta = \delta(u-vx) \bmod n$ →	
		$\lambda = \beta^{-1} \bmod n$ $t^4 \equiv_n \alpha(x^2+1)\lambda^2$
	← (t, λ)	
$c = \delta\lambda(ux+v) \bmod n$ $s = bt \bmod n$		

Verify (c, m, s, w) : $s^4 \overset{?}{\equiv}_n H_{1:w}(m)(c^2+1)$

Divide (c, m, s, w) into q sub-cash :

$$(c, m, s, w) \Longrightarrow \begin{array}{c} (c, m_1, s, w_1) \\ (c, m_2, s, w_2) \\ \vdots \\ (c, m_q, s, w_q) \end{array}$$

where $w_1 + w_2 + ... + w_q = w$

$$m_i = H_{(e_i + w_i + 1) : w}(m)$$

$$e_i = w_1 + w_2 + ... + w_{i-1}$$

Verification:

For each (c, m_i, s, w_i) :

$$s^4 \overset{?}{\equiv}_n H_{1:(e_i+w_i)}(m_i)(c^2+1)$$

where $e_1 = 0$

$e_i = w_1+w_2+...+w_{i-1}$

The Division Tree:

$w = e_q + w_q$

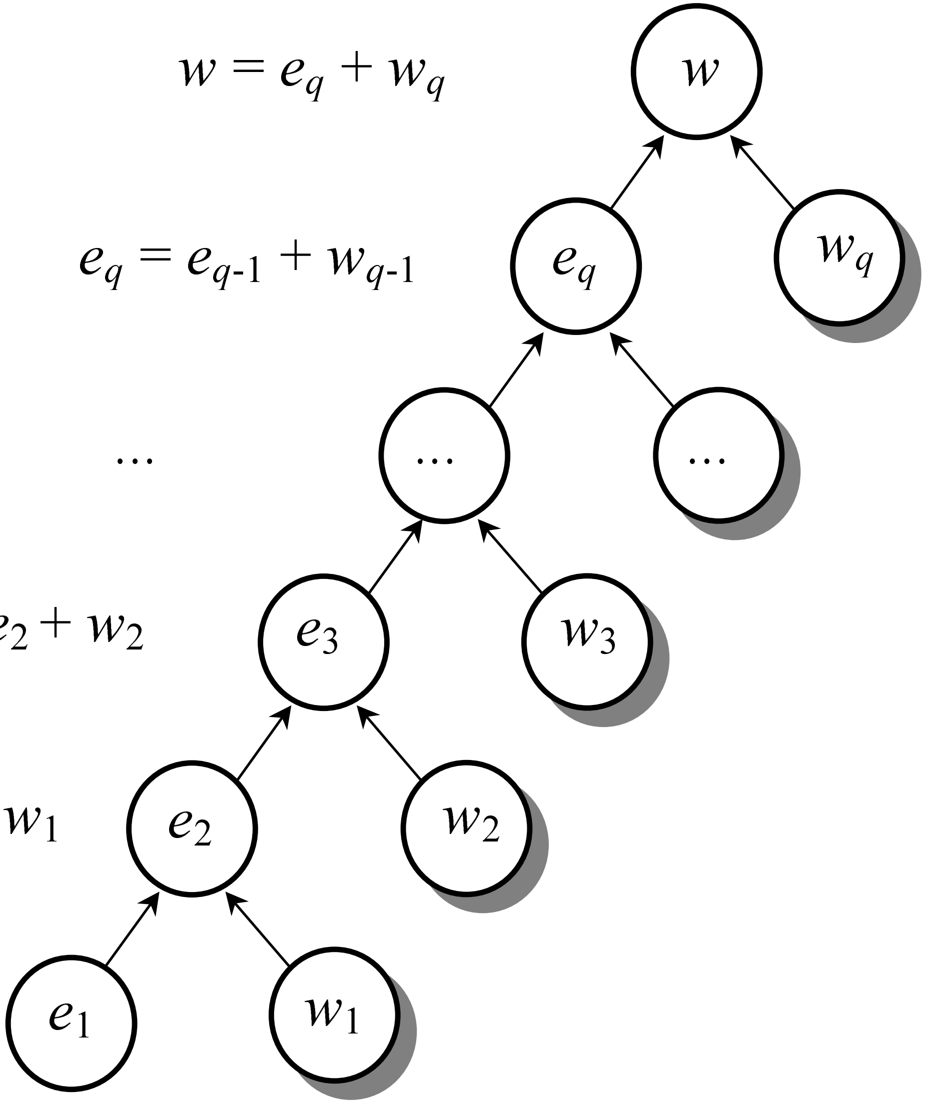
$e_q = e_{q\text{-}1} + w_{q\text{-}1}$ $\qquad$ $s^4 \overset{?}{\equiv}_n H_{1:(e_q+w_q)}(m_q)(c^2+1)$

… $\qquad$ …

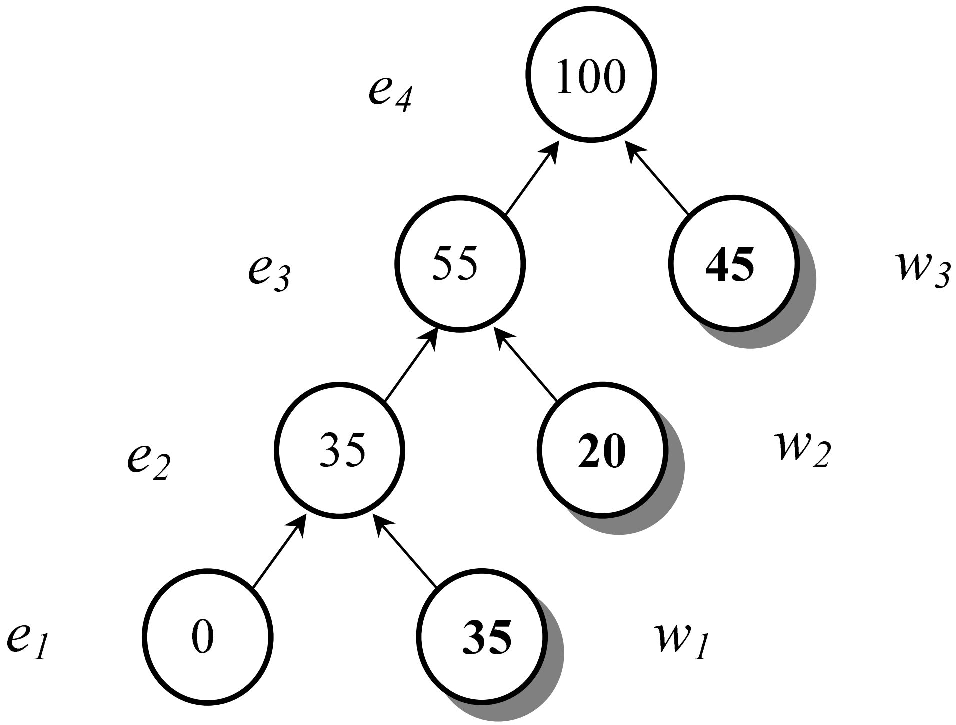
$e_3 = e_2 + w_2$ $\qquad$ $s^4 \overset{?}{\equiv}_n H_{1:(e_3+w_3)}(m_3)(c^2+1)$

$e_2 = e_1 + w_1$ $\qquad$ $s^4 \overset{?}{\equiv}_n H_{1:(e_2+w_2)}(m_2)(c^2+1)$

$e_1 = 0$ $\qquad$ $s^4 \overset{?}{\equiv}_n H_{1:(e_1+w_1)}(m_1)(c^2+1)$



An Example:

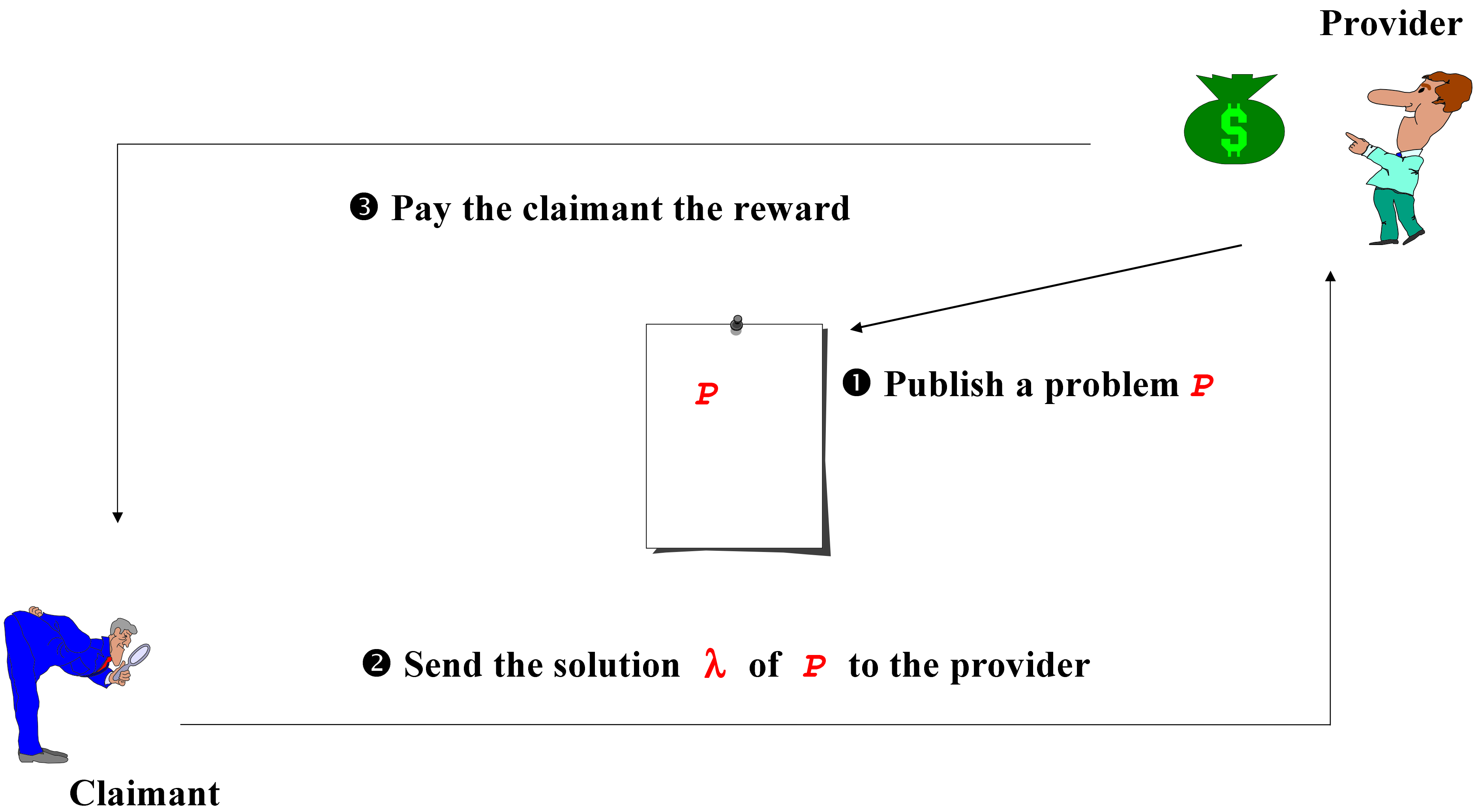


$$s^4 \overset{?}{\equiv}_n H_{1:100}(m_3)(c^2+1)$$

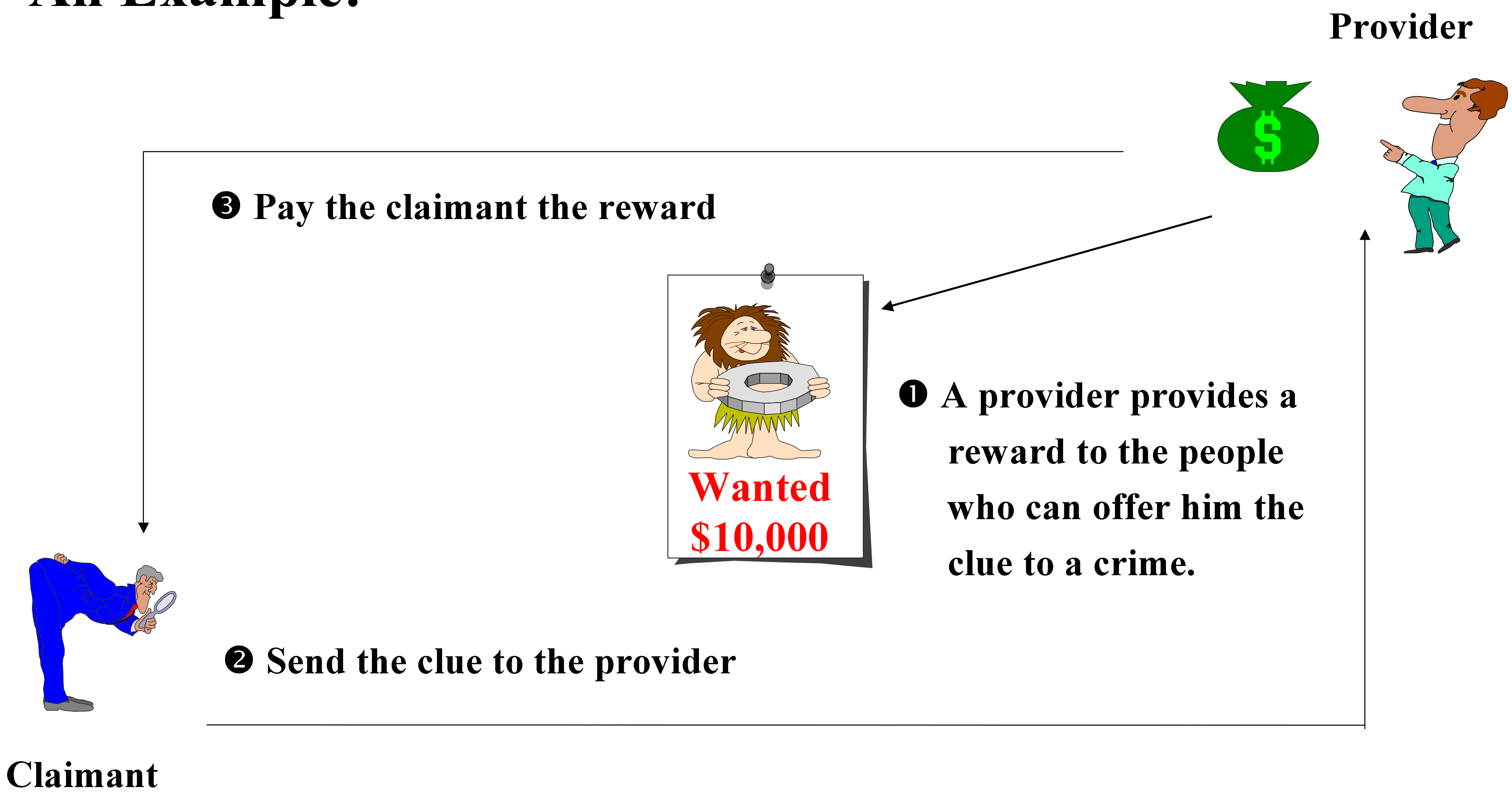
$$s^4 \overset{?}{\equiv}_n H_{1:55}(m_2)(c^2+1)$$

$$s^4 \overset{?}{\equiv}_n H_{1:35}(m_1)(c^2+1)$$

■ Anonymous Rewarding Schemes

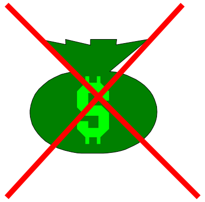


An Example:


Provider
❸ Pay the claimant the reward
Wanted
$10,000
❶ A provider provides a reward to the people who can offer him the clue to a crime.
❷ Send the clue to the provider
Claimant

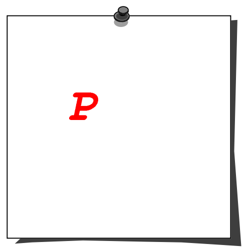
Possible Weaknesses:

Provider



The Trust Weakness:

The provider may refuse to pay the reward.



❶ Publish a problem *P*

The Privacy Weakness:

The identities of claimants may be not protected.

❷ Send the solution λ of *P*

Claimant

The Proposed Rewarding Scheme

- **A Reward Provider**

  A person who publishes a problem and offers a reward.

- **A Reward Claimant**

  A   has the solution of the problem and claims the reward.

- **A Verifier**

  It has enough power to verify the solution of the problem, and it does not reveal the solution to the provider before he pays the reward.

  For example: the credit bureaus or the government

- **A Bank**

  It issues electronic cash.

Claimant **Verifier** **Provider** **Bank**

1. Publish a problem.

2. E_V(solution), H(solution), a blinded message

3. S_P(E_V(solution), blinded message, sequence no.)s

4. Verify solutions.

5. The first qualified one

6. The blinded message

7. A blinded cash

8. Publish the blinded cash.

9. Verify the blinded cash.

10. The selected solution and all previous solutions

11. Unblind the blinded cash.

Discussions:

- The identities of the reward claimants are protected against anyone else.
- The reward provider cannot decline the selected claimant his entitled reward after the provider obtains the solution.
- The verifier cannot select a claimant other than the first qualified one to obtain the reward without being detected by the provider.

■ Information Attachable Electronic Cash

X is the underlying blind signature scheme.

***M* = {1, 2, …, *t* }** is the set of messages.

G and ***H*** are two public one-way hash functions.

$G^i(u) = G(G^{i-1}(u))$ with $i \in M$ where $G^0(u) = u$.

$H^i(v) = H(H^{i-1}(v))$ with $i \in M$ where $H^0(v) = v$.

User		**Bank**
$r, u, v \in R$		
$\alpha = (G^t(u) \| H^t(v))$		
	$B_X(\alpha, r)$ →	
		Signing :
		$\beta = S_X(B_X(\alpha, r))$
	← β	

Unblinding : $U_X(\beta, r) = S_X(\alpha)$

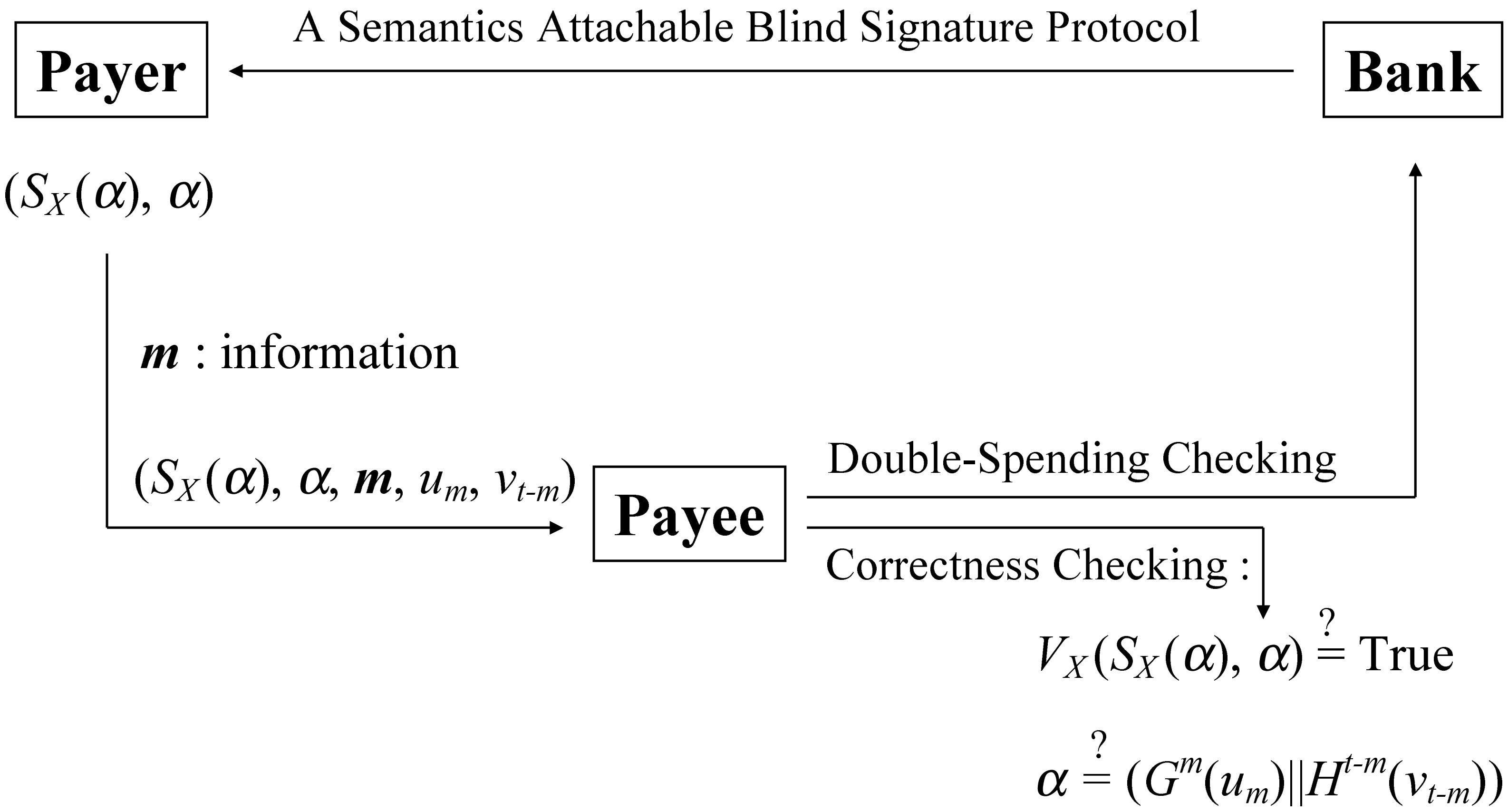
Choose $\boldsymbol{m \in M}$.

$u_m = G^{t\text{-}m}(u)$ and $v_{t\text{-}m} = H^m(v)$

Signature : $(S_X(\alpha), \alpha, m, u_m, v_{t\text{-}m})$

Verifying : $\boldsymbol{V_X(S_X(\alpha), \alpha) \stackrel{?}{=}}$ True

$$\boldsymbol{\alpha \stackrel{?}{=} (G^m(u_m) \| H^{t\text{-}m}(v_{t\text{-}m}))}$$


A Semantics Attachable Blind Signature Protocol
Payer
Bank
$(S_X(\alpha), \alpha)$
$\boldsymbol{m}$: information
$(S_X(\alpha), \alpha, \boldsymbol{m}, u_m, v_{t-m})$
Payee
Double-Spending Checking
Correctness Checking :
$V_X(S_X(\alpha), \alpha) \stackrel{?}{=} \text{True}$
$\alpha \stackrel{?}{=} (G^m(u_m) || H^{t-m}(v_{t-m}))$

Anonymous Electronic Voting

■ A User Efficient Electronic Voting Scheme

Voter ← The proposed blind signature protocol — **Center**

m = intention

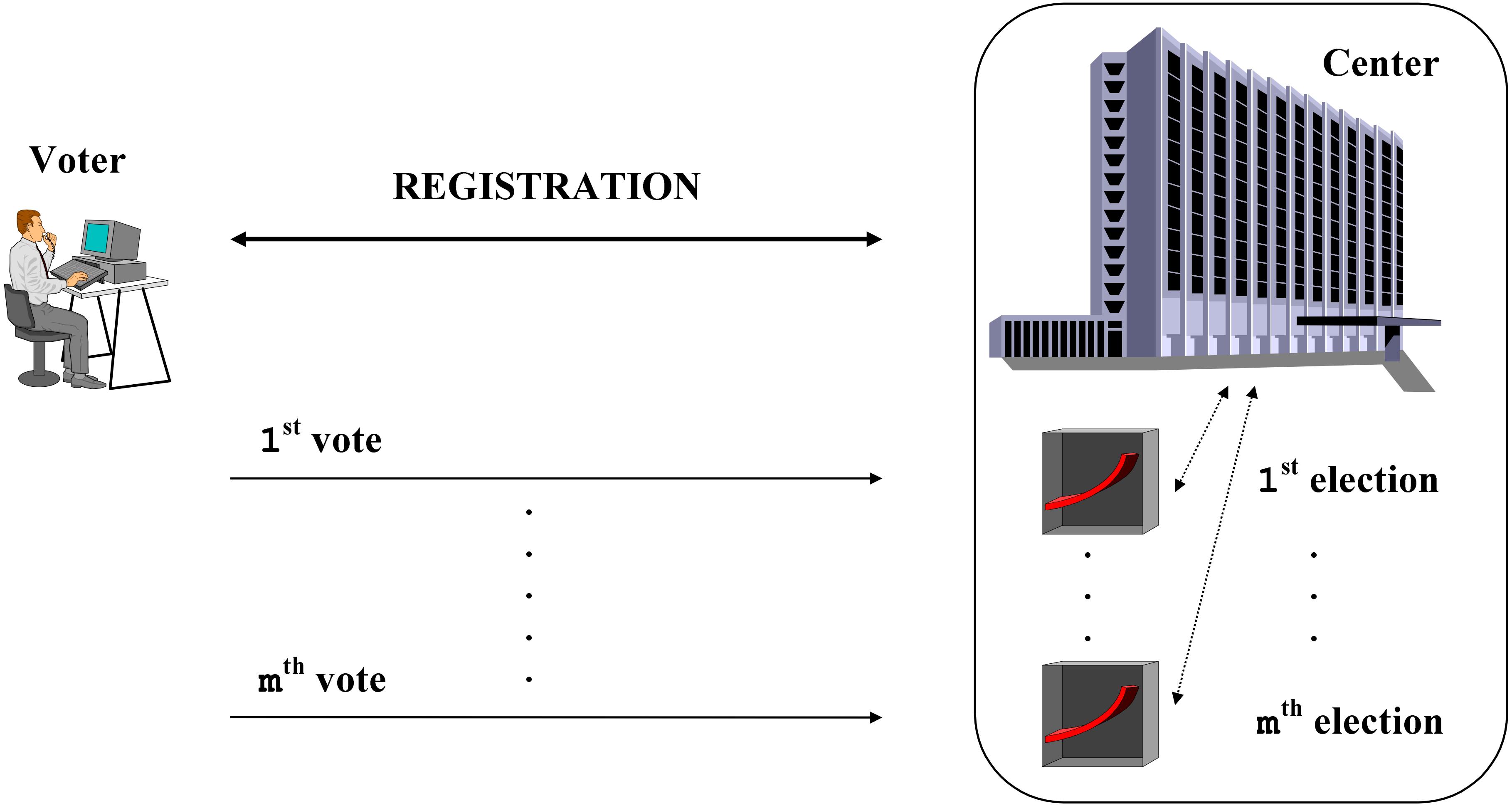
vote: (c, m, s)

(c, m, s) → Anonymous Channel

Verify $s^4 \overset{?}{\equiv}_n H(m)(c^2+1)$.

Publish (c, m, s).

■ Multi-Recastable Electronic Voting



The Proposed Multi-Recastable Voting Scheme

❶ **Initialization**

❷ **Requesting : Encrypted messages** →

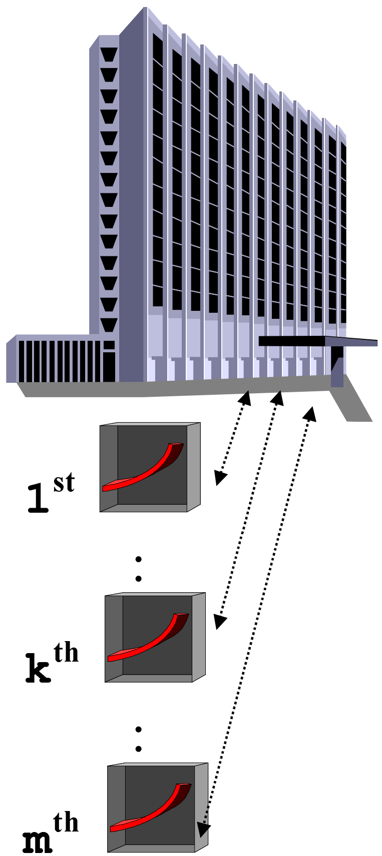
Voter

❸ **Registration : Blind signatures** ←

❹ **Extraction : Obtaining multi-recastable tickets (m-tickets)**

❺ **The k^{th} election $(1 \leq k \leq m)$** →

❻ **The k^{th} tally $(1 \leq k \leq m)$**



❶ Initialization :

p_1, p_2, p_3, p_4 are large primes.

Publish $n_{aff} = p_1 p_2$, $n_{opp} = p_3 p_4$, $n = n_{aff} n_{opp}$.

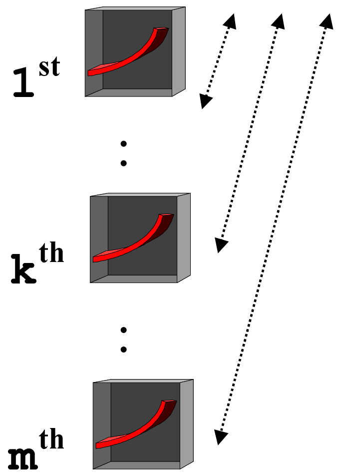
❷ Requesting : Encrypted messages

❸ Registration : Blind signatures

❹ Extraction : Obtaining m-tickets

❺ The kth election $(1 \le k \le m)$

❻ The kth tally $(1 \le k \le m)$



❷ Requesting :

$H,\ R_0,\ R_{k,aff},\ R_{k,opp}$: **random integers** $(1 \le k \le m)$

$w_0 \bmod n = (H \| RE_0 \| R_0)$ → **authentication message**

$w_k \bmod n_{aff} = (H \| RE_k \| R_{k,aff})$ → **affirmative message**

$w_k \bmod n_{opp} = (H \| RE_k \| R_{k,opp})$ → **oppositive message**

Encrypted Message (EM) $\equiv_n (u^2 + v^2) r^{2^{m+2}} \prod_{i=0}^{m} w_i^{2^{i+1}}$

❶ Initialization :

$n_{aff} = p_1 p_2$

$n_{opp} = p_3 p_4$

$n = n_{aff} n_{opp}$

❸ Registration : Blind signatures

❹ Extraction : Obtaining m-tickets

❺ The k^{th} election $(1 \le k \le m)$

❻ The k^{th} tally $(1 \le k \le m)$

❶ Initialization :

$$n_{aff} = p_1 p_2$$

$$n_{opp} = p_3 p_4$$

$$n = n_{aff} n_{opp}$$

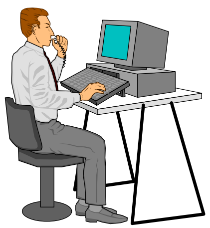
❷ Requesting :

$$w_0 \bmod n = (H \| RE_0 \| R_0)$$

$$w_k \bmod n_{aff} = (H \| RE_k \| R_{k,aff})$$

$$w_k \bmod n_{opp} = (H \| RE_k \| R_{k,opp})$$

$$EM \equiv_n (u^2 + v^2) r^{2^{m+2}} \prod_{i=0}^{m} w_i^{2^{i+1}}$$



❸ Registration :

$$\mathbf{t} \equiv_{\mathbf{n}} \sqrt[2^{m+2}]{(\mathbf{EM})(\mathbf{x}^2 + \mathbf{y}^2)(\mathbf{uy} - \mathbf{vx})\mathbf{b}^{2^{m+1}}}\,)^{-2}$$

❹ Extraction : Obtaining m-tickets

❺ The k^{th} election $(1 \le k \le m)$

❻ The k^{th} tally $(1 \le k \le m)$

❷ Requesting :

$$w_0 \bmod n = (H \| RE_0 \| R_0)$$

$$w_k \bmod n_{aff} = (H \| RE_k \| R_{k,aff})$$

$$w_k \bmod n_{opp} = (H \| RE_k \| R_{k,opp})$$

$$EM \equiv_n (u^2+v^2) r^{2^{m+2}} \prod_{i=0}^{m} w_i^{2^{i+1}}$$

❶ Initialization :

$$n_{aff} = p_1 p_2$$

$$n_{opp} = p_3 p_4$$

$$n = n_{aff} n_{opp}$$

❸ Registration :

$$t \equiv_n \sqrt[2^{m+2}]{(EM)(x^2+y^2)(uy-vx)b^{2^{m+1}}}\ )^{-2}$$

❹ Extraction :

$$\mathbf{s \equiv_n r^{-1} bt}$$

$$\mathbf{c \equiv_n (ux+vy)(uy-vx)^{-1}}$$

$$\textbf{m-ticket} = \left(\mathbf{s}, \prod_{i=0}^{m} \mathbf{w}_i^{2^{i+1}}, \mathbf{c}\right)$$

Extract $\boldsymbol{\beta_0 \equiv_n \sqrt[4]{w_0^2(c^2+1)}}$ **from s.**

Submit $\boldsymbol{(\beta_0, w_0, c)}$ **to the authority.**

The center verifies :

$$\boldsymbol{(\beta_0)^4 \stackrel{?}{=} (w_0)^2(c^2+1) \pmod{n}}$$

❺ The k^{th} election $(1 \le k \le m)$

❻ The k^{th} tally $(1 \le k \le m)$

❶ Initialization :

$$n_{aff} = p_1 p_2$$

$$n_{opp} = p_3 p_4$$

$$n = n_{aff} n_{opp}$$

❷ Requesting :

$$w_0 \bmod n = (H \| RE_0 \| R_0)$$

$$w_k \bmod n_{aff} = (H \| RE_k \| R_{k,aff}) = \mathbf{w_{k,aff}}$$

$$w_k \bmod n_{opp} = (H \| RE_k \| R_{k,opp}) = \mathbf{w_{k,opp}}$$

$$EM \equiv_n (u^2 + v^2) r^{2^{m+2}} \prod_{i=0}^{m} w_i^{2^{i+1}}$$

❸ Registration :

$$t \equiv_n \sqrt[2^{m+2}]{(EM)(x^2 + y^2)(uy - vx) b^{2^{m+1}}}\,)^{-2}$$

❹ Extraction :

$$s \equiv_n r^{-1} b t$$

$$c \equiv_n (ux+vy)(uy-vx)^{-1}$$

$$\text{m-ticket} = (s, \prod_{i=0}^{m} w_i^{2^{i+1}}, c)$$

❺ **The k^{th} election $(1 \le k \le m)$:**

Extract $\beta_k \equiv_n \sqrt{w_k \sqrt{\ldots \sqrt{\ldots}}}$ from s.

inten = aff or opp.

$$\beta_{k,inten} = \beta_k \bmod n_{inten}.$$

The k^{th} vote = $(\beta_{k,inten}, w_{k,inten})$

❻ The k^{th} tally $(1 \le k \le m)$

❶ Initialization :

$$n_{aff} = p_1 p_2$$
$$n_{opp} = p_3 p_4$$
$$n = n_{aff} n_{opp}$$

❷ Requesting :

$$w_0 \bmod n = (H \| RE_0 \| R_0)$$
$$w_k \bmod n_{aff} = (H \| RE_k \| R_{k,aff}) = w_{k,aff}$$
$$w_k \bmod n_{opp} = (H \| RE_k \| R_{k,opp}) = w_{k,opp}$$
$$EM \equiv_n (u^2 + v^2) r^{2^{m+2}} \prod_{i=0}^{m} w_i^{2^{i+1}}$$

❸ Registration :

$$t \equiv_n \sqrt[2^{m+2}]{(EM)(x^2 + y^2)(uy - vx) b^{2^{m+1}}}^{\,-2}$$

❹ Extraction :

$$s \equiv_n r^{-1} b t$$
$$c \equiv_n (ux+vy)(uy-vx)^{-1}$$
$$m\text{-ticket} = (s, \prod_{i=0}^{m} w_i^{2^{i+1}}, c)$$

❺ The k^{th} election $(1 \le k \le m)$:

Extract $\beta_k \equiv_n \sqrt{w_k \sqrt{\ldots \sqrt{\ldots}}}$ from s.

$inten = aff$ or opp.

$\beta_{k,inten} = \beta_k \bmod n_{inten}$.

The k^{th} vote $= (\beta_{k,inten}, w_{k,inten})$

❻ The k^{th} tally $(1 \le k \le m)$:

Verification :

$$\beta_{k,inten}^{2^{k+2}} \stackrel{?}{\equiv}_{n_{inten}} w_{k,inten}^{2^{k+1}} \cdots$$

Discussions:

- Only one round of registration action is needed for a voter to participate in a sequence of different elections.

- If both affirmative and oppositive votes are cast by a voter in an election, then they can be detected.

- All of the votes cast by a voter in a sequence of elections can be linked together by the tally center.

■ An Efficient Election Scheme for Resolving Ties

X is the underlying blind signature scheme.

$\boldsymbol{M = \{1, 2, \ldots, t\}}$ is the set of messages.

G and ***H*** are two public one-way hash functions.

$G^i(u) = G(G^{i-1}(u))$ with $i \in M$ where $G^0(u) = u$.

$H^i(v) = H(H^{i-1}(v))$ with $i \in M$ where $H^0(v) = v$.

User | **Center**

i : intention

$r,\ u,\ v \in R$

$\alpha = (G^t(u) \| H^t(v))$

$\xrightarrow{B_X(i\|\alpha,\ r)}$

Signing :

$\beta = S_X(B_X(i\|\alpha,\ r))$

$\xleftarrow{\beta}$

Unblinding : $U_X(\beta,\ r) = S_X(i\|\alpha)$

Vote = $(S_X(i\|\alpha),\ (i\|\alpha))$ $\longrightarrow$

Verifying :

$V_X(S_X(i\|\alpha),(i\|\alpha)) \stackrel{?}{=} \text{True}$

Ties :

Choose $m \in M$.

$u_m = G^{t-m}(u)$ **and** $v_{t-m} = H^m(v)$

Re-voting : Submit $(\alpha,\ m,\ u_m,\ v_{t-m})$.

Verifying :

$\alpha \stackrel{?}{=} (G^m(u_m) \| H^{t-m}(v_{t-m}))$

Voter ← A Semantics Attachable Blind Signature Protocol — **Center**

Voting: $(S_X(i\|\alpha), i\|\alpha)$ →

$V_X(S_X(i\|\alpha), (i\|\alpha)) \stackrel{?}{=} \text{True}$

Re-voting: $(\alpha, \boldsymbol{m}, u_m, v_{t\text{-}m})$ →

$\alpha \stackrel{?}{=} (G^m(u_m)\|H^{t\text{-}m}(v_{t\text{-}m}))$

■ A Receipt Free Electronic Voting Scheme

- It is easier to buy votes in a typical electronic election.

- In a receipt free electronic voting scheme, every voter cannot convince any other voter of the value of his vote.

- The proposed receipt free voting scheme is based on probabilistic encryption methods (PEM) and blind signatures.

Probabilistic Encryption Methods (PEM)

Encryption:

- For every message m, the encryption $E(m)$ is an element in $R_E(m)$
- $E(m_1)E(m_2) = E(m_1+m_2)$.
- $E(m_1)E(m_2)^{-1} = E(m_1-m_2)$.

Decryption:

- Given $z \in R_E(m)$, the decryption $D(z) = m$.
- A certificate $D'(z)$ can prove that $z \in R_E(m)$.

Protocol *Show_Zero*(*a*):

- If $a \in R_E(0)$, the center can convince the voter in a voting booth that a is indeed in $R_E(0)$ without revealing the certificate $D'(a)$.

Protocol *Show_Zero_One*(*a*):

- If $a \in R_E(0) \cup R_E(1)$, the center can convince all voters that a is indeed in $R_E(0) \cup R_E(1)$ without revealing the certificate of $D'(a)$.

- The protocol cannot show that a is exactly in $R_E(0)$ or $R_E(1)$.

The Proposed Voting Protocol

$\boldsymbol{M}$ is the underlying set of messages.

$\boldsymbol{R}$ is a finite set of random integers.

$\boldsymbol{S_X : M \rightarrow M}$ is the signing function kept secret by the center.

$\boldsymbol{V_X : S_X(M) \times M \rightarrow}$ **{true, false}** is the verification formula.

$\boldsymbol{B_X : M \times R \rightarrow M}$ is the blinding function.

$\boldsymbol{U_X : S_X(M) \times R \rightarrow S_X(M)}$ is the unblinding function.

User | **Center**

$m \in \{0, 1\}$

$E(m) \in R_E(m)$

$r \in R$

$B_X(E(m), r)$ →

Signing :

$b = E(0) \in R_E(0)$

(In a voting booth)

$\beta = S_X(B_X(E(m)b, r))$

← (β, b)

Perform *Show_Zero*(b).

$U_X(\beta, r) = S_X(E(m)b) = S_X(E(m'))$

$(S_X(E(m')), E(m'))$ ⋯→

Verify $V_X(S_X(E(m')), E(m')) \stackrel{?}{=}$ True.

Perform *Show_Zero_One*$(E(m'))$.

$A = \Pi_i E(m'_i) = E(\Sigma_i m'_i)$

Publish $D(A)$ and $D'(A)$.

Discussions:

- PEM + Interactive Proof Protocols → Freedom from Receipts
- Since every voter has no license of his vote, he cannot convince any other voter of the value of his vote.
- Blind Signatures + Anonymous Channels → Privacy Protection
- In the proposed scheme, the privacy of every voter is protected against anyone else.

Conclusions

■ Blind Signatures

- Efficiency: User Efficient Blind Signatures
  Low-Computation Partially Blind Signatures
  Efficient Fair Blind Signatures

- Variations: Divisible Blind Signatures
  Semantics Attachable Blind Signatures

■ Applications

- Digital Cash: Anonymous Rewarding Schemes
  Divisible Digital Cash
  Information Attachable Electronic Cash

- Electronic Voting: Multi-Recastable Electronic Voting
  Receipt Free Electronic Voting
  Efficient Elections for Resolving Ties

■ Future Research

- Enlarge the domain of the attached messages in the proposed information attachable electronic cash scheme.

- Design efficient methods to allow arbitrary-valued votes in the proposed multi-recastable voting and receipt free elections.