

行政院國家科學委員會專題研究計畫 期中進度報告

大規模產生分散式金鑰之研究(1/2)

計畫類別：個別型計畫

計畫編號：NSC94-2213-E-009-110-

執行期間：94 年 08 月 01 日至 95 年 07 月 31 日

執行單位：國立交通大學資訊科學學系(所)

計畫主持人：曾文貴

計畫參與人員：林孝盈、周昆逸、劉易儒

報告類型：精簡報告

處理方式：本計畫可公開查詢

中 華 民 國 95 年 6 月 5 日

行政院國家科學委員會補助專題研究計畫 ☐ 成果報告
☒ 期中進度報告

大規模產生分散式金鑰之研究(1/2)

Distributed Key Generation in Large Scale

計畫類別：☒ 個別型計畫 ☐ 整合型計畫

計畫編號：NSC 94-2213-E-009-110-

執行期間：94 年 8 月 1 日至 95 年 7 月 31 日

計畫主持人：曾文貴 教授

計畫參與人員：林孝盈、周昆逸、劉易儒

成果報告類型(依經費核定清單規定繳交)：☒ 精簡報告 ☐ 完整報告

本成果報告包括以下應繳交之附件：

☐ 赴國外出差或研習心得報告一份

☐ 赴大陸地區出差或研習心得報告一份

☐ 出席國際學術會議心得報告及發表之論文各一份

☐ 國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、
列管計畫及下列情形者外，得立即公開查詢

☐ 涉及專利或其他智慧財產權，☐ 一年☐ 二年後可公開查詢

執行單位：國立交通大學 資訊科學系

中 華 民 國 95 年 5 月 31 日

中文摘要

一般的公開金鑰密碼系統中，使用者 U 擁有一對公開金鑰 PK_U 與密鑰 SK_U ，擁有 SK_U 就等於擁有使用者 U 的身份，就能代替 U 執行相對的密碼工作，因此如何保護 SK_U 是一個重要的研究課題。將 SK_U 以金鑰分享的技術分給 n 個參與者，不但是不錯的保護技術，也使系統具有容錯性。本計畫的主要目的是研究當 n 很大時如何有效率地產生金鑰持份。Canny 和 Sorkin 的基於 DL 的大規模產生金鑰持份方法相當創新，但也有一些缺點，我們希望研究改進其缺點，提出新的分散式金鑰持份產生方法及新的證明方法。除此之外，我們也將研究大規模產生基於 RSA 金鑰持份的方法，證明其安全性。最後我們希望將系統實出來，做實際的應用。

本年度（第一年度）我們已完成協定的設計，並作了一些分析，我們已經把成果寫成論文，即將投稿出去。

關鍵詞：分散式門金鑰產生、隨機技巧、安全證明。

英文摘要

In a public-key cryptosystem, each user U has a pair of public key PK_U and private key SK_U . As long as one possesses the private key SK_U , he/she owns the identity of U and can do every cryptographic work associated with SK_U . Therefore, it is very important to protect SK_U from leakage. One way of protection of SK_U is to share SK_U to n participants such that a number of them over a threshold can recover SK_U . Distributed key generation is that all participants generate their own secret shares by exchanging messages via open network. In the previously proposed schemes, due to real-world limitation and practical consideration, the number n of participants

cannot be very large. In this project we shall study the problem of distributed key generation for the case of large n . Canny and Sorkin has proposed a DL-based distributed key generation method for large-scale n . Their method is quite ingenious and elegant. They use the random walk technique to analyze the threshold of the proposed scheme. Nevertheless, there are still some disadvantages in their protocol. We hope to improve the protocol for more flexibility and better efficiency. Furthermore, we shall study RSA-based distributed key generation for large n and implement our proposed protocols.

一、計畫緣起及目的

一般的公開金鑰密碼系統中，使用者 U 擁有一對公開金鑰 PK_U 與密鑰 SK_U ，擁有 SK_U 就等於擁有使用者 U 的身份，可以通過對 U 的身份認證，就能代替 U 執行相對的密碼工作，例如，電子簽章等。因此如何保護 SK_U 是一個重要的研究課題。保護 SK_U 的方法很多，例如以下各方法及它們的憂缺點：

1. 將 SK_U 儲存於智慧卡中，讓 SK_U 永遠不離開智慧卡以免被竊取。好處是使用方便，安全性還算不錯；壞處是萬一失竊，則 U 的身份將被完全冒用，非常危險；又如果遺失且沒有備份或信託，就必須註銷，有一些管理上的不便。
2. 將 SK_U 以 (t,n) -金鑰分享的方式分散到 n 個安全參與者 P_i , $1 \leq i \leq n$ ，每一 P_i 得到一密鑰持份 (secret share) $SK_{U,i}$ 。將來如果有 t 個 P_i 's 參與就可以執行使用 SK_U 的工作。這樣做的好處是攻擊者必須攻破至少 t 個 P_i 's 才能得到 SK_U ，因此安全性得到一定程度的保障，如果有一些個別的 P_i (少於 $n-t$ 個) 無法工作，整個系統仍然可以運作，因此金鑰分享的作法也可以達到容錯 (fault tolerance) 的效果。缺點是這樣的系統通常比較複雜，達到

安全的門檻相當高， t 必須達到 $2n/3$ 之上；又因為系統長時間運作， P_i 擁有的 $SK_{U,i}$ 是固定的，攻擊者有從容的時間來攻擊各個 P_i ，成功的機會較大。

3. 利用預防式機制 (proactive) 增進金鑰分享機制的安全性。如前所述，因為系統存在的時間很長，攻擊者可以從容的攻擊 P_i 以得到各個 $SK_{U,i}$ ，預防式機制就是每隔一段時間 T ，所有的 P_i 就會用密碼方法交換訊息以更新 $SK_{U,i}$ ，新的 $SK_{U,i}$ 和舊的 $SK_{U,i}$ 是獨立的 (但是 SK_U 不變)，因此如果攻擊者無法在 T 時間內攻破 t 個以上的 P_i ，得到 t 個以上的金鑰持份以求得 SK_U ，系統就是安全的，因為在下一個 T 時段裡，舊的 $SK_{U,i}$ 就不再有用了。預防式金鑰更新機制基本上解決了金鑰分享機制裡金鑰持份是固定的缺點，但是門檻 t 過高的問題仍然存在。

產生金鑰持份又分為集中式和分散式產生 (Distributed Key Generation) 的模式。集中式產生金鑰持份模式是由一金鑰產生中心 (Key Distribution Center, KDC) 產生金鑰 SK_U 及金鑰持份 $SK_{U,i}$ ，然後透過安全的通訊管道將 $SK_{U,i}$ 傳給 P_i 。分散式金鑰持份產生模式是由所有的參與者共同產生，產生的方式是由所有的參與者透過公開的網路交換訊息，最後再由各個 P_i 計算出自己的金鑰持份 $SK_{U,i}$ ，實際的金鑰是由所有的 $SK_{U,i}$ 決定，但沒有任一參與者知道。集中式模式的安全較容易討論，因為假設 KDC 是可以信賴的；但分散式模式的安全性討論就非常困難，因為未必每一位參與者都是可信賴的。因此在分散式模式裡，必須假設一定比例的參與者是可信賴的。

先前的分散式金鑰產生系統裡，所有 n 個參與者都必須相互溝通才可以產生金鑰持份，為了實用性、網路穩定性及參與者特性 (例如，每一參與者只能聯絡一定人數等) 的考量， n 不能太大。本計畫的主要目的是

研究當 n 很大時如何分散式地產生金鑰持份，先前方法的主要缺點是任兩參與者間需要相互交換訊息，因此我可以考慮使每一參與者只和少數的參與者交換信息，如此可以使傳輸的負擔減輕，符合參與者的特性，也更具有容錯的能力。

二、研究成果

本年度(第一年度)的研究成果如下：

1. 我們完成論文『Efficient large scale distributed key generation against burst interruption』，請見附件，即將要投稿到國際會議去。

這篇論文主要的精神為利用機率方法來討論一個隨意的 bipartite graph，如果只有少數的邊，也具有很高的 order。我們將 $t \times n$ evaluation 矩陣 E 的每一列中任意 k 個 entry 設為非零的值，這相當是在一個 bipartite graph $G=(U,V,E)$ 中將少數邊加到 U 和 V 的節點上，形成一個 expander graph，再利用機率方法來討論這樣的 E 矩陣的 order 為 t 。最後我們將以實驗的方式來驗證我們的方法。

2. 我們也利用本計畫的經費完成論文，『Cheating prevention in visual cryptography』，已投稿到 IEEE Trans. Image Processing. 正在修訂 (minor revision) 中。

三、計畫成果自評

我們在這第一年度完成兩篇論文，一篇即將投稿出去，另一篇正在修訂中，很有希望被接受。以成果來看，我們達成了本計畫的目的。

參考文獻

1. Noga Alon, V. D. Milman: Eigenvalues, Expanders and Superconcentrators (Extended Abstract). FOCS 1984: 320-322, 1984.

2. R. Canetti, "Security and composition of multiparty cryptographic protocols", *Journal of Cryptology* 13(1), pp. 143-202, 2000.
3. R. Canetti, R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, "Adaptive security for threshold cryptosystems", *Proceedings of Advances in Cryptology -- Crypto '99, Lecture Notes in Computer Science* 1666, pp.98-115, Springer-Verlag, 1999.
4. R. Canetti, S. Halevi, A. Herzberg, "Maintaining authenticated communication in the presence of break-ins", *Journal of Cryptology* 13(1), pp.61-106, 2000.
5. R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin. Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. *Eurocrypt '99*, pages 295-310, 1999.
6. R. Canetti, E. Kushilevitz, R. Ostrovsky, A. Tosen, "Randomness versus fault-tolerance", *Journal of Cryptology* 13(1), pp.107-142, 2000.
7. J. Canny, S. Sorkin. Practical Large-scale Distributed Key Generation. In *Proceedings of Advances in Cryptology -- Eurocrypt '04*, pages 138-152, 2004.
8. Y. Desmedt, Y. Frankel, "Threshold cryptosystems", *Proceedings of Advances in Cryptology -- Crypto 89, Lecture Notes in Computer Science* 435, pp.307-315, Springer-Verlag, 1989.
9. P. Feldman. A Practical Scheme for Non-Interactive Verifiable Secret Sharing. In *Proc. 28th FOCS*, pages 427-437, 1987.
10. M. Franklin, R.N. Wright, "Secure communication in minimal connectivity models", *Journal of Cryptology* 13(1), pp.9-30, 2000.
11. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Robust Threshold DSS Signature". *Proceedings of Advances in Cryptology - Eurocrypt 96. Springer-Verlag Lecture Notes in Computer Science* 1070, pp. 354-371, 1996.
12. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, "Secure Distributed Key Generation for Discrete-Log Based Cryptosystems", *Proceedings of Advances in Cryptology -- Eurocrypt 99, Lecture Notes in Computer Science*, Springer Verlag, 1999.
13. David Gillman: A Chernoff bound for random walks on expander graphs. *FOCS 1993*: 680-691.
14. M. Hirt, U. Muarar, "Palyer simulation and general adversary structures in perfect multiparty computation", *Journal of Cryptology* 13(1), pp.31-60, 2000.
15. I. Ingemarsson, G.J. Simmons, "A protocol to set up shared secret schemes without the assistance of a mutually trusted party", *Proceedings of Advances in Cryptology -- Eurocrypt 90, Lecture Notes in Computer Science* 473, pp.266-282, Springer-Verlag, 1990.
16. T. Pedersen, "A threshold cryptosystem without a trusted party", *Proceedings of Advances in Cryptology -- Eurocrypt 91, Lecture Notes in Computer Science* 547, pp.522-526, Springer-Verlag, 1991.
17. T. Pedersen. Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. *Proceedings of Advances in Cryptology -- Crypto '91*, pages 129-140, 1991.
18. David Peleg, Eli Upfal: Constructing Disjoint Paths on Expander Graphs (Extended Abstract). *STOC 1987*: 264-273.
19. A. Shamir, "How to share a secret", *Communications of the ACM* 22(11), pp.612-613, 1979.

Efficient Large-Scale Distributed Key Generation Against Burst Interruption

Abstract

A distributed key generation system allows the key servers to share a secret key and then compute the corresponding public key. Canny and Sorkin [CS04] proposed a *probabilistic* threshold distributed key generation scheme that is suitable for the case that the number of key servers is large. The communication cost of their scheme is much less than that of previous schemes. Nevertheless, it is possible to improve their scheme in some aspects. In this paper we employ the randomness technique to cope with the problems encountered by their scheme. Our contribution is twofold. Firstly, our scheme is secure against a large cluster of dishonest key servers and the DoS attack. Secondly, its performance is better than theirs in many situations. We support this point by a series of simulation experiments.

Keywords: security protocols, applied cryptography, distributed key generation, key server.

1 Introduction

The security of a cryptographic scheme usually lies on protecting a secret key. One way to protect such a key is to distribute it to a set of key servers (players) such that each key server holds a share of the key. Key sharing not only enhances key protection, but also provides a robustness property for use of the key. For example, in a threshold key sharing system, a set of key servers over a threshold number can recover the secret.

A distributed key generation system allows the key servers to share a secret key and then compute the corresponding public key. In this paper we focus on discrete-logarithm-based threshold distributed key generation schemes, in which the secret key is x and the public key is $y = g^x \bmod p$. Almost all threshold distributed key generation schemes use secret sharing schemes as building blocks. Shamir [Sha79] proposed the first threshold secret sharing scheme based on polynomial interpolation. Feldman [Fel87] added verification of secret shares (verifiable secret sharing, VSS) to Shamir's scheme. Pedersen [Ped91a] further improved the scheme by making the secret shares unconditional secure.

Based on his verifiable secret sharing scheme, Pedersen [Ped91b] proposed a threshold distributed key generation scheme, which has most important properties that a threshold distributed key generation scheme should have. Gennaro et. al. [GJKR99] found that an adversary can bias the distribution of the secret key by a subtle maneuvering in previously proposed threshold distributed key generation schemes. They then gave a formal definition and proposed a secure scheme. Chu and Tzeng [CT02] further pointed out that dishonest key servers should not obtain valid key shares to avoid abused use. Canny and Sorkin [CS04] proposed a *probabilistic* threshold distributed key generation scheme that is

suitable for the case that the number n of involved key servers is large, say, in the level of hundreds or thousands. The main merit of their scheme is that the total number of communications between key servers is greatly reduced from $O(n^2)$ to $O(nl/\epsilon^2)$, where l and ϵ are security and robustness parameters, respectively.

Canny and Sorkin's method is quite clever. Nevertheless, it is possible to improve their scheme in some aspects. Firstly, since the arrangement of key servers is very regular, the scheme is vulnerable to a large cluster of dishonest key servers¹. Inferring from this, if the DoS attack occurs to block a cluster of honest key servers from connecting to Internet, the execution of the scheme would fail. Secondly, it needs a dealer not only to generate the evaluation matrix, but also to assign labels to the key servers.

In this paper we employ the randomness technique to cope with the problems encountered by Canny and Sorkin's scheme. We assign non-zero values to *random entries*, while Canny and Sorkin's scheme assigns non-zero values to fixed entries. Our contribution is twofold. Firstly, our scheme is secure against a large cluster of dishonest key servers and the DoS attack. Secondly, its performance is better than Canny and Sorkin's method in many situations. We support this point by a series of simulation experiments.

2 Preliminary

Let $p = 2q + 1$ be a large prime, say, of 2048-bit length, where q is also prime. Let G_q be the subgroup of quadratic residues in Z_p^* and g and h be generators of G_q . Hereafter, the operations used in exponents of g and h are over Z_q . Assume that there are n key servers and the threshold is t . A bold character is either a matrix, like \mathbf{E} , or a vector, like \mathbf{a}_i .

A probabilistic threshold distribution key generation (PTDKG) scheme consists of three stages: setup, key share establishment and secret key recovery. A PTDKG scheme should satisfy the following conditions.

Definition 1. An (α, β, δ) -PTDKG scheme should satisfy the following conditions:

- C1. The shares of any subset of key servers define the same secret key x , or not at all.*
- C2. Any subset of βn key servers can recover the secret key x with probability $1 - \delta$ at least.*
- C3. The secret key x is uniformly distributed in Z_q .*
- S1. Any adversary who controls probabilistically up to αn key servers cannot get any information about the secret key x except the information computed from the public key y directly.*

In condition S1, we assume that the adversary probabilistically selects a fraction α of key servers to control. Otherwise, it is not possible to reduce the communication cost under suitable parameters. To see this, if each key server U_i communicates with r key servers, α must be less than r/n , which is very small for $r = O(\log n)$. Otherwise, the adversary who controls those r key servers can get the information of the key server U_i . Note that although in Canny and Sorkin definition, the adversary is defined as to control

¹Stated in the paper, "any large cluster of dishonest players leaves part of the private key vulnerable."

an *arbitrary* fraction α of all key servers, it is indeed required to control a *probabilistic* fraction of all key servers. Otherwise, the α of their scheme should be $\frac{u}{n} = O(\log n/n)$, not $\frac{1}{f} - \epsilon$.

The flaw of Canny and Sorkin's analysis is that α is set as the fraction of $r/|Q_i|$. However, it should be the fraction of r/n . For details, please see Section 4.4 of [CS04].

2.1 Previous approaches

In a distributed key generation scheme based on Shamir's secret sharing scheme, each key server U_i runs a key sharing scheme to share its chosen secret \mathbf{a}_i to others key servers. We shall use the matrix representation to show the key sharing method. The operations are over Z_q . Let the *evaluation matrix* be

$$\mathbf{E} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 2^{t-1} & \cdots & n^{t-1} \end{bmatrix}.$$

Each key server U_i chooses a random t -dimensional vector

$$\mathbf{a}_i = [a_{i,1} \ a_{i,2} \ \cdots \ a_{i,t}].$$

Let

$$\mathbf{a}_i \mathbf{E} = \mathbf{s}_i = [s_{i,1} \ s_{i,2} \ \cdots \ s_{i,n}].$$

The key server U_i sends $s_{i,j}$ to the key server U_j , $1 \leq j \leq n$. Let $H \subseteq \{1, 2, \dots, n\}$ be the set of honest key servers after the key share establishment stage. The established secret key is $x = \sum_{i \in H} a_{i,0}$. We have

$$\left(\sum_{i \in H} \mathbf{a}_i \right) \mathbf{E} = \sum_{i \in H} \mathbf{s}_i.$$

Each key server U_j in H gets its key share $x_j = \sum_{i \in H} s_{i,j}$ of the secret $x = \sum_{i \in H} a_{i,0}$.

For $A \subset \{1, 2, \dots, n\}$, let \mathbf{E}^A be the matrix with columns restricted to A . In the key recovery stage, a set T of key servers from H can recover the secret x if and only if \mathbf{E}^T has the full rank, i.e., $\text{rank}(\mathbf{E}^T) = t$. We can solve x by selecting t independent columns $\mathbf{E}^{T'}$ from \mathbf{E}^T , $T' \subseteq T$, and compute

$$\sum_{i \in H} \mathbf{a}_i = \left(\sum_{i \in H} \mathbf{s}_i \right)^{T'} (\mathbf{E}^{T'})^{-1}. \quad (1)$$

Since any t rows of \mathbf{E} form a Vandermonde matrix, these rows are independent. Therefore, any t key servers can recover the secret x , which is the first entry of $\sum_{i \in H} \mathbf{a}_i$. Any set of less than t key servers cannot compute the secret key x . Thus, the above defines a $(\frac{t-1}{n}, \frac{t}{n}, 0)$ -PTDKG scheme.

One disadvantage of the above method is that each key server U_i has to communicate with each other key server. The total number of communications between the key servers is $O(n^2)$, which shall entail heavy network overhead when n is large.

Distributed key generation schemes based on Feldman's and Pedersen's verifiable secret sharing schemes are similar except that the received shares of each key server are verifiable [Fel87, Ped91b].

2.2 Canny and Sorkin's approach

The idea of Canny and Sorkin to reduce the communication cost is to make \mathbf{s}_i very sparse. For a zero entry $s_{i,j}$, the key server U_i need not send $s_{i,j}$ to the key server U_j . By this, the communication cost from U_i to U_j is saved. If \mathbf{s}_i is very sparse, the communication cost from U_i to other key servers U_j is much reduced.

Let \mathbf{E} be a $t \times n$ -dimensional evaluation matrix with a band of non-zero entries as follows, where \star means a random number in Z_q , which is non-zero with probability almost 1:

$$\mathbf{E} = \begin{bmatrix} \star & \star & \star & \star & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \star & \star & \star & \star & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \star & \star & \star & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \star & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \star & \star & \star \end{bmatrix}$$

Let l be the width of the band and f be the offset of the band between two consecutive rows. For example, the above band matrix has $l = 4$ and $f = 2$. In the scheme, a dealer chooses \mathbf{E} and publishes it. Each key sever U_i chooses a t -dimensional block vector

$$\mathbf{a}_i = [0 \quad \cdots \quad 0 \quad a_{i,j} \quad a_{i,j+1} \quad \cdots \quad a_{i,j+k-1} \quad 0 \quad \cdots \quad 0]$$

where j is a pre-determined index and k is the block width. The vector $\mathbf{s}_i = \mathbf{a}_i \mathbf{E}$ has only $(k-1)f + l$ non-zero entries. The key server U_i need send non-zero share $s_{i,j}$ to the key servers U_j . With fixed t and n , we can make $(k-1)f + l$ small by tuning parameters k , l and f .

Canny and Sorkin's PTDKG (called CS-PTDKG hereafter) scheme is $(\frac{1}{f} - \epsilon, \frac{1}{f} + \epsilon, \delta)$, for some small ϵ and δ , $0 < \epsilon, \delta < 1$. Overall, their method needs $n((k-1)f + l)$ node-to-node communications, while most previous methods need $n(n-1)$ node-to-node communications. They suggest that $l = O(\log n)$ and $k = l/(2\epsilon^2)$. This saves quite a lot of communications between key servers overall when n is large.

Though the idea of saving communication costs is quite clever, their method has some disadvantages:

1. The arrangement of the evaluation matrix and key servers is very regular. It is vulnerable to a large cluster of dishonest key servers and to the DoS attack, which can attack routers and cause a larger cluster of key servers out of connecting to the Internet.
2. The dealer not only generates the evaluation matrix, but also assigns labels to the key servers. Since the number of key servers is large and the key servers are distributed, the task of assigning labels to key server is not practical in some situations.

3 Our construction

In Canny and Sorkin's method, \mathbf{E} and \mathbf{a}_i is very regular and this regularity makes the system vulnerable to burst interruption. We employ the randomness technique to cope

with the problems. We choose \mathbf{E} and \mathbf{a}_i randomly such that it is more robust against burst interruption.

For each row of \mathbf{E} , we randomly choose l entries and assign random values in Z_q to them. Each key server U_i randomly chooses k entries of \mathbf{a}_i and assigns random values in Z_q to them. We see that $\mathbf{s}_i = \mathbf{a}_i \mathbf{E}$ has about kl non-zero entries. Although the number of non-zero entries is more than $(k-1)f + l$ in Canny and Sorkin's method if k and l are the same. We show that our system needs smaller k and l to achieve the same level of robustness in simulation.

Before presenting our method, we need to discuss some theoretical problems concerning the feasibility of our construction. We consider \mathbf{E} as the matrix representation of a bipartite graph $G = (U, V, E)$, where U denotes the set of the vertices in rows and V denotes the set of the vertices in columns. Thus, $|U| = t$ and $|V| = n$. The bipartite graph $G = (U, V, E)$ is left l -regular since every vertex $u \in U$ has degree l . We consider simple bipartite graphs (no self-loops and multi-edges).

The property of the full rank of \mathbf{E} is related to *perfect matching* from U to V of G , $|U| \leq |V|$. Assume that $M \subseteq E$ is a perfect matching from U to V , that is, $|M| = t$ and all vertices of the edges in M are distinct. We can use the matching edge $(u, v) \in M$ as the pivot entry (u, v) of \mathbf{E} to eliminate non-zero entries in column v . Furthermore, since the values in non-zero entries are randomly selected from a very large set Z_q , it is very unlikely that the elimination process by a pivot would cause another pivot to be zero. Therefore, the t columns v 's of E are independent, $(u, v) \in M$. We would say that \mathbf{E} has the full rank t if and only if G has a perfect matching from U to V . The criteria for a bipartite graph to have a perfect matching is known as Hall's lemma.

Lemma 1 (Hall). *A bipartite graph $G = (U, V, E)$ has a perfect matching from U to V if and only if for every subset $S \subseteq U$, $|\Gamma(S)| \geq |S|$, where $\Gamma(S)$ is the set of S 's neighbor vertices in V .*

We show that the probability that a random left l -regular bipartite graph has a perfect matching is close to 1.

Theorem 1. *For appropriate positive integers j, t, l and n such that, for $3 \leq j \leq t$,*

$$\frac{j(j-1)}{(t-j+1)(n-j+2)} \left(\frac{j-2}{j-1}\right)^{(j-1)l} \left(\frac{n}{j-1}\right)^l \geq 1.$$

The probability that a random left l -regular bipartite graph $G = (U, V, E)$ has a perfect matching is $1 - \frac{t^3}{2} \left(\frac{1}{n}\right)^{2l-1}$ at least, where $|U| = t$, $|V| = n$ and $t \leq n$.

Proof. We compute the probability that the condition in Hall's lemma is not satisfied. For a subset $S \subseteq U$ of j vertices and a subset $T \subseteq V$ of $j-1$ vertices, the probability that all edges from S hit into the set T is

$$\left(\frac{j-1}{n}\right)^{jl}.$$

The probability that there is a subset $S \subseteq U$ of j vertices whose edges hit within a subset of fewer than j vertices of V is at most

$$p_j = \binom{t}{j} \binom{n}{j-1} \left(\frac{j-1}{n}\right)^{jl}.$$

Since, for $3 \leq j \leq t$,

$$\frac{p_{j-1}}{p_j} = \frac{j(j-1)}{(t-j+1)(n-j+2)} \left(\frac{j-2}{j-1}\right)^{(j-1)l} \left(\frac{n}{j-1}\right)^l \geq 1,$$

the probability that a left l -regular random bipartite graph does not satisfy Hall's lemma is at most

$$\sum_{j=2}^t p_j \leq (t-1)p_2 = \frac{t(t-1)^2}{2} \left(\frac{1}{n}\right)^{2l-1} < \frac{t^3}{2} \left(\frac{1}{n}\right)^{2l-1}.$$

Thus, the theorem holds. \square

We notice that the probability can be made arbitrarily small even with rather small l since l is in the exponent of $\frac{1}{n}$ and n is large.

Now, we consider the recoverability of the secret key after the key share establishment stage. After dishonest and unavailable key servers are discarded, a set H of honest key servers is formed. The secret key is computed from the contribution of the key servers in H . The key servers in H can recover the secret key if and only if \mathbf{E}^H has the full rank t , as explained in Equation (1). Assume that H is randomly selected from $\{1, 2, \dots, n\}$. The probability that \mathbf{E}^H has the full rank depends on the size of H . We show that as long as H is not too small, the probability is close to 1. Let V' (that is, the set H of honest key servers) be a subset of V by randomly deleting m vertices from V . Then, $(U, V', E|_{U \cup V'})$ has a perfect matching from U to V' with an overwhelming probability with proper parameters, where $E|_{U \cup V'}$ is the set of edges incident to vertices in $U \cup V'$.

Theorem 2. *For appropriate positive integers j, t, l and n such that, for $3 \leq j \leq t$,*

$$\frac{j(j-1)}{(t-j+1)(n-m-j+2)} \left(\frac{j-2+m}{j-1+m}\right)^{(j-1)l} \left(\frac{n}{j-1+m}\right)^d \geq 1.$$

Let $G = (U, V, E)$ be a left l -regular random bipartite graph. After deleting random m vertices from V , the probability that the remained bipartite graph has a perfect matching is $1 - (n-m) \frac{t(t-1)^2}{2} \left(\frac{m+1}{n}\right)^{2l}$ at least, where $|U| = t$, $|V| = n$ and $t \leq n$.

Proof. Let V' be the subset of V after deleting m vertices, where $|V'| = n' = n - m$. An edge from a vertex in U that hits a vertex in $V - V'$ makes no contribution to Hall's lemma. For a subset $S \subseteq U$ of j vertices and $T \subseteq V'$ of $j-1$ vertices, the probability that Hall's lemma does not hold on S to T is

$$\left(\frac{j-1+m}{n}\right)^{jl}.$$

Thus, the probability that there is a subset of $S \subseteq U$ of j vertices whose edges hit a subset of fewer than $j-1$ vertices in V' or $V - V'$ is at most

$$p_j = \binom{t}{j} \binom{n-m}{j-1} \left(\frac{j-1+m}{n}\right)^{jl}$$

Since

$$\frac{p_{j-1}}{p_j} = \frac{j(j-1)}{(t-j+1)(n-m-j+2)} \left(\frac{j-2+m}{j-1+m}\right)^{(j-1)l} \left(\frac{n}{j-1+m}\right)^l \geq 1,$$

we have

$$\sum_{j=2}^t p_j \leq (t-1)p_2 = \frac{t(t-1)^2}{2}(n-m)\left(\frac{m+1}{n}\right)^{2l},$$

which is an upper bound for the probability that Hall's lemma fails. \square

3.1 Our distributed key generation scheme

The structure of our scheme is based on Gennaro et. al.'s study on secure distributed key generation [GJKR99]. It is secure against skewing the secret key distribution by a malicious key server.

At beginning, a dealer chooses a $t \times n$ -dimensional evaluation matrix \mathbf{E} and publishes it in a public bulletin board. Then, each key server randomly chooses a random number in Z_q and broadcasts it to the other key servers. By the order of these numbers, each key server gets its label $\in \{1, 2, \dots, n\}$. Since Z_q is a very large set, it is unlikely that a collision occurs.²

Our distributed key generation scheme is as follows:

Setup:

1. A dealer does:
 - (a) Select a large prime $p = 2q + 1$, where q is also prime.
 - (b) Compute generators g and h of G_q , where $G_q = \{a^2 | a \in Z_p^*\}$ is the subgroup of quadratic residues of Z_p^* .
 - (c) Choose a $t \times n$ -dimensional evaluation matrix \mathbf{E} , as described in Section 3.
2. All key servers label themselves into $\{1, 2, \dots, n\}$ by the method described above. Then, we call these key servers as U_1, U_2, \dots, U_n .

Key share establishment:

1. Each key server U_i does the following:
 - (a) Select two t -dimensional vectors \mathbf{a}_i and \mathbf{a}'_i which each consists of k non-zero random entries. The non-zero entries are in the same indexes of \mathbf{a}_i and \mathbf{a}'_i .
 - (b) Compute $\mathbf{s}_i = \mathbf{a}_i \mathbf{E}$, $\mathbf{s}'_i = \mathbf{a}'_i \mathbf{E}$ and the set of his communication key servers $Q_i = \{j | s_{i,j} \neq 0 \vee s'_{i,j} \neq 0\}$.
 - (c) Send $s_{i,j}$ and $s'_{i,j}$ to key server U_j via a secure channel, $j \in Q_i$.
 - (d) Broadcast $C_{i,j} = g^{s_{i,j}} h^{s'_{i,j}} \mod p$, $1 \leq j \leq t$, to all the key servers in Q_i .
2. Each key server U_j does the following:
 - (a) Check validity of the received shares, for each $i, j \in Q_i$,

$$g^{s_{i,j}} h^{s'_{i,j}} \equiv \prod_{k=1}^t C_{i,k}^{\mathbf{E}_{k,j}} \pmod{p}. \quad (2)$$

²By the birthday paradigm, the probability that an collision occurs is at most $\binom{n}{2}/q$. Since n is much much smaller than q , the probability is almost 0.

If the check fails for i , U_j broadcasts a complaint against U_i to the key servers in Q_j .

- (b) If U_j is complained by U_i , it sends $s_{j,i}$ and $s'_{j,i}$ to the key servers in Q_j .
The other key servers in Q_j check validity of $s_{j,i}$ and $s'_{j,i}$ by Equation (2).
If U_j fails the test, it is marked as "dishonest" by the key servers in Q_j .

3. Each key server U_j builds a set H of honest key servers and sets his key share as $x_j = \sum_{i \in H, j \in Q_i} s_{i,j} \bmod q$, which is the j th entry of $(\sum_{i \in H} \mathbf{a}_i) \mathbf{E}$. Note that the secret key is $x = (\sum_{i \in H} \mathbf{a}_i) \cdot \vec{\mathbf{1}}$, where $\vec{\mathbf{1}} = [1 \ 1 \ \dots \ 1]$.

4. Compute the public key $y = g^x \bmod p$.

- (a) Each key server $U_i \in H$ broadcasts $A_{i,k} = g^{a_{i,k}} \bmod p$, $1 \leq k \leq t$, to the key servers in H .
- (b) Each key server U_j in Q_i checks validity of $A_{i,k}$ by verifying whether

$$g^{s_{i,j}} \equiv \prod_{k=1}^t A_{i,k}^{\mathbf{E}_{k,j}} \pmod{p}. \quad (3)$$

If the check fails, U_j broadcasts a complaint against U_i and sends $s_{i,j}$ and $s'_{i,j}$ to the key servers in Q_i .

- (c) If U_i is ever complained, all the key servers in Q_i reconstruct \mathbf{a}_i by solving $\mathbf{s}_i = \mathbf{a}_i \mathbf{E}$ and compute correct $A_{i,k}$, $1 \leq k \leq t$.
- (d) Then, each key server in H computes the public key as

$$y = \prod_{i \in H} \prod_{j=1}^t A_{i,j} \bmod p = g^{(\sum_{i \in H} \mathbf{a}_i) \cdot \vec{\mathbf{1}}} \bmod p.$$

Secret key recovery: Note that in some situations, we don't need to recover the secret key x to finish a task. Only each U_i computes a partial result from its key share x_i .

1. Let T be the set of shown-up key servers in H . If E^T is full-ranked, solve $\sum_{i \in H} \mathbf{a}_i$ by the system of equations

$$\left(\sum_{i \in H} \mathbf{a}_i \right) \mathbf{E}^T = \sum_{i \in H} \mathbf{s}_i.$$

2. The secret key is $x = \sum_{i \in H} \mathbf{a}_i \cdot \vec{\mathbf{1}}$.

3.2 Analysis

The correctness and security of our scheme is shown in the following theorem.

Theorem 3. Assume that n, t, j, l , and m satisfy the condition in Theorem 2. The scheme in Section 3.1 is a secure $(\frac{1}{l} - \epsilon, 1 - \frac{m}{n}, (n - m) \frac{t(t-1)^2}{2} (\frac{m+1}{n})^{2l})$ -PTDKG scheme for some small ϵ , $0 < \epsilon < 1$.

Proof. (Sketch) Correctness follows from the results of Gennaro et. al [GJKR99] almost in the same way.

The bounds $\beta = 1 - \frac{m}{n}$ and $\delta = (n - m)^{\frac{t(t-1)^2}{2}(\frac{m+1}{n})^{2l}}$ are from Theorem 2 directly. For $\alpha = \frac{1}{l} - \epsilon$, each Q_i contains kl key servers at most. Any adversary who controls up to a fraction α of them contains less than k dishonest key servers in Q_i . Since there are k unknown entries in each \mathbf{a}_i , the adversary who controls less than k key servers in Q_i cannot know the information about \mathbf{a}_i .

For the uniform distribution of x over Z_q , we construct a simulator for the scheme. The details are deferred to the full paper. \square

4 Experiments and comparison

We first analyze the probability that the full rank is achieved after deleting about a half of key servers. Recall that l is the band of \mathbf{E} , f is the offset, and t is the number of rows.

For the CS-PTDKG scheme, due to the arrangement of \mathbf{E} , the number of rows is fixed to $t = (n - l)/f$. On allowing $n(1/2 - \epsilon)$ dishonest key servers (say, $\epsilon = 1/10$), Canny and Sorkin suggests $f = 2$, $l = 17 \log n$ and $t = (n - 17 \log n)/2$. Theoretically, the probability of achieving the full rank is $O(n^{-2})$.

For our PTDKG scheme, we shall do some simulation experiments to obtain appropriate l' on the condition that the probability of achieving the full rank is the same as that of the CS-PTDKG scheme.

We take $n = 1000$ and delete about $m = 500$ dishonest key servers randomly. We consider different offsets ($f = 2$, $f = 3$, and $f = 4$) for the CS-PTDKG scheme. The results are shown in Figures 1-3. In each figure, the x -axis indicates the probability of achieving the full rank and the y -axis indicates the number l of non-zero entries in each row of \mathbf{E} . The probability is computed by randomly sampling 500 key servers as "dishonest" many times. We summarize the comparison results in Table 1 on 90% of achieving the full rank. From the table, we can see that the number l' of non-zero entries in each row of our \mathbf{E} is much smaller than that (l) of the CS-PTDKG scheme.

Table 1: Comparison of l with 90% of achieving the full rank. There are $n = 1000$ key servers and $m=500$ of them are dishonest.

	$t = 408 (f = 2)$	$t = 318 (f = 3)$	$t = 242 (f = 4)$
CS-PTDKG	$l = 185$	$l = 45$	$l = 33$
Ours	$l' = 14$	$l' = 11$	$l' = 8$

5 Discussion

Our scheme and the CS-PTDKG scheme have different security parameters. For ours, $\alpha = \frac{1}{l'} - \epsilon$ and $\beta = 1 - \frac{m}{n}$. For the CS-PTDKG scheme $\alpha = \frac{1}{f} - \epsilon$ and $\beta = \frac{1}{f} + \epsilon$. These two set of parameters can be used for different situations. For example, if the number of dishonest key server is relatively small (about one in l' key servers), our scheme is suitable.

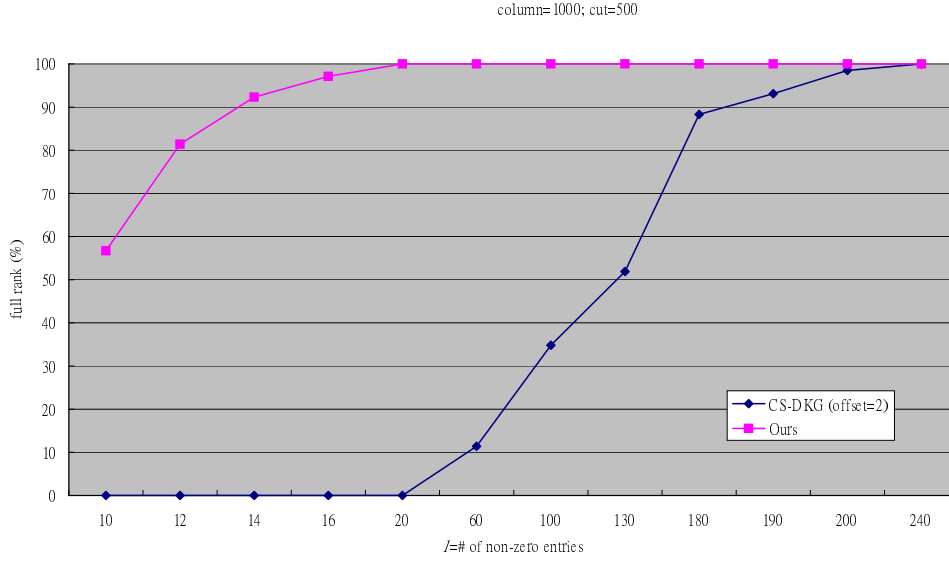


Figure 1: Probability of achieving the full rank for different l , when $f = 2$

Since we are talking about a large number of key servers, a small percent of dishonest key servers is very likely. Our β is adjustable under some constraints. If larger β is desirable, our scheme provides such choice.

The communication cost of our scheme is $k'l'$ and that of the CS-TPDKG scheme is $(k-1)f + l$. If we want our scheme to have the same communication cost as that of the CS-PTDKG scheme, we set $k' = \frac{(k-1)f+l}{l'}$, the number of non-zero entries in each \mathbf{a}_i of our scheme. Note that k' (and $k = \frac{l}{2\epsilon^2}$ for $f=2$ in the CS-PTDKG scheme) affects only the communication cost, not security of the scheme.

References

- [CS04] John Canny and Stephen Sorkin. Practical large-scale distributed key generation. In *Proceedings of Advances in Cryptology - EUROCRYPT '04*, volume 3027 of *LNCS*, pages 138–152. Springer-Verlag, 2004.
- [CT02] Cheng-Kang Chu and Wen-Guey Tzeng. Distributed key generation as a component of an integrated protocol. In *Proceedings of the 4th Information and Communications Security - ICICS '02*, volume 2513 of *LNCS*, pages 411–421. Springer-Verlag, 2002.
- [Fel87] Paul Feldman. A practical scheme for non-interactive verifiable secret sharing. In *28th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 427–437. IEEE, 1987.
- [GJKR99] Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Secure distributed key generation for discrete-log based cryptosystems. In *Proceedings of Advances in Cryptology - EUROCRYPT '99*, volume 1592 of *LNCS*, pages 295–310. Springer-Verlag, 1999.

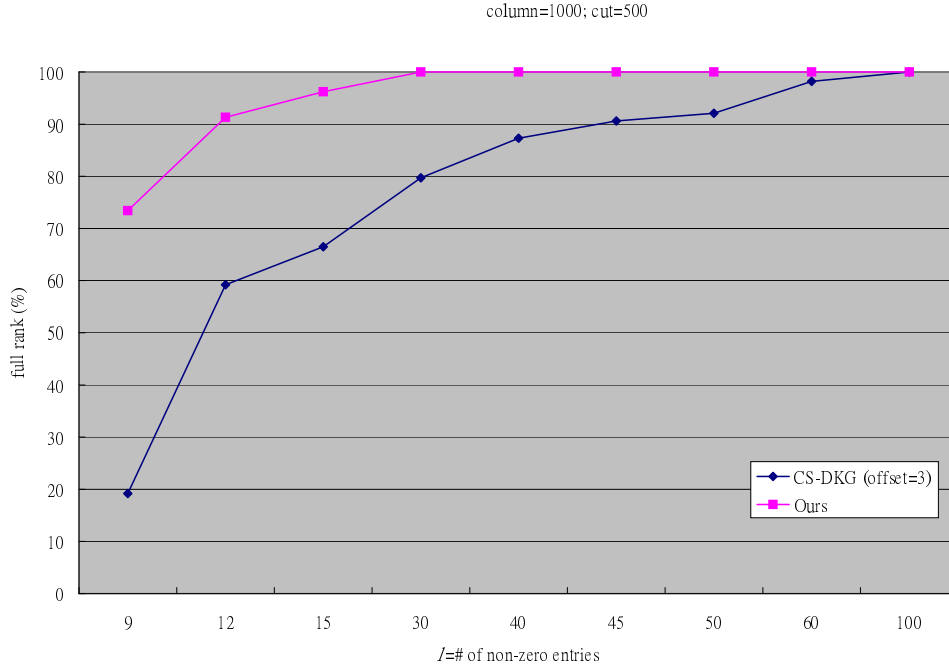


Figure 2: Probability of achieving the full rank for different l , when $f = 3$

- [Ped91a] Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *Proceedings of Advances in Cryptology - CRYPTO '91*, volume 576 of *LNCS*, pages 129–140. Springer-Verlag, 1991.
- [Ped91b] Torben P. Pedersen. A threshold cryptosystem without a trusted party. In *Proceedings of Advances in Cryptology - EUROCRYPT '91*, volume 547 of *LNCS*, pages 522–526. Springer-Verlag, 1991.
- [Sha79] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

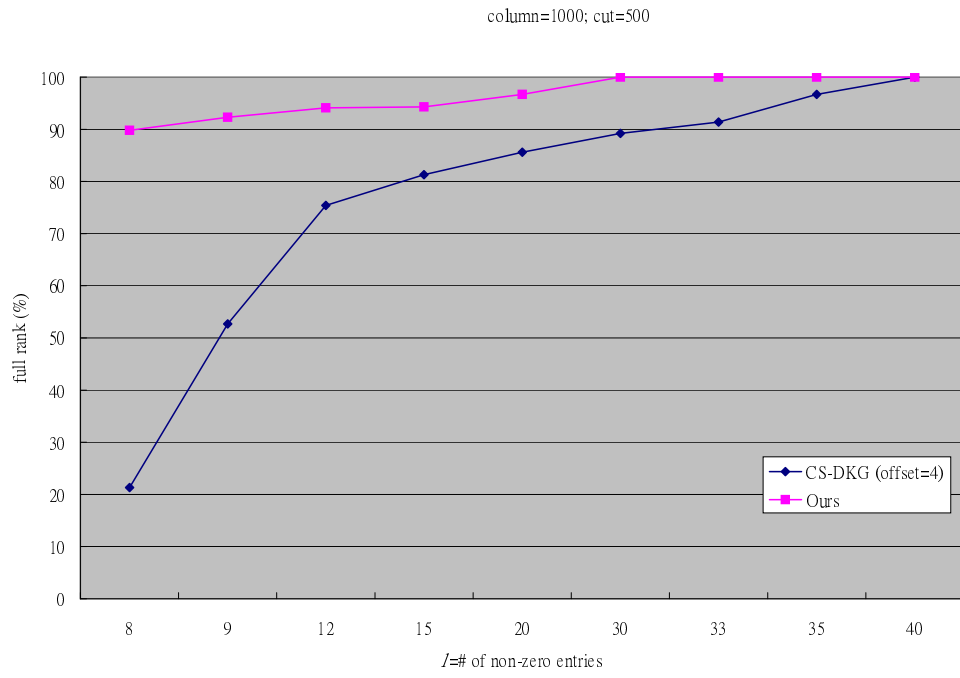


Figure 3: Probability of achieving the full rank for different l , when $f = 4$