

國立交通大學

電機學院與資訊學院 資訊學程

碩士論文

門檻式單次代理簽章之方法

Threshold One-Time Proxy Signature Scheme



研究生：蔡志村

指導教授：葉義雄 教授

中華民國九十五年一月

門檻式單次代理簽章之方法
Threshold One-Time Proxy Signature Scheme

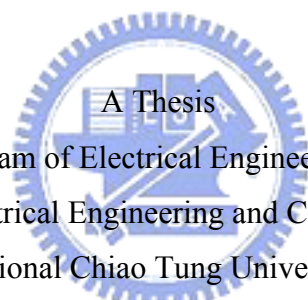
研究生：蔡志村

Student : Chih-Chun Tsai

指導教授：葉義雄

Advisor : Dr. Yi-Shiung Yeh

國立交通大學
電機學院與資訊學院 資訊學程
碩士論文



Submitted to Degree Program of Electrical Engineering and Computer Science

College of Electrical Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Computer Science

January 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年一月

國立交通大學電機學院與資訊學院 資訊學程（研究所）碩士班

摘 要

最近，很多傑出的代理簽章方案已經被提議。對於先前更多的安全議題來說，這裡我們提出一種更安全的方法。它允許一個或更多實體共同合作扮演出一名代理簽章者的角色來代表原始簽章者簽章的權力。我們使用門檻值的特性並且用信任單元和組合 PHF 物件來提出我們的方案。在一個 (w, n) 門檻值代理簽章裡，原始簽章者能授予 n 個代理簽章者簽核訊息的能力，即任 w 個或多個代理簽章者共同合作產生一個代理簽章來代表原始簽章者，但是 $(w-1)$ 個或更少的代理簽章者就不能運作。繼上述相同的模型之後，我們首先提出一個新的門檻式單次代理簽章的方式基於單向功能的簽章方案。這個工作原理不同於其他相關的代理簽章方案。除了給這把代理金鑰提供機密性防護之外，這種方法對全部有關係的實體提供了不可否認的機制。特別是，它保護代理簽章者以防止原始簽章者否認授權給代理簽章者代表其簽章的能力，及防止代理簽章者否認其所產生的代理簽章，以及防止簽章接受者否認代理簽章者所產生的代理簽章。我們的方案仍然保持著單次簽章的快速簽章驗證特性和較低計算能力特性，並且適用於各式各樣的無線網路應用。

(Threshold One-Time Proxy Signature Scheme)

student : Chih-Chun Tsai

Advisors : Dr. Yi-Shiung Yeh

Degree Program of Electrical Engineering Computer Science National Chiao Tung University

ABSTRACT

Recently, many excellent proxy signature schemes have been proposed. For advanced security issue, here we present a more secure method. It allows the delegation of signature power to one or more entities that jointly play the role of a proxy signer. We use characteristic of threshold and trust party and combinatorial object PHF to propose our scheme. In a (w, n) threshold proxy signature, the original signer can delegate the power of signing messages to n proxy signers so that any w or more proxy signers cooperatively generate a proxy signature on behalf of the original signer, but $(w-1)$ or less of them cannot. Following the same model, we first propose a new threshold proxy one-time signature scheme based on one-way functions. This work is different from other related proxy signature schemes. In addition to providing confidentiality protection to the proxy key, the method provides non-repudiation services to all the parties involved. In particular, it protects proxy signer against repudiation of signature delegation of the original signer, repudiation of proxy signature generation of the proxy signer and repudiation of receipt of the proxy signature of the signature recipient. Our scheme still preserves the fast signature verification and low computation power of one-time signature, and it is suitable for various wireless applications.

誌 謝

在交大電機學院與資訊學院碩士在職專班學習進修的二年裡，對我們這種白天在職場上班，晚上在學校進修的同儕來講，實在是非常的辛苦，但是此種辛苦是值得的，因為大家都是經過職場的歷練後，才再選擇重拾課本，追求自我所欠缺的知識及技能，所以對此種進修的機會會更加的珍惜與努力。在這段時間裡，我要感謝所有指導過我的教授，包括曾煜棋教授、陳登吉教授、林一平教授、蔡文能副教授、曾建超教授、簡榮宏教授、鍾崇斌教授、楊武教授等對我的教導，讓我能學習到網路通訊相關領域的基本技能；當然也要特別感謝我的指導教授—葉義雄教授，在我撰寫論文期間，對於學術研究給我相當大的指引與支援，並適時地教導我一些網路安全、密碼學的研究課題，使我的研究能夠有一明確的方向，在此特別致上我最誠摯的感謝。

另外要特別感謝博士班的銘智學長給予我無私的幫忙，不管是對於技術論點方面或者是論文寫作方面，在整個研究過程中均給予相當大的指導與建議，使我能夠充分了解並撰寫出該篇論文所要表達的論點。

除此之外，還有要感謝我們 92 級的同班同學，大家在就學期間互相的支援與合作，使我們每一科均能順利過關；還有要感謝與我工作同單位同時也是此在職專班的學長一芳、明傑，在學習的課業上及工作上給予我相當程度的支援與幫忙，還有感謝開基對這篇論文的指導與建議，當然還要感謝美娟及威亭不厭其煩地幫我校稿，使我能夠順利完成這篇論文。

最後，當然要好好地感謝我的老婆瑞蘭，在我這二年多來的在職進修過程中，幫我照顧好我們倆的兩個寶貝兒女，讓我能無後顧之憂的去專心研讀學問，在精神上與生活上給我相當大的鼓勵與支持。

蔡志村 謹識

中華民國九十四年十二月八日

目 錄

中文提要	i
英文提要	ii
誌謝	iii
目錄	iv
表目錄	vi
圖目錄	vii
符號說明	viii
一、	Introduction	1
1.1	Motivation.....	1
1.2	The past history of Digital Signature.....	2
1.3	About this Thesis.....	6
二、	Relative Model	7
2.1	Proxy signature.....	7
2.2	Type of proxy Signature.....	8
2.3	Threshold Proxy Signature.....	11
2.4	A (w, n) Threshold Proxy One-Time Signatures.....	12
三、	One-Time Signature Scheme	13
3.1	Lamport One-Time Signature Scheme.....	13
3.2	Improving Lamport one-time signature scheme by Chang.....	15
四、	Threshold MAC	19
4.1	Perfect Hash Families (PHF) and Cover Free Family (CFF)	19
4.2	Threshold CFF MAC scheme.....	19
五、	Proposed Threshold One-Time Proxy Signature Scheme	21
5.1	A new (w, n) Threshold One-Time Proxy Signature...	21
5.2	Key Generation.....	21
5.3	Proxy Signature Generation.....	24
5.4	Proxy Signature Verification.....	26
六、	System Analysis and Application	29
6.1	Security analysis.....	29
6.1.1	Correctness.....	29
6.1.2	Security.....	29
6.2	Comparison.....	34
6.3	Application of our proposed scheme.....	35
七、	Conclusion	37
7.1	Conclusion.....	37

7.2	Future Work.....	37
參考文獻	38



表 目 錄

Table 6.1	The comparison of Al-Ibrahim's TOTP and our proposed scheme.....	35
-----------	------------------------------------------------------------------	----



圖 目 錄

Figure 5.1	The Key Generation.....	24
Figure 5.2	The Proxy Signature Generation.....	26
Figure 5.3	The Proxy Signature Verification.....	27
Figure 6.1	Examples of application for proposed method.....	36



符 號 說 明

(w,n)	: n proxy signers so that any w or more proxy signers cooperatively generate a proxy signature on behalf of the original signer
TP	: Trust Party
SK	: Private Key
PK	: Public Key
PHF	: Perfect Hash Families
PHF($N;n,m,w$)	: N : N rows, n : n cols, $N \times n$ array m : max number in $N \times n$ array w : group number for threshold
CFF	: Cover Free Family
(n,m,w) -CFF	: n : n cols m : max number in $N \times n$ array w : group number for threshold
Q	: Private keys generates from original signer
B_q	: Combine PHF & $h(i,j,q)$ array
(l, B_l)	: proxy signing keys
P_q^I	: phase I public keys for q
P_q^{II}	: phase II public keys for q
\overline{B}_w	: the union of any w columns in B_q
ID_j	: proxy signer j 's secure identifier
$(j_1, j_2, \dots, j_w, ID_{j_1}, ID_{j_2}, \dots, ID_{j_w}, H^2(\overline{B}_w, q))$: final public keys, every w -subset $\{B_{j_1}, B_{j_2}, \dots, B_{j_w}\}$ (corresponding to $\{ID_{j_1}, ID_{j_2}, \dots, ID_{j_w}\}$) and for all $q \in Q$.
BIBD	: Balanced Incomplete Block Design
(v, b, r, k, λ) -BIBD	: is a set system (X, F) where X has v elements (or points) and F is a collection of b k -subsets (blocks) of X
$K_{j_i} H_{j_i}(q)$	Threshold Proxy Signature

一、Instruction

1.1 Motivation

Proxy signatures were first proposed by Mambo et al. [10, 12]. They defined three classes of proxy signature schemes: full delegation, partial delegation, and delegation by warrant schemes. A full delegation scheme assumes that a proxy signer is given the same signing keys that the original signer has. So, the proxy signer has the same signing capability as the original signer. A signature with partial delegation [10, 12, 17] allows the original signer using a original signing key to generate proxy signing keys, so their signatures are distinguishable. Hence, the original signer can delegate the power of a proxy signer in such a way. A signature with partial delegation by warrant limits the range of messages a proxy signer can sign by an additional piece of message (called a warrant). This type of delegation has proposed in [13, 18]. Furthermore, Wang et al. [20] classified proxy signature schemes into proxy-unprotected and proxy-protected schemes dependent on whether an original can generate a validate proxy signature or not. Following the development, there have been many threshold proxy signature schemes [7, 8, 16, 21] proposed for fitting various practical situations. Unlike Mambo et al.'s proxy signature, a threshold proxy signature allows the original signer to delegate her/his signing capability to a group of proxy signers. In [7, 8, 16, 21], they used the threshold Shamir secret sharing method to share secret proxy signing keys and the homomorphism property of traditional authenticating schemes [5, 15] to combine all the partial proxy signatures,

which are generated by the share of secret proxy signing keys, into a valid proxy signature.

One-time signature schemes were first proposed by Rabin [14] and Lamport [9] and are based on one-way functions. With their fast signature verification and low computation power, they have attracted more and more attention, as an ideal option for various wireless applications that use resource-constrained devices such as mobile phones, PDAs etc. Following the history of the traditional signature technology based on public-key cryptography development, the proxy and threshold signature based on one-way functions were also important for various wireless applications. To our best knowledge, there have been three schemes [1, 3, 19] based on one-way functions proposed for proxy signature. In [1], the authors also proposed a threshold proxy signature scheme based on one-way functions. However, their model is different from the previous works [4, 7, 8, 16, 19 and 21]. In the (w, n) threshold proxy one-time signature scheme of [1], the original signer is the group of n signers, and the proxy signers are any w signers. Therefore, their scheme is still a threshold one-time signature scheme regardless of their model. In this paper, we present a threshold proxy-protected signature scheme following the model, that an original signer shares her/his proxy signing key to a group of n proxy signers and any w or more partial proxy signatures generated by w or more proxy signers can be combined into a valid signature, of previous works (called original model).

1.2 The past history of Digital Signature

An original signer allows a designated person, called a proxy signer,

to sign a message on behalf of original signer. Proxy signatures have two properties, one is unforceability, and another is verifiability. Proxy signature is original singer can not sign the document cause by some reason. And then the sign authority transferred to another proxy signer. Proxy singer sign the document efficiency same as original signer sign the document. These signatures have a problem what if original signer purposely falsely incriminate proxy signer. The proxy signature is non-repudiation. So the Mambo *et al* [10] proposed a method, call a Proxy-protected Proxy signature. This kind of stamped signature is mainly, the agent joins him self's own gold key of secret (Secret key) at the time of stamped signature, when act for stamped signature in verification except that the persons who can verify primitive stamped signature agree to this stamped signature, because sign to join the agent's gold key of secret in the course, so can verify the agent at the same time, frame the agent on purpose for the person who prevents primitive stamped signature from.

In the society now, an official document often needs through many people's signature just validity and responsible for together, so, a lot of are proposed the colony and several signature technology that many people lead. In 1991, Chaum and Heyst [22] propose the signature of a kind of new-type attitude, is called colony's signature. The so-called colony's signature is the mechanisms of a kind of combination evidence (Credential) and member's identity authentication (Authentication). Utilize this mechanism, can prove to the person who prove that he belongs to this colony while belonging to a certain member of this colony and don't will expose its identity. When there are disputes to take place,

the identity of signature of this colony signature person can just be by its identity that colony's administrator (Group authority) or but all member's cooperation are open. We can sum up the characteristic with the following of colony's signature:

- (1) Only by belonging to the members of this colony can it sign.
- (2) The recipient of signature can verify the exactness of this colony signature, the identity of the person who calculates signature in but unable to be signed by this colony.
- (3) Have dispute emerge, is it can calculate by colony this who sign to sign. However, only colony's administrator or the members of colonies work in concert and can calculate.

But, if colony members change in their system, every member must alter its gold key. In addition, colony of them stamped signature system unable to let colony administrator prove colony stamped signature the lucky number identity of chapter person. Colony stamped signature system proposed the above belong to getting interactive system all, so efficiency it's good. Several is improved the mechanism to also propose, but mostly unable to meet all characteristics afterwards. In 1999, Camenisch and Michels [23] proposed a method of improving. But, in their system, have the following disappearances: Its calculation amount and information are also too large in length.

In 1997, the people, such as Park [24], etc. proposed that one regards identity as the colony stamped signature system of the base. The major advantage of this system is the open gold key of the person who is signed is but information need not be verified in its identity (ID), so not need to set up one by common authentication and disclosing the gold key

file enormously publicly of letter unit. But there is a serious problem in this colony stamped signature system which regards identity as the base; Namely the persons who sign must use the identity gold key of the members of colonies while signing, cause colony members to lose efficiency like changing colony's stamped signature. In addition, this length of stamped signature produced of colony stamped signature system will increase with member's figure.

Threshold signature call colony lead stamped signature (Group-oriented signature) [25, 26, 27]. And it is that members in the colony signed the same file that the threshold is worth several stamped signature ways, but the threshold is worth not needing all members in the colony to all participate in several stamped signature, only have to exceed threshold value in number of members. It is that the person who proves does not know that there are those members that participate in signing that it has a characteristic in addition. Because the threshold is worth not needing all members in the colony to all participate in several-stamped signature, and the persons who prove do not know that those members participate in signing, so cause it while disputing, it is unable to learn that those members participate in signing the file actually in a colony. For offer different application, have so-called traceable threshold of person who sign (Traceable signer) worth several proposition to sign art of composition [28, 29]. That is to say, the persons who prove can also know that those members participate in signing the file actually at the same time while verifying colony's stamped signature of threshold value.

Threshold proxy signature [16, 30] as its name is combining the

Group-oriented signature and designs the idea with proxy signature. So the signature method of threshold proxy signature that is a proxy signer who appoints a colony with many people of persons is signed by original signer. The number of members in this colony can produce legal proxy signature while exceeding threshold value. The verifier is unable to know that those members participate in signing.

1.3 About this Thesis

The rest of this paper is organized as follows. In Section 2, we discuss the related model and some security requirements for threshold proxy signatures. Section 3 briefly reviews Lamport proxy one-time signature scheme and Change's improving Lamport proxy one-time signature scheme. Section 4 briefly reviews threshold MAC. In Section 5, we expand Change's improving Lamport proxy one-time signature scheme and combinatorial object PHF into threshold proxy one-time proxy signature scheme. Section 6 analyzes the security of the proposed scheme. Finally, we conclude this article in Section 7.

二、Relative Model

2.1 Proxy signature

In an era of digitization, there are digital signatures often used to sign various kinds of electronic document. For making digital signature more valid and undeniable, people must use one self's secret key to sign, and it is verified by one's public key in order to determine the legitimacy of digital signature. When people are unavailable to sign the document by personal, but he have to sign out at that time; he can find an agent to help finish this action for him. This is called "proxy signature". Proxy signature was proposed in 1996 by Mambo, etc. [10]. It allows people to assign others to be his agent to sign documents on time. The document signed by proxy signer is effective as original signer. Generally speaking, there are two characteristics of proxy signature: unable to forge and able to verify. It is for can't forge, another for person who can verify. The former characteristic means that only original singer or proxy signer can produce an effective proxy signature. The latter one means signature is errorless when the proxy signature is proved to be right. Proxy signatures have the following characteristics:

1. Distinguishability: Agent's stamped signature can be distinguished from primitive stamped signature of owner.
2. Unforgeability: Only original signer and proxy signer appointed by original signer can produce effective proxy signature. And

no one could forge proxy signature.

3. Verifiability : From the proxy signature, the verifier can believe original signer also confirm to sign the document °
4. Identifiably : From the proxy signature, the original signer could identify who the proxy signer is °
5. Undeniability : Proxy signer can not deny he used to sign the document personally °

2.2 Type of proxy Signature

Now, the original signer warrant proxy signer to sign the document, there are three types of delegation :

1. Full delegation : The original signer hands over its secret key to the proxy signer with secretly method. With the secret key, the proxy signer can sign the document. However, the full delegation will destroy privacy of original signer. The proxy signer may make a random sign to the document, because of having the secret key from original signer. Therefore, the signature is unable to be distinguished signed by original signer from signed by proxy signer. If the proxy signer upsets the signature on purpose, he made a random sign to the document. The original signer will be unable to prove by who is stamped signature place sign.
2. Partial delegation [10, 11] : Based on the shortcoming of the full delegation, the original signer authorized signing power to the proxy signer. Utilizing the secret key of original signer to calculate out the proxy key instead of directly conveying the

original key to the proxy signer, and then convey this proxy key to the proxy signer. Proxy signer can sign document and produces proxy signature by proxy key. It is different from the signature produced by original secret key, so the verifier can distinguish signature from proxy signature. In addition, this kind of delegation way can be subdivided into two kinds in accordance with "proxy of protection". First, proxy signature cannot protect the proxy signer, namely except proxy signer can produces proxy signature, the original signer also possesses proxy key that can produce proxy signature. Therefore, the proxy signature can not be confirmed by who produced proxy signer or original signer. This is unfair as for proxy signer. If the original signer produce proxy signature, it regards proxy signature as sign from proxy signer, and proxy signer must be responsible for the signature. In order to solve this problem, that lets original singer and proxy signer have independent signature power. So another kind of mandate way is the "protection proxy signer". As original signer wants to authorize the proxy signer while signing power, at first, a number value produced according to original signer's secret key, and convey this number value to the proxy signer secretly, then proxy signer utilizes secret key by oneself and combine received number value to calculate again to get the derived proxy key. The new proxy key can produce a proxy signature. Due to proxy key is produced by the proxy signer, so only people with this secret key can produce the proxy signature. Therefore, except the proxy signer, no one (including original signer) may produce

proxy signature. This proxy signature produced by this way makes proxy signature can be traced, undeniable, and let the signer must be responsible for signed the document.

3. Delegation by warrant : The original signer products certifications of warrant to proxy signer. The certification of warrant use secret key from original signer. It not only delegate proxy signer the sign power of original signer, but also some specific declaration, such as the authority of delegation, type of signature etc. After proxy signer got certifications by warrant, he can use secret key of himself to sign proxy signature document, and it comprise the proxy signature. The verifier must check two steps: first, verify the correctness of signature. If it is correct, it checks certifications by warrant, and judge legal proxy signature of proxy signer.

4. Partial delegation and combine delegation by warrant: In 1997, Kim [8] et al. proposed a new proxy signature method that combines item 2 & 3. The original signer at first establishes the signature power for proxy signer, such as signature time limit, type of signature etc. Then original signer use secret key and signature power to calculate out proxy key to proxy signer. According to the message received, the proxy signer uses his owner secret key to calculate out the proxy signature key. Therefore, the proxy signer can use the proxy signature key to sign the document and product the proxy signature. The proxy signature includes the sign power of original signer. And it can be regarded as legal proxy signature while it totally tallies with signs power.

2.3 Threshold Proxy Signature

In [7, 8, 16 and 21], the authors provided not only various constructions for threshold proxy signature schemes, but also various security requirements. Hwang et al. summarized the following requirements for a (w, n) threshold proxy signature:

- **Secrecy.** No proxy signers can derive the original's private key from any information such as the shares of the proxy signing key, proxy signature etc. Even if all proxy signers collude together, they cannot get the original signer's private key.
- **Proxy protected.** Only the delegated proxy signer can generate partial proxy signature. It is infeasible for the original signer to forge partial signatures.
- **Unforgeability.** A valid proxy signature can only be cooperatively generated by w or more proxy signers. This means that if a signature has been generated by w or more proxy signers, $(w-1)$ or less proxy signers, or any third parties (not delegated proxy signers) can not forge the signature.
- **Nonrepudiation.** Any valid proxy signature must be generated by w or more proxy signers. That is, the scheme guarantees that proxy signers can not deny that they have signed the message and the original signer can not deny having delegated the power of signing messages to the proxy signers.
- **Time constraint.** The proxy signing keys can be used only during the appointed period. Once they expired, those keys cannot be used to generate a valid signature.
- **Traceable signers.** For internal auditing purposes, the system is able

to identify these signers who actually sign the message on behalf of the proxy group.

Although the above requirements are derived from threshold proxy signature schemes based on public-key cryptography, they are also suitable for a threshold one-time proxy signature scheme (or simply TOTP signature) based on one-way functions. Thus, this paper will follow these security requirements given above.

2.4 A (w, n) Threshold Proxy One-Time Signatures

To our best knowledge, there is only one paper [1] about TOTP signature. Al-Ibrahim's (w, n) TOTP signature scheme includes a trust party TP and a group of n signers $P_i, i = 1, 2, \dots, n$, together with three phases: key generation and share distribution, signing, and verification. These three phases is roughly depicted as follows. In the first phase, the signers select randomly secret key $s_j, j = 1, 2, \dots, v$, and divide into n shares, $s_{i,j}$ where $i' = 1, 2, \dots, n$, by the threshold Shamir secret sharing method, and send securely to $P_{i'}$ where $i' = 1, 2, \dots, n$. Then, the signers compute $p_j = h(s_j)$, and send to TP. In signing phase, each signer $P_i, i = 1, 2, \dots, t$, encodes the message m based on 2 as $m = (j_1, j_2, \dots, j_r)$. Then, each signer P_i computes partial signature $(s_{i,j_1}, s_{i,j_2}, \dots, s_{i,j_r})$ and sends it to each other. Finally, the signers jointly compute the signature $(m, j_k, s_{j_k}), k = 1, 2, \dots, r$, using Lagrange interpolation, and send it to a verifier. In verification phase, the verifier waits until all $(s_{j_1}, s_{j_2}, \dots, s_{j_r})$ and fetches p_j from TP. Then, the verifier checks whether $p_{j_k} = h(s_{j_k})$ where $k = 1, 2, \dots, r$.

三、One-Time Signature Scheme

In this section, we briefly describe the necessary cryptographic schemes which are used in our construction of TOTP signatures.

3.1 Lamport One-Time Signature Scheme

One-time signature schemes were first proposed by Rabin [14] and Lamport [9] and based on the idea of committing public keys to secure keys using one-way functions. For more than 25 years, Lamport one-time signature schemes have been proposed and investigated by many researchers. Indeed, one-time signature schemes have found many interesting applications, including on-line/off-line signatures, digital signatures with forward security properties, broadcast authentication protocols and proxy signatures etc.

In recent years, one-time signature schemes have attracted more and more attention, as an attractive alternative to the traditional signature schemes based on public key cryptography. One of the main advantages of one-time signature schemes is their reliance on one-way functions that can be implemented using fast hash function. The resulting signatures are the order of magnitude faster than signatures based on public cryptography applying on the resource-constrained, small devices, such as cellular phones, pagers, smart cards etc. The other of advantage of such a scheme is that it is generally quite fast. However, the scheme tends unwieldy when used to authenticate multiple messages because additional data needs to be generated to both sign and verify each new message. By contrast, with conventional signature schemes like RSA [15],

the key pair can be used to authenticate multiple documents, which will face the threat of replay attacks.

In this section, we briefly review the Lamport one-time signature, which includes three algorithms: key generation, signature signing and verification. Suppose that $h: Y \rightarrow Z$ is a one-way hash function.

(Key generation)

It should do the following steps as below:

- (1) Select $2k$ elements $y_{i,j} \in Y$ at random with $1 \leq i \leq k$ and $j = 1, 0$ where k is the length of message based on 2.
- (2) Compute $z_{i,j} = h(y_{i,j})$ for all i, j .
- (3) The key K consists of the $2k$ y 's and $2k$ z 's. The private key SK box and the public key PK box are as follows:

$$SK = \begin{Bmatrix} y_{1,0} & y_{2,0} & \cdots & y_{k,0} \\ y_{1,1} & y_{2,1} & \cdots & y_{k,1} \end{Bmatrix} \quad PK = \begin{Bmatrix} z_{1,0} & z_{2,0} & \cdots & z_{k,0} \\ z_{1,1} & z_{2,1} & \cdots & z_{k,1} \end{Bmatrix}$$

(Signature)

To sign the k -bit message $m=m_1\dots m_k$, we should do the following steps:

- (1) The corresponding entries of the message $m_1\dots m_k$ are $y_{1,m_1}, \dots, y_{k,m_k}$.
- (2) We define the signature

$$\text{Sig}(m_1\dots m_k) = (y_{1,m_1}, \dots, y_{k,m_k})$$

- (3) We just select corresponding entries from the key box to create signature.

For example, we want to sign a message $m=10\dots 1$. The signature is

$$\text{sig}(m_1 \dots m_k) = \left\{ \begin{array}{ccc} y_{1,0} & [y_{2,0}] & \dots & y_{k,0} \\ [y_{1,1}] & y_{2,1} & \dots & [y_{k,1}] \end{array} \right\}$$

$$= (Y_{1,1} \ Y_{2,0} \ \dots \ Y_{k,1})$$

On message $m_1 \dots m_k$.

(Verification)

To verify signature $(Y_{1,1} \ Y_{2,0} \ \dots \ Y_{k,1})$ on message $m_1 \dots m_k$, we check if

$$h(m_i) = y_{i,m_i} \text{ for } 1 \leq i \leq k \text{ holds.}$$

If it holds accept the signature, or reject it.

$$f(y_{1,1}, y_{2,0}, \dots, y_{k,1}) = \left\{ \begin{array}{ccc} z_{1,0} & [z_{2,0}] & \dots & z_{k,0} \\ [z_{1,1}] & z_{2,1} & \dots & [z_{k,1}] \end{array} \right\}$$

3.2 Improving Lamport one-time signature scheme by Chang [4]

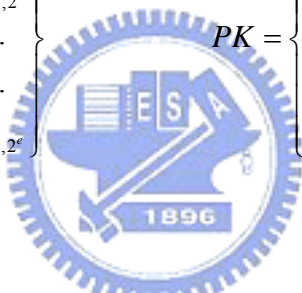
In [4], the authors propose a new scheme to improve the size of Lamport one-time signature. The Lamport one-time scheme requires a large amount of space for storage of authentic information if a large number of messages are signed. So in [4] improve the Lamport one-time signature on the amount of storage space for public keys and signed message saving storage space and propose an efficient scheme to sign a long message. The new scheme includes three algorithms: key generation, signature and verification. It is described as follows:

(Key generation)

It should do the following steps as below:

- (1) Select a number e and set $v=2^{e+1}$
- (2) Based on v , encode message $m = (m_1, \dots, m_l)_v$, where l is length of message after encoding.
- (3) For each column i , randomly select $e + 1$ elements $\in Y$ with suffix by power of 2 as $y_{i,2^0}, y_{i,2^1}, \dots, y_{i,2^e}$, where $1 \leq i \leq l$ and Y is mentioned in section 3.1.
- (4) Compute the corresponding public key box by using hash function.

Thus private key SK box and the public key PK box are shown as follows:

$$SK = \begin{Bmatrix} y_{1,2^0} & y_{2,2^0} & \dots & y_{l,2^0} \\ y_{1,2^1} & y_{2,2^1} & \dots & y_{l,2^1} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ y_{1,2^e} & y_{2,2^e} & \dots & y_{l,2^e} \end{Bmatrix} \quad PK = \begin{Bmatrix} z_{1,2^0} & z_{2,2^0} & \dots & z_{l,2^0} \\ z_{1,2^1} & z_{2,2^1} & \dots & z_{l,2^1} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ z_{1,2^e} & z_{2,2^e} & \dots & z_{l,2^e} \end{Bmatrix}$$


(Signature)

To sign the message m , we should do the following steps:

- (1) Encode the message based on v as $m = (m_1, \dots, m_l)_v$.
- (2) Decode each digit m_i based on 2 as $m_i = (u_1, u_2, \dots, u_e)_2$.
- (3) According to $m_i = (u_1, u_2, \dots, u_e)_2$, select corresponding entries of Y at column i . For example, if $u_j = 1$ then select $y_{i,2^j}$, else discard it, where $1 \leq j < e$.

Thus, the signature of message m is the selected items in the private key box (a_1, a_2, \dots, a_n) .

To sign on message m for example, we select $e=3$ and set $v=2^{(3+1)}=16$ and encode $M=(6B\dots1)_{16}$; then we decode $6=(0110)_2$, $B=(1011)_2$ and

1=(0001)2, etc. and select the corresponding entries in the private key box $(a_1, a_2, \dots, a_n) = (y_{1,2^1}, y_{1,2^2}, y_{2,2^0}, y_{2,2^1}, y_{2,2^3}, \dots, y_{1,2^0})$ as a signature

$$\text{sign}(6B\dots1) = \left\{ \begin{array}{cccc} y_{1,2^0} & [y_{2,2^0}] & \dots & [y_{l,2^0}] \\ [y_{1,2^1}] & [y_{2,2^1}] & \dots & y_{l,2^1} \\ [y_{1,2^2}] & y_{2,2^2} & \dots & y_{l,2^2} \\ y_{1,2^3} & [y_{2,2^3}] & \dots & y_{l,2^3} \end{array} \right\}$$

$$= (y_{1,2^1}, y_{1,2^2}, y_{2,2^0}, y_{2,2^1}, y_{2,2^3}, \dots, y_{1,2^0}) .$$

(Verification)

To verify the signature (a_1, a_2, \dots, a_n) on message m . We hash each elements of the signature (a_1, a_2, \dots, a_n) and check whether equal to corresponding entries in public key PK box.

For example, we check the signature $(y_{1,2^1}, y_{1,2^2}, y_{2,2^0}, y_{2,2^1}, y_{2,2^3}, \dots, y_{1,2^0})$ on message $M=(6B\dots1)_{16}$; select the corresponding entries in the public key box and check that the pre-image of the selected entries are the signature as the follows:

$$f(y_{1,2^1}, y_{1,2^2}, y_{2,2^0}, y_{2,2^1}, y_{2,2^3}, \dots, y_{1,2^0}) = \left\{ \begin{array}{cccc} z_{1,2^0} & [z_{2,2^0}] & \dots & [z_{l,2^0}] \\ [z_{1,2^1}] & [z_{2,2^1}] & \dots & z_{l,2^1} \\ [z_{1,2^2}] & z_{2,2^2} & \dots & z_{l,2^2} \\ z_{1,2^3} & [z_{2,2^3}] & \dots & z_{l,2^3} \end{array} \right\}$$

In this improving scheme, a message to be signed is based on the power of 2. The message is divided into l digits. Each digit of the message is signed individually. The signature is the corresponding entries of private key box with 1's binary in each digit encoded

by based 2. The verification is checking whether each items of signature is the pre-image of the corresponding public key entries.



四、Threshold MAC

4.1 Perfect Hash Families (PHF) and Cover Free Family (CFF)

We review the definition of PHF $(N; n, m, w)$ and (n, m, w) -CFF as follows.

Definition 2 [2] Let n, m and w be integers such that $n \geq m \geq w \geq 2$. Let V be a set with $|V| = n$ and let F be a set with $|F| = m$. Let A be an $N \times n$ array with entries in F . A set X of columns of A is *separated* by the i th row of A if the i th components of columns in X are all distinct. An (n, m, w) -perfect hash family is an $N \times n$ array A with entries in the set F if for every subset X of the columns of A with $|X| = w$ there exists at least one row that separates X . Let $\text{PHF}(N; n, m, w)$ denote an (n, m, w) -perfect hash family which has N rows.

Definition 3. [6] Let (X, F) be a set system with $X = \{x_1, x_2, \dots, x_m\}$ and $F = \{B_i \subseteq X \mid i = 1, 2, \dots, n\}$. We call (X, F) be an (n, m, w) -CFF (or (n, m, w) -CFF for short) if $B_i \not\subseteq B_{j_1} \cup B_{j_2} \cup \dots \cup B_{j_w}$ for all $B_{j_1}, B_{j_2}, \dots, B_{j_w} \in F$, where $i \notin \{j_1, j_2, \dots, j_w\}$.

4.2 Threshold CFF MAC scheme [12]

In [12], the authors presented a threshold MAC based on CFF as follows. Suppose a set system (X, B) is an (n, m, w) -CFF and $h: K \times M \rightarrow H$ is a secure MAC function. A (w, n) threshold MAC works as follows.

(Key Generation)

The receiver chooses m keys from K , $X = \{ \}$, and partitions X into n k -subsets B_1, B_2, \dots, B_n such that (X, B) is an (n, m, w) -CFF. Then, the receiver securely sends the k -subset B_i to sender P_i , for $i = 1, 2, \dots, n$.

(MAC Generation)

Suppose w senders $A = \{ P_{i_1}, P_{i_2}, \dots, P_{i_w} \}$ want to generate a MAC on message m . They compute $I = \{ j | k_j \in B_{j_1} \cup B_{j_2} \cup \dots \cup B_{j_w} \}$, Then the senders in A jointly calculate $\sigma = \bigoplus_{j \in I} h(k_j, m)$ and send (σ, m, I) to the receiver.

**(Verification)**

The receiver computes $\sigma' = \bigoplus_{j \in I} h(k_j, m)$ when the receiver receives (σ, m, I) . Then, the receiver checks whether $\sigma' = \sigma$.

五、Proposed Threshold One-Time Proxy Signature Scheme

In this chapter, we propose a new (w, n) TOTP signature scheme that combines the Change's improving Lamport one-time proxy signature scheme [4] and combinatorial object PHF. Therefore, the new schemes have advantages of Change's scheme that is improve the Lamport one-time signature on the amount of storage space for public keys and signed message saving storage space and propose an efficient scheme to sign a long message. Besides, we have enhanced more character of security that is really to protect the proxy signer and provides non-repudiation services to all parties involved.

5.1 A new (w, n) Threshold One-Time Proxy Signature

In this section, we introduce an efficient and securely scheme for threshold one-time proxy signature. There are three entities: an original signer, proxy signers, and a trust party (or simply TP) in the scheme and it works as follows.

5.2 Key Generation

(Key Generation)

Given an array A which is PHF $(N; n, m, w)$ and a hash function h with three inputs, the algorithm consists of the following three steps.

(1) We select a number e and set $v = 2^{e+1}$. Based on v , encode message

$m = (w_1, w_2, \dots, w_k)_v$, where k is length of message after encoding. The original signer generates the private keys, denoted by Q , of the Change's improving Lamport one-time proxy signature scheme as follows.

$$Q = \begin{Bmatrix} q_{1,2^0} & q_{2,2^0} & \dots & q_{k,2^0} \\ q_{1,2^1} & q_{2,2^1} & \dots & q_{k,2^1} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ q_{1,2^e} & q_{2,2^e} & \dots & q_{k,2^e} \end{Bmatrix}$$

First, the original signer computes

$$P = \{h(i, j, q) \mid \forall q \in Q, i = 1, \dots, N, j = 1, \dots, m\}$$

, and arranges each $h(i, j, q) \in P$, where i is the row index and j is the entry A_{ij} of the array A , as following array.

$$B_q = \begin{pmatrix} h(1, A_{11}, q) & h(1, A_{12}, q) & \dots & h(1, A_{1n}, q) \\ h(2, A_{21}, q) & h(2, A_{22}, q) & \dots & h(2, A_{2n}, q) \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ h(N, A_{N1}, q) & h(N, A_{N2}, q) & \dots & h(N, A_{Nn}, q) \end{pmatrix}$$

Let B_l be the l th column of B_q . Suppose that n proxy signers have been assigned a unique index between 1 and n . Then, the original signer securely sends each column (l, B_l) to each proxy signer l as proxy signing keys, $l = 1, \dots, n$.

Next, the original signer computes the *phase I public keys* for q

$$P_q^l = \{P_l^q \mid \forall q \in Q, l = 1, 2, \dots, n\}, \text{ where } P_l^q = h(\bigoplus_{h(i,j,q) \in B_l} h(i, j, q))$$

, and sends the phase I public keys to TP.

Finally, the original signer generates the *phase II public keys* for q as follows. Let \overline{B}_w be the union of any w columns in B_q . The original signer computes

$$P_q^H = \{H^2(\overline{B}_w, q) \mid \forall q \in Q, \overline{B}_w = B_{j_1} \cup B_{j_2} \cup \dots \cup B_{j_w}, \forall \{j_1, \dots, j_w\} \subseteq \{1, \dots, n\}\}$$

, where $H^2(\overline{B}_w, q) = h(\bigoplus_{h(i,j,q) \in \overline{B}_w} h(i, j, q))$ and sends $(j_1, j_2, \dots, j_w, H^2(\overline{B}_w, q))$ to TP.

(2) When receives B_j , proxy signer j computes $\bigoplus_{h(i,j,q) \in B_j} h(i, j, q)$ and applies h to it. Then, proxy signer j checks the result with corresponding components of the phase I public keys. If the validation goes through, proxy signer j selects a random number K_j as private key and computes the secure identifier $ID_j = h(K_j)$. Then, proxy signer j sends ID_j to TP.



(3) The TP generates final public keys from phase II public keys and secure identifier ID_j , $j = 1, 2, \dots, n$, as follows.

The TP publishes the final public keys $(j_1, j_2, \dots, j_w, ID_{j_1}, ID_{j_2}, \dots, ID_{j_w}, H^2(\overline{B}_w, q))$ for every w -subset $\{B_{j_1}, B_{j_2}, \dots, B_{j_w}\}$ (corresponding to $\{ID_{j_1}, ID_{j_2}, \dots, ID_{j_w}\}$) and for all $q \in Q$.

Figure 5.1 is a simplified view of the scenario by the key generation. The components in this scenario are Requester, Original Signer, Trust Party and Proxy Signers.

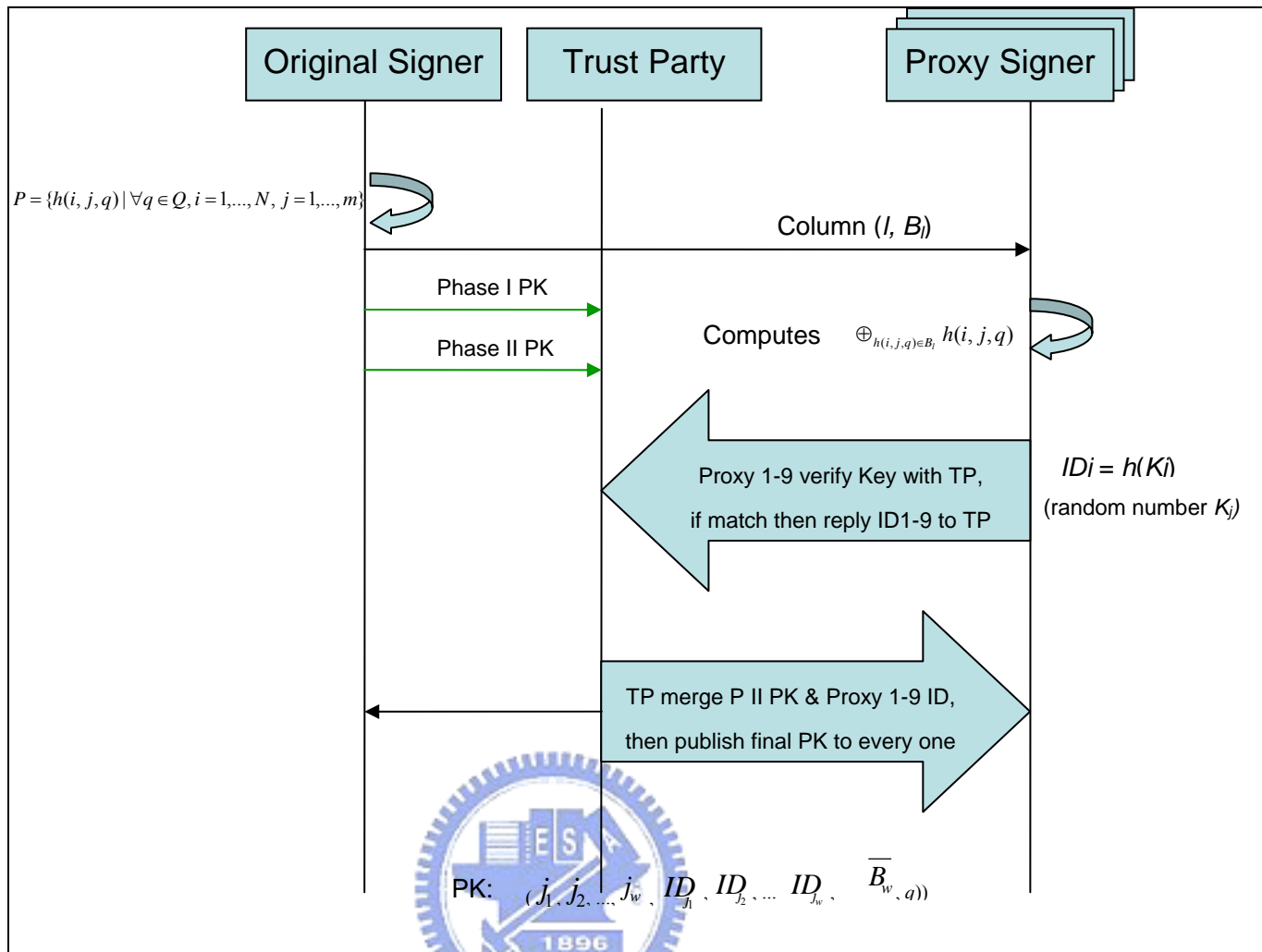


Figure 5.1. The Key Generation

5.3 Proxy Signature Generation

(Proxy Signature Generation)

Suppose that any w proxy signers $\{j_1, j_2, \dots, j_w\}$ want to sign a proxy signature on tk -bit message $m = (m_1, m_2, \dots, m_k)_2$. It works as following three steps.

(1) The proxy signers compute $r = h(m)$ and send to TP.

(2) The proxy signers construct the following array A' from A .

$$A' = \begin{pmatrix} (1, A_{11}) & (1, A_{12}) & \dots & (1, A_{1n}) \\ (2, A_{21}) & (2, A_{22}) & \dots & (2, A_{2n}) \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ (N, A_{N1}) & (N, A_{N2}) & \dots & (N, A_{Nn}) \end{pmatrix}$$

- (3) Each proxy signer j_i computes $S_{j_i} = A'_{j_i} \setminus \cup_{l=1}^i S_{j_{l-1}}$, where $S_{j_0} = \phi$ and $i=1, 2, \dots, w$.
- (4) For all $q \in \{q_{m_1}, q_{m_2}, \dots, q_{m_k}\}$, each proxy signer j_i computes partial signature $H_{j_i}(q) = (\bigoplus_{(i,j) \in S_{j_i}} h(i, j, q)) \oplus K_{j_i}$ where $l = 1, 2, \dots, w$.
- (5) Each proxy signers securely sends $(K_{j_i}, H_{j_i}(q))$ to the verifier.

Figure 5.2 is a simplified view of the scenario by the Proxy Signature generation. The components in this scenario are Trust Party, Proxy Signers and Verifier.



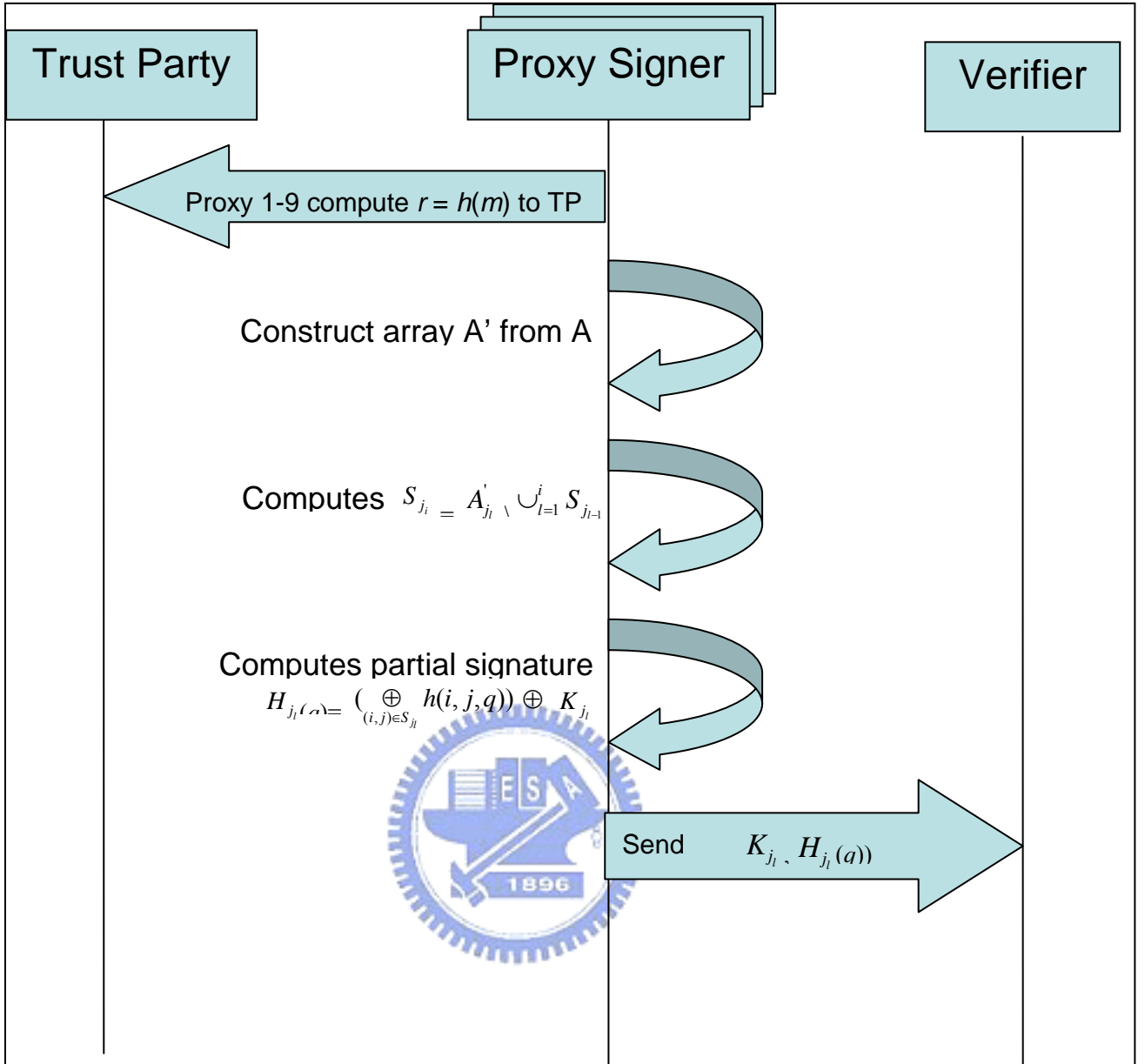


Figure 5.2. The Proxy Signature Generation

5.4 Proxy Signature Verification

(Proxy Signature Verification)

- (1) The verifier gets r from TP and checks whether $r = h(m)$.
- (2) The verifier checks whether $ID_{j_i} = h(K_{j_i})$, $i=1, 2, \dots, w$.
- (3) The verifier gets $H^2(\overline{B_w}, q_{m_i})$ from TP.
- (4) The verifier computes $H(\overline{B_w}, q_{m_i}) = \bigoplus_{i=1}^w (H_{j_i}(q_{m_i}) \oplus K_{j_i})$.

(5) The verifier applies h on $H(\overline{B}_w, q_{m_l})$ and checks whether the result equals to $H^2(\overline{B}_w, q_{m_l})$ where $\overline{B}_w = B_{j_1} \cup B_{j_2} \cup \dots \cup B_{j_w}$ and $l=1, 2, \dots, k$.

Figure 5.3 is a simplified view of the scenario by the Proxy Signature Verification. The components in this scenario are Trust Party and Verifier.

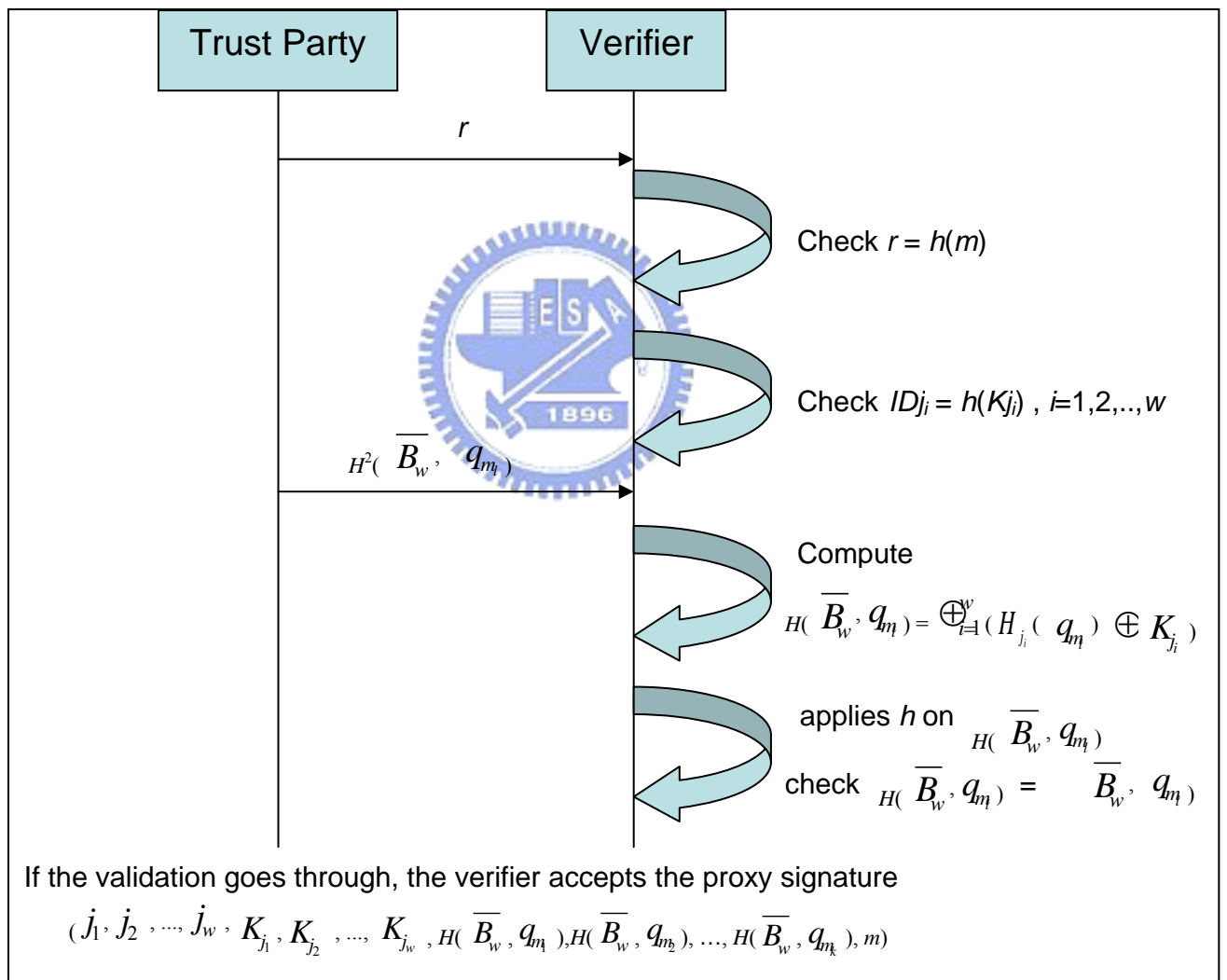


Figure 5.3. The Proxy Signature Verification

If the validation goes through, the verifier accepts the proxy signature $(j_1, j_2, \dots, j_w, K_{j_1}, K_{j_2}, \dots, K_{j_w}, H(\overline{B_w}, q_{m_1}), H(\overline{B_w}, q_{m_2}), \dots, H(\overline{B_w}, q_{m_k}), m)$ which is collaboratively generated by the signers $\{j_1, j_2, \dots, j_w\}$ on behalf of the proxy group $\{1, 2, \dots, n\}$.



六、System Analysis and Application

6.1 Security analysis

In this section, we examine the correctness and the security of this scheme.

6.1.1 Correctness

In our scheme, the proxy signer j_i computes $S_{j_i} = A'_{j_i} \setminus \cup_{l=1}^i S_{j_{l-1}}$ from the array A . Expanding the equation, we obtain $S_{j_1} = A'_{j_1}$, $S_{j_2} = A'_{j_2} \setminus S_{j_1}$, ..., and $S_{j_w} = A'_{j_w} \setminus \cup_{l=1}^w S_{j_l}$. It is easy to see that $\{h(i, j, q) \mid \forall (i, j) \in \cup_{l=1}^w S_{j_l}\} = \overline{B_w} = B_{j_1} \cup B_{j_2} \cup \dots \cup B_{j_w}$, since the entries of B_q and A' have the same position (i, j) . We have proved the following result.

Lemma 1. For a w -subset $\{j_1, j_2, \dots, j_w\}$ of proxy signers group $\{1, 2, \dots, n\}$, they can compute $\overline{B_w}$ together by the iteration equation $S_{j_i} = A'_{j_i} \setminus \cup_{l=1}^i S_{j_{l-1}}$, $i=1, 2, \dots, w$.

From lemma 1, we have $H(\overline{B_w}, q) = \bigoplus_{(i,j) \in \cup_l S_{j_l}} h(i, j, q)$. Therefore, the verifier can derive a validate proxy signature because that.

6.1.2 Security

(1) **Secrecy:** In our scheme, all proxy signing keys and public keys are derived from the original signer's private keys by hash function. So, the secrecy of private keys is reliance on one-way

hash functions without trapdoors.

(2) Proxy Protection: It is obvious that the original signer can not generate partial signature $H_{j_i}(q)$ because that each proxy private key K_j , where $j \in \{1, 2, \dots, n\}$, is unknown in our scheme. However, K_j is known for the original signer when proxy signature is published. Hence, the original signer can swallow the message and the signature, and then generate another one. To avoid such attack, the proxy signers register the hash of the message with TP and any verifier can check the message from TP in our scheme. Therefore, the original signer can not substitute for proxy signers.



(3) Unforgability: Consider the proxy signing key array

$$B_q = \begin{pmatrix} h(1, A_{11}, q) & h(1, A_{12}, q) & \dots & h(1, A_{1n}, q) \\ h(2, A_{21}, q) & h(2, A_{22}, q) & \dots & h(2, A_{2n}, q) \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ h(N, A_{N1}, q) & h(N, A_{N2}, q) & \dots & h(N, A_{Nn}, q) \end{pmatrix}$$

, which is generated from $P = \{h(i, j, q) | \forall q \in Q, i=1, \dots, N, j=1, \dots, m\}$ and array A which is a PHF($N; n, m, w$). Let B_l be the l th column of B_q , $l = 1, 2, \dots, n$. For any w columns $\overline{B_w} = \{B_{j_1}, B_{j_2}, \dots, B_{j_w}\}$, there exists at least one index i such that the i th component of all columns in $\overline{B_w}$ are all distinct since A is a PHF. It implies that $h(i, A_{ij_1}, q), h(i, A_{ij_2}, q), \dots, h(i, A_{ij_w}, q)$ are w distinct elements in $\overline{B_w}$.

So the union of any $w-1$ columns in B_q can not cover the remaining one. We have proved the following result.

Theorem 1. The set system (P, B_q) is (Nm, n, w) -CFF, where P and B_q as mention above.

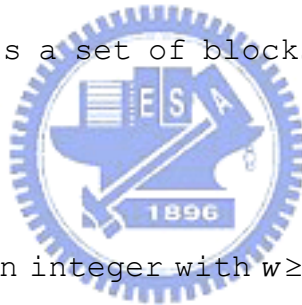
From theorem 1, a validate proxy signature cannot be generated by $(w-1)$ or less proxy signers. Therefore, the property of unforgeability is satisfied.

(4) Non-repudiation: From the proxy signature $(j_1, j_2, \dots, j_w, K_{j_1}, K_{j_2}, \dots, K_{j_w}, H(\overline{B_w}, q_{m_1}), H(\overline{B_w}, q_{m_2}), \dots, H(\overline{B_w}, q_{m_k}), m)$, the proxy signers cannot deny having signed the message since only the proxy signers have the private keys $K_{j_1}, K_{j_2}, \dots, K_{j_w}$. Because that $H(\overline{B_w}, q)$ is derived from the private key q , the original signer cannot deny having delegated the power of signing message to the proxy signers as well.

(5) Traceable signers: In our scheme, the system can identify the actual signers of a given threshold proxy signature $(j_1, j_2, \dots, j_w, K_{j_1}, K_{j_2}, \dots, K_{j_w}, H(\overline{B_w}, q_{m_1}), H(\overline{B_w}, q_{m_2}), \dots, H(\overline{B_w}, q_{m_k}), m)$ through $K_{j_1}, K_{j_2}, \dots, K_{j_w}$ and $H(\overline{B_w}, q_{m_1}), H(\overline{B_w}, q_{m_2}), \dots, H(\overline{B_w}, q_{m_k})$. It is easy to see that $(K_{j_1}, K_{j_2}, \dots, K_{j_w})$ is distinct for every different proxy signature. $H(\overline{B_w}, q)$ are also distinct for every

$\overline{B}_w \subseteq B_q$ as showing in the following theorems. Therefore, we can identify the actual signers.

In order to prove that $H(\overline{B}_w, q)$ are distinct for every $\overline{B}_w \subseteq B_q$, we will construct $H(\overline{B}_w, q)$ from a resolvable Balanced Incomplete Block Design (or simply BIBD). First, let us recall the definition of BIBD. A (v, b, k, λ) -BIBD is a set system (X, F) where X has v elements (or points) and F is a collection of b k -subsets (blocks) of X . Each point is contained r blocks, each block contains k points, and each pair of distinct points is contained in λ blocks. A BIBD is resolvable if there is a partition of its set of blocks F into parallel classes that is a set of blocks contains no point of the design more than once.



Theorem 2. [2] Let w be an integer with $w \geq 2$. If there is a resolvable (v, b, r, k, λ) -BIBD such that $w > \lambda \binom{w}{2}$, then there exists a PHF $(N; v, v/k, w)$ with $N = \lambda \binom{w}{2} + 1$.

Theorem 3. There is a construction for B_q such that the unions of any w columns in B_q are all distinct.

Proof. Given a resolvable $(v, b, 3, 1)$ -BIBD, by theorem 2, we can construct a PHF $(N; v, v/3, w)$ A , where $N = \binom{w}{2} + 1$, through the following fashion. It is easy to see that the BIBD has r parallel classes; each class has $v/3$ blocks. Therefore we can assign the points in

the same block an index between 1 and $v/3$ for every class. Hence, we can construct B_q from the PHF A . By theorem 1, we know that (B_q, P) is a CFF. For example [], given a $(9, 12, 4, 3, 1)$ -BIBD as follows.

class1	class2	class3	class4	assign
(1 2 3)	(1 4 7)	(1 5 9)	(1 6 8)	← 1
(4 5 6)	(2 5 8)	(2 6 7)	(2 4 9)	← 2
(7 8 9)	(3 6 9)	(3 4 8)	(3 5 7)	← 3

Following the assigned number, we can construct the following PHF.

	1	2	3	4	5	6	7	8	9
class1	1	1	1	2	2	2	3	3	3
class2	1	2	3	1	2	3	1	2	3
class3	1	2	3	3	1	2	2	3	1
class4	1	2	3	2	3	1	3	1	2

Consider the array A given above in which the entries can check whether the corresponding points belong to the same blocks when the entries have same values. Any w columns in A have $\binom{w}{2}$ pairs of distinct points that have the same values, since there is one block in a BIBD containing the pair. Suppose that there are \overline{B}_w and \overline{B}_w' such that $H(\overline{B}_w, q) = H(\overline{B}_w', q)$. Assume that \overline{B}_w and \overline{B}_w' have at least one distinct element. Then, there are at most $\binom{w}{2}$ different pairs of distinct points between them and the different pairs have same values and appear in different parallel classes. This implies that there is at least one pair of distinct points which belong to the different blocks. This contract is the definition of a BIBD. Therefore, there are no \overline{B}_w and \overline{B}_w' such that $H(\overline{B}_w, q) = H(\overline{B}_w', q)$.

6.2 Comparison

Our propose scheme is extend from Change's Improving Lamport One-Time signature scheme[4] and combinatorial object PHF. So Our propose scheme has same advantage of Change's Improving Lamport One-Time signature scheme too. And our propose scheme added advanced security. The comparison of Al-Ibrahim's (w, n) Threshold Proxy One-Time Signature Scheme and our proposed scheme is in Table 6.1.

We note that the new model of [1] is different from previous works. If we apply it to the original model, the TOTP signature scheme of [1] does not satisfy some requirements given above. We will discuss some weaknesses caused by their scheme using in the original model. First, the verifier cannot identify the actual proxy signer from the proxy signature. Therefore, the requirement "Traceable signers" is not satisfying. Second, the proxy signing key does not derive from the private key of the original signer. This means that the TP must guarantee that the original signer cannot refuse having delegated the power of signing messages to the proxy signers. Therefore, the TP is not merely to keep the public key and to prevent repeated signing. Third, there is no mechanism about preventing the signer from forging a valid proxy signature. Therefore, some important requirements such as "Nonrepudiation" and "Proxy protected" are not satisfied.

Table 6.1. The comparison of Al-Ibrahim's TOTP and our proposed scheme

Item	Al-Ibrahim's (w,n) TOTP	Proposed scheme
Secrecy	✓	✓
Proxy protected	✗	✓
Unforgeability	✗	✓
Non-repudiation	✗	✓
Time constraint	✓	✓
Traceable signers	✗	✓

6.3 Application of our proposed scheme

Numerous examples of group application include the stock exchange, collaborative tasks, and many other multicast applications.

According to above-mentioned theories, we can apply it to the countersign systems of general enterprises, suppose its institutional organization of company in order to there are 9 supreme executives of department - division chief under CEO, and the system administration (SA) service department establishes as trust center (Trust Party) among them; Section chief (Requester) who a product protected the department now, should work and need it to N days on business of China, he happens that their CEO also contacts with the business to other country on business during this section while applying for and going on business the form, and because this colleague's journey on business is urgent and needs dividing the expenses in the branch in advance badly at this moment,

it is not paid in advance until the CEO comes back after making comments and instructions that unable, then their CEO (Original singer) establishes agent's system to start in the easy system of countersigning before being on business, is it act as agent by the following 9 department affiliated supreme executive - division chief (Proxy signer) their signature operation to come, its rule, in order to so long as let 3 division chiefs signatures and agree to produce and act for signing the nuclear file of the CEO effectively, finish this go on business to signature operation form , let this staff can on business to is it finish the work to go prepaid expenses as scheduled smoothly. Its scenario is as figure 6-1 shows.

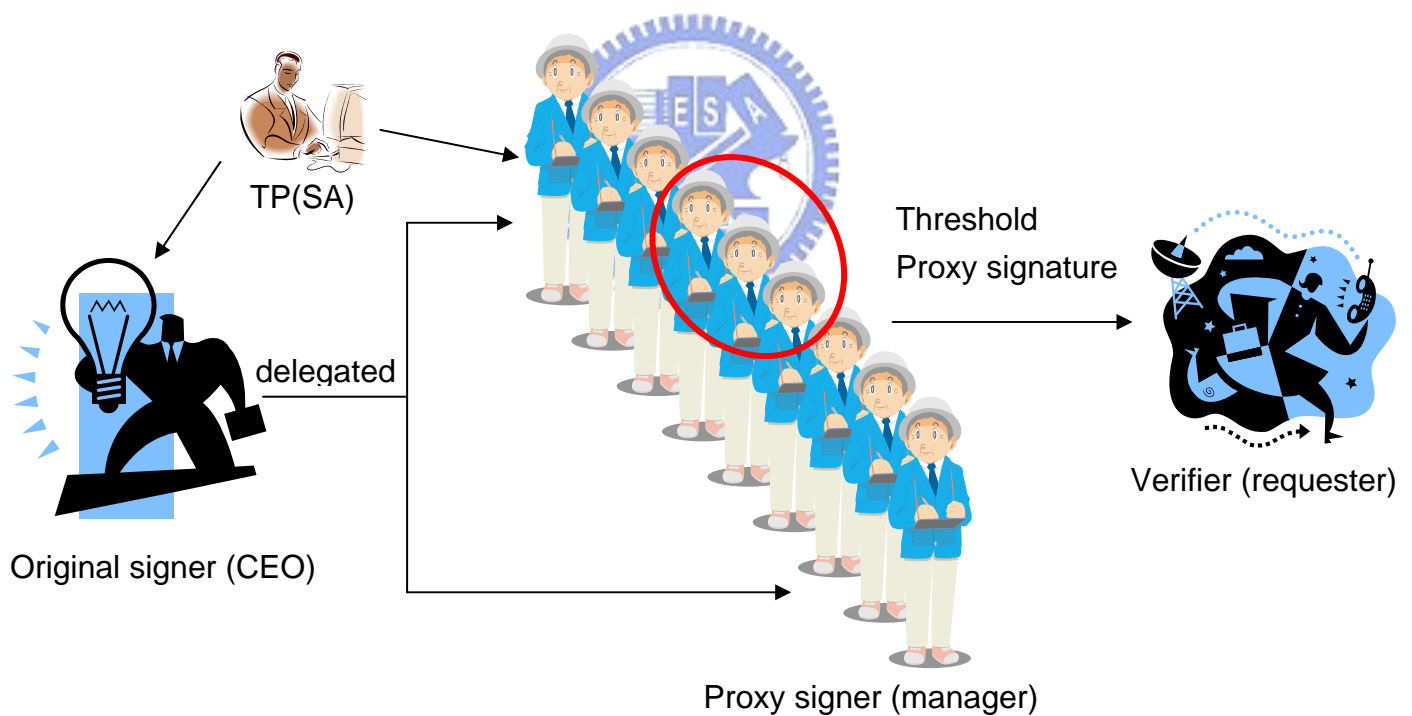


Figure 6.1. Examples of application for proposed method

七、Conclusion

7.1 Conclusion

Base on Chang's improving Lamport one-time signature scheme to execute threshold CFF MAC scheme. We present a new (w, n) threshold proxy one-time signature scheme that meets all the requirements of [7, 8, 16 and 21] under the original model. It protects proxy signer against repudiation of signature delegation of the original signer, repudiation of proxy signature generation of the proxy signer, and repudiation of receipt of the proxy signature of the signature recipient. Our scheme preserves the fast signature verification and low computation power of one-time signature, and so is suitable for various wireless applications. Furthermore, the proposed scheme inherits the character of Change's improving Lamport one-time proxy signature scheme as well.

7.2 Future Work

In this thesis, I just consider "one-time" proxy signature. This model is very secure method for protected proxy. If we can extend "multi-time" method to this model, it then will be better. So maybe research this issue in the future.

Furthermore, we proposed the scheme requires extra a large amount of space for storage of proxy signing key of PHF object array data. So maybe need to search other method or algorithm to saving storage space and propose a more efficient scheme for it.

參 考 文 獻

- [1] M. Al-Ibrahim and A. Cerny, "Proxy and Threshold One-Time Signatures," *In: Proc. of the 1th International Conference Applied Cryptography and Network Security (ACNS'03)*, LNCS 2846, pp. 123-136, Springer-Verlag, 2003.
- [2] S. R. Blackburn, "Combinatorics and Threshold Cryptology," *in Combinatorial Designs and their Applications (Chapman and Hall/CRC Research Notes in Mathematics)*, CRC Press, pp. 49-70, London, 1999.
- [3] C. C. Lindner and C. A. Rodger, "Design Theorey," *CRC Press*, Boca Raton, 1997.
- [4] M.-H. Chang, "Improving Lamport one-time signature scheme", *Applied Mathematics and Computation*, *ARTICLE IN PRESS*, AMC 9126, 2004.
- [5] T. ElGamal, "A Public-Key Cryptosystem and a signature Scheme Based on Discrete Logarithm," *IEEE Trans. Information Theory*, vol. 31, no. 4, pp. 469-472, 1985.
- [6] P. Erdős, P. Frankl, and Z. Furedi, "Families of finite sets in which no set is covered by the union of r others," *Israel Journal of Mathematics*, 51(1985), 79-89.
- [7] M.-S. Hwang, E. J.-L. Lu, and I.-C. Lin, "A Practical (t, n) Threshold Proxy Signature Scheme Based on the RSA

Cryptosystem," *IEEE Trans. Knowledge and Data Engineering*, vol. 15, no. 6, pp. 1552-1560, 2003.

[8] S. Kim, S. Park, D. Won, "Proxy signatures," revisited. *ICICS'97*, LNCS 1334, pp. 223-232, Springer, Berlin, 1997.

[9] L. Lamport, "Constructing digital signatures from a one-way function," *Technical report CSL-98*, SRI International, Palo Alto, 1979.

[10] M. Mambo, K. Usuda, E. Okamoto, "Proxy signatures: delegation of the power to sign messages," *IEICE Trans. Fundamentals E79-A* (9) (1996), pp. 1338-1354, 1996.

[11] M. Mambo, K. Usuda, E. Okamoto, "Proxy signatures for delegating signing operation," *Proc. 3rd ACM Conference on Computer and Communication Security*, ACM press, 1996, pp.48.

[12] K. Martin, J. Pieprzyk. R. Safavi-Naini, H. Wang, and P. Wild, "Threshold MACs," *Information Security and Cryptology - ICISC 2002*, LNCS 2587, pp. 237-252, Springer-Verlag, 2003.

[13] B.C. Neuman, "Proxy-based authorization and accounting for distributed systems," *Proc. 13th International Conference on Distributed Systems*, pp. 283-291, 1993.

[14] M. O. Rabin, "Digitalized signatures," *Foundations of*

- Secure Communication," *Academic Press*, pp. 155-168, 1979.
- [15] R.L. Rivest, A. Shamir, and L.M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Comm. ACM*, vol. 21, pp. 120-126, 1978.
- [16] H.-M. Sun, "An Efficient Nonrepudiable Threshold Proxy Signature Scheme with Known Signers," *Computer Comm.*, vol. 22, no. 8, pp. 717-722, 1999.
- [17] K. Usuda, M. Mambo, T. Uyematsu, E. Okamoto, "Proposal of an automatic signature scheme using a compiler," *IEICE Trans. Fundamentals E79-A* (1) (1996), pp. 94-101, 1996.
- [18] V. Varadharajan, P. Allen, and S. Black, "Analysis of the proxy problem in distributed systems," *Proc. 1991 IEEE Computer Society Symposium on Research in Security and Privacy*, 1991, pp. 255-275.
- [19] H. Wang and J. Pieprzyk "Efficient One-Time Proxy Signatures," *Advances in Cryptology- ASIACRYPT 2003* (ASIACRYPT'03), LNCS 2894, pp. 507-522, Springer-Verlag, 2003.
- [20] Guilin Wang, Feng Bao, Jianying Zhou, and Robert H. Deng, "Proxy Signature Scheme with Multiple Original Signers for Wireless E-Commerce Applications," *Proceedings of 60th IEEE Vehicular Technology Conference, Session 4.6: Wireless*

Sensor/Network Security, Los Angeles, California, September 2004, IEEE Vehicular Technology Society Press.

- [21] K. Zhang, "Threshold proxy signature schemes," *1997 Information Security Workshop*, Japan, September, 1997, pp. 191-197.
- [22] D. Chaum and E. Heyst, "Group signatures", *Advances in Cryptology: Eurocrypt'91*, pp. 257-265, 1992.
- [23] J. Camenisch and M. Michels, "Separability and efficiency for generic group signature schemes", *Advances in Cryptology: Crypto'99*, pp.106-121, 1999
- [24] S. Park, S. Kim and D. Won, "ID-based group signature", *Electronics Letters*, vol.33, no.19, pp.1616-1617, 1997.
- [25] Y. Desmedt and Y. Frankel, "Shared generation of authenticators and signatures", *Advances in Cryptology: Crypto'91*, pp.457-469, 1992.
- [26] L. Harn, "Group-oriented (t, n) threshold digital signature scheme and digital multisignature", *IEE Proc. Computers and Digital Techniques*, vol.141, no.5, pp.307-313, 1994.
- [27] K. Miyazaki and K. Takaragi, "A threshold digital signature scheme for a smart card based system", *IEICE Trans. Fundamentals*, Vol.E-84-A, no.1, pp.205-213, 2001.
- [28] C.T. Wang, C.H. Lin and C.C. Chang, "Threshold signature

- schemes with traceable signers in group communications",
Computer Communications, vol.21, no.8, pp.771-776, 1998.
- [29] Y.M. Tseng and J.K. Jan, "Attacks on threshold signature
scheme with traceable signers", Information Processing Letters,
vol.71, no.1, pp.1-4, 1999.
- [30] H.M. Sun, N.Y. Lee and T. Hwang, "Threshold proxy
signatures", IEE Proc. Comp. Digit. Tech., vol.146, no.5,
pp.259-263, 1999.

