An RSA-based (*t*, *n*) Threshold Proxy Signature Scheme without any Trusted Dealer

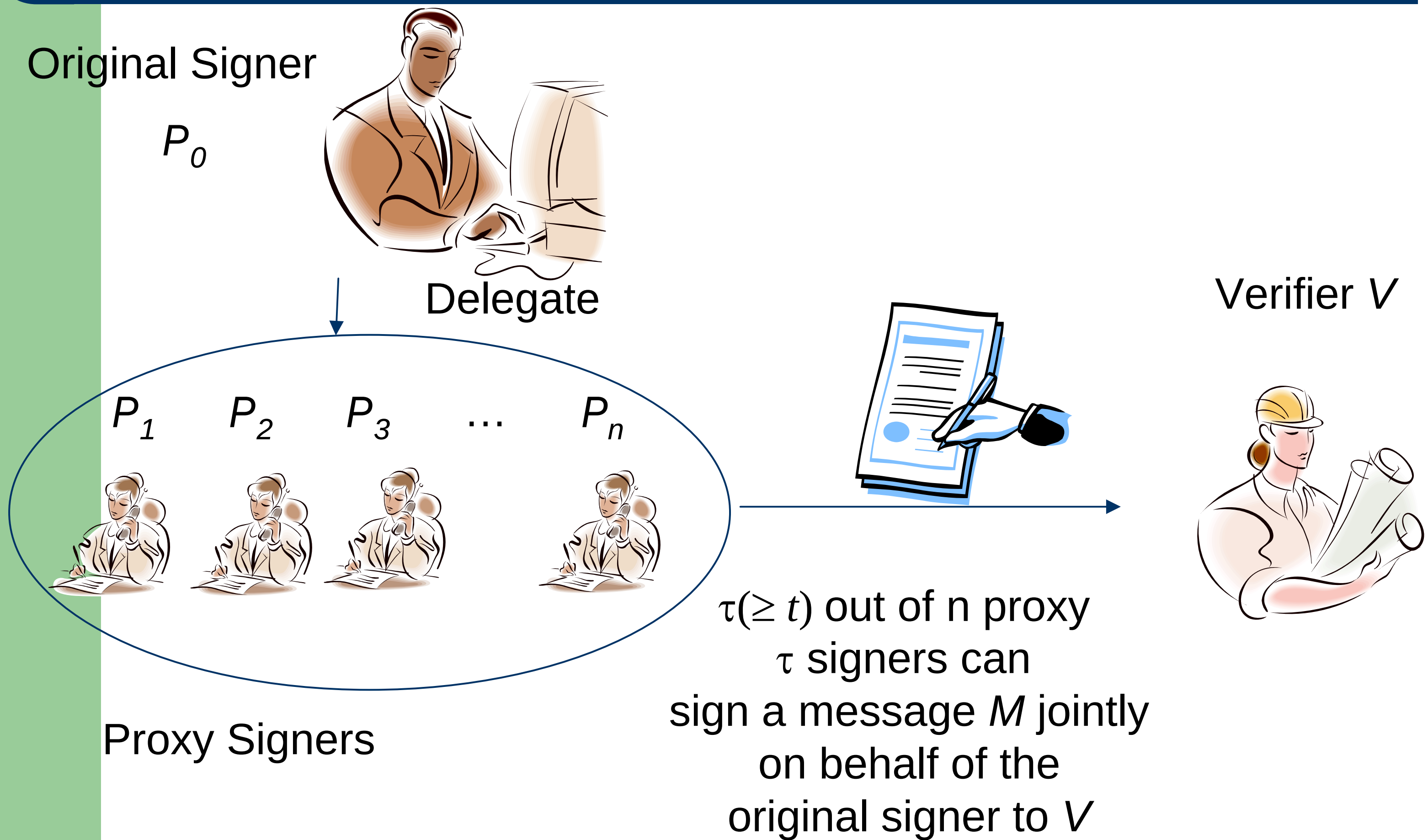
Pei-yih Ting
Xiao-Wei Huang

Department of Computer Science and Engineering,
National Taiwan Ocean University, Taiwan, R.O.C.
Institute of Information Science,
Academia Sinica, Taiwan, R.O.C.

Outline

- The threshold proxy signature
- Our proposed scheme
- The security requirements of the proxy signature
- Security analysis

The Threshold Proxy Signature



The Related Work

Original signer

Verifier

Proxy signers

The Related Work

Original signer

Verifier

Proxy signers

Trusted combiner

[KC05] Kuo, W.C., Chen, M.Y.: A Modified (t, n) Threshold Proxy Signature Scheme based on the RSA cryptosystem. In: Proceedings of the Third International Conference on Information Technology and Applications (ICITA), 2005.
[CC07] Chang, Y.F., Chang, C.C.: An RSA-based (t, n) Threshold Proxy Signature Scheme with Free-will Identities. International Journal of Information and Computer Security 1(1/2), 201–209, 2007.

Our Result

- Is the trusted combiner necessary on the threshold proxy signature based on RSA?
- We design an RSA-based threshold proxy signature scheme without any trusted dealer

Our Proposed Threshold Scheme (1)

P_0

$(e_0, n_0), d_0$ (e_0 is a prime greater than n)

$n_0=p_0q_0$

$p_0=2p_0'+1$

$q_0=2q_0'+1$

$m_0=p_0'q_0'$

The warrant w

n proxy signers:

P_i

$(e_i, n_i), d_i$

V

A simple construction

- Let the signature be

$$(h(w)^{h(m)})^{d_0}$$

- It can be verified by $((h(w)^{h(m)})^{d_0})^{e_0}$.
- We can not share d_0 or $d_0^{h(w)}$ directly since P_0's private key d_0 must be kept secret.
- Due to this fact, Chang's work is to share $(h(w)^{h(m)})^{d_0}$. But it leads to the combining process requires a trusted combiner.

Our construction

- Let the signature be

$$S=C^{Dh(m)}=((h(w)^{h(m)})^{d_0/D})^D$$

  where the proxy signing key D is a random number chosen by P_0.
- Let $C=h(w)^{d_0/D}$ be the proxy public key generated by P_0 initially.

Our construction

- In our construction, the signature is

$$S=C^{Dh(m)}=((h(w)^{h(m)})^{d_0/D})^{D}$$

D_1 D_2 … D_n

$C^{h(m)D_1}$

Our Proposed Threshold Scheme (2)

The proxy sharing protocol:

1. P_0 picks a and computes

$$D = a (\bmod \phi(n_0)),$$

$$E = e_0 (\bmod \phi(n_0))$$

$$C = h(w)^G (\bmod n_0), \text{ where } G = (DE)^{-1}.$$

2. P_0 secretly picks a polynomial

$$f(x) \;=\; D + r_1 x^1 + \ldots + r_{t-1} x^{t-1} (\bmod m_0)$$

and sends $D_i = f(i)$ to P_i secretly.

3. P_0 publishes $\{E, C, w, \sigma_w = h(E||C||w)^{d_0}\}$.

Our Proposed Threshold Scheme (3)

The proxy signature signing protocol:

1. P_i computes

$$S_i = (C^{h(m)})^{2\Delta D_i} \pmod{n_0}, \text{ where } \Delta = n!$$

$$\sigma_i = h(S_i)^{d_i} \pmod{n_i}.$$

$$L_i = \prod_{i,j \in T, j \neq i} \frac{-j}{i-j} (\bmod \phi(n_0))$$

Our Proposed Threshold Scheme (4)

The proxy signature combing protocol:

1. The proxy signers jointly compute

$$\overline{S} = \prod_{i \in T} S_i^{\,2\Delta L_i} \pmod{n_0} \left(= S^{4\Delta^2}\right)$$

$$L_i = \prod_{i,j \in T, j \neq i} \frac{-j}{i-j} (\bmod \phi(n_0))$$

2. Since gcd($4\Delta^2, E$)=1, there are $\tilde{a}, \tilde{b}$ such that $4\Delta^2 \tilde{a} + E\tilde{b} = 1$.

$$S = \overline{S}^{\tilde{a}} h(w)^{h(m)\tilde{b}} \pmod{n_0}.$$

3. The proxy signature is $\sigma=(S,\{\sigma_i\}_{i \in T})$.

Our Proposed Threshold Scheme (5)

The proxy signature verification protocol:

V checks

$$(\sigma_w)^{e_0} = h(E \| C \| w)(\bmod n_0),$$

$$S^E = h(w)^{h(m)} (\bmod n_0),$$

$$\sigma_i^{e_i} = h(S_i)(\bmod n_i) \text{ for all } i \text{ in } T.$$

Security Requirements

- Secrecy
- Proxy protected
- Unforgeability
- Non-repudiation
- Time Constraint
- Known signers

Security analysis

- The RSA assumption:
  - Given an RSA public key (n_0,e_0) and a ciphertext $c=m^{e_0} \bmod n_0$, it is hard to compute the plaintext m without the RSA private key.
- The composite-exponent RSA assumption:
  - Given an RSA public key (n_0,e_0), two integer factors E,D, i.e. $e_0=ED$, and a ciphertext $c=m^{e_0} \bmod n_0$, it is hard to compute the plaintext m without the RSA private key.

Security analysis

- Thm 1. The “RSA assumption” implies the “composite-exponent RSA assumption”

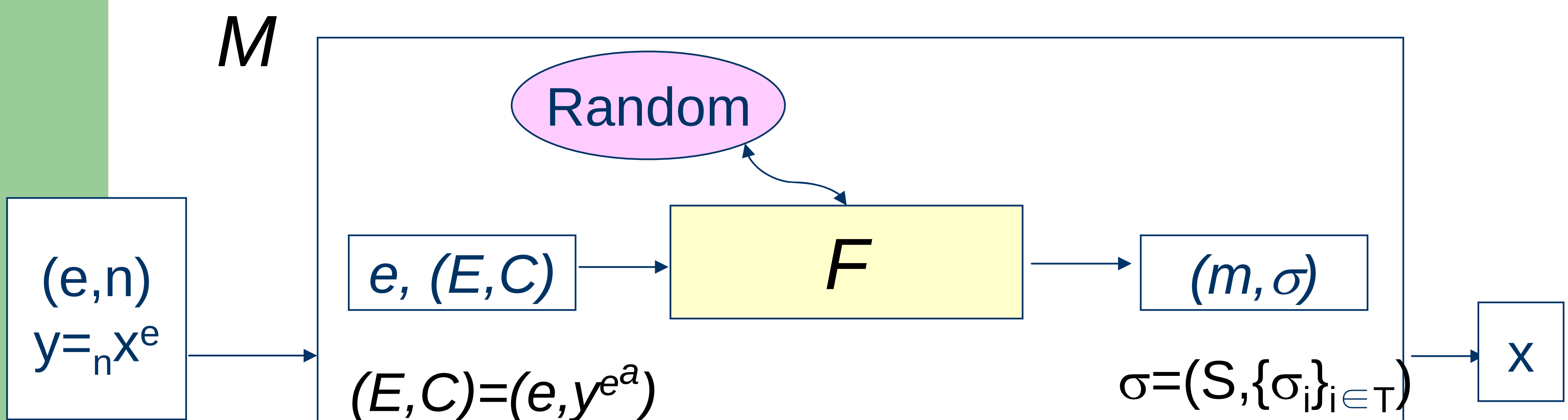
Security analysis

- Secrecy
  - The "composite-exponent RSA assumption" guarantees that it would be hard to find the signature of arbitrary message without the knowledge of *G* and keeps the key of the original signer secret.

Security analysis

- Unforgeability
  - The existential unforgeability under no message attack in the random oracle of the proposed scheme can be proven under the RSA assumption.
  - To further limit the potential dangers of chosen message attacks, we can invoke the constructions such as key-refreshing and authentication tree.

Security analysis



$(E,C)=(e,y^{ea})$

$\sigma=(S,\{\sigma_i\}_{i\in T})$
s.t. $S^{e_0}=h(w)^{h(m)}$

RO:

If *F* queries *w*, *M* returns *h(w)=y*,

Else *M* returns random number *r* and sets *r=1* in the *t*-th query, where t is a random choice.

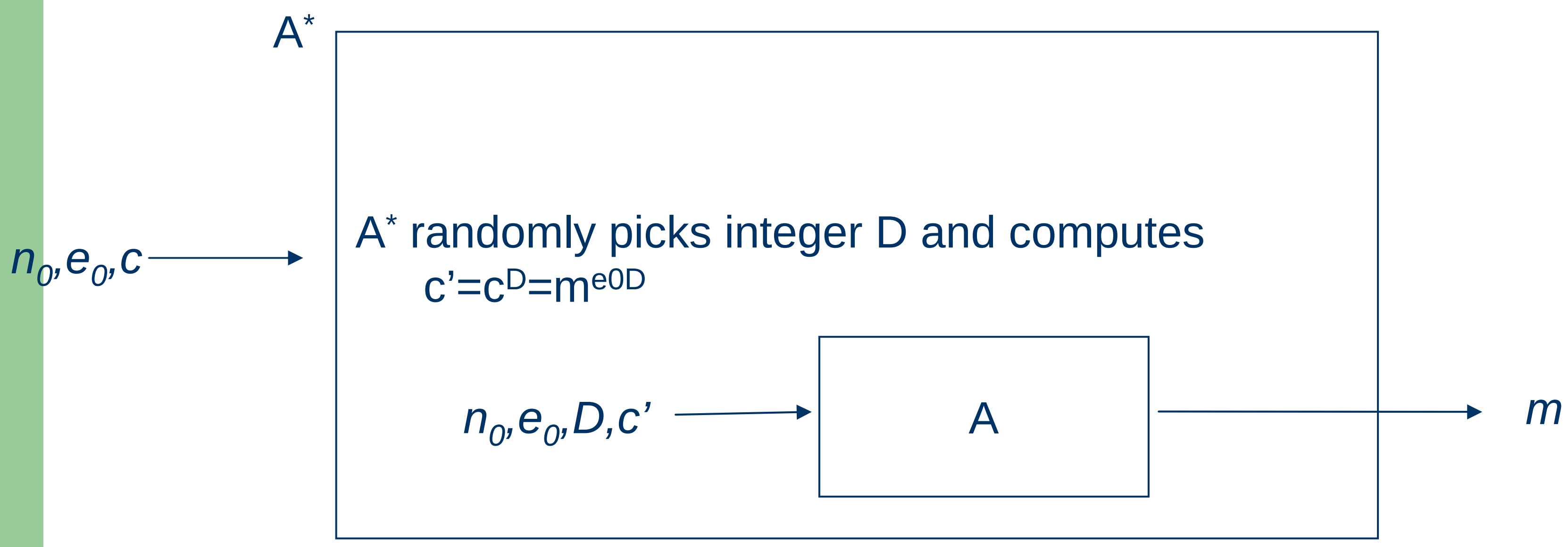
Full version of this paper.

Available at: http://iml3.cs.ntou.edu.tw/full_version_ISC.pdf

Thanks for your attentions!

Security analysis

- Thm 1. The “RSA assumption” implies the “composite-exponent RSA assumption”



Authentication Tree