On the Security of a Practical Constant-Size Ring Signature Scheme

Jianhong Zhang[1], Wenle Bai[1,2], and Zhengtao Jiang[1,3,4]
(Corresponding author: Weile Bai)
School of Electronic and Information Engineering, North China University of Technology[1]
Beijing 100144, China
(Email: bwl@ncut.edu.cn)
Guangxi Key Laboratory of Cryptography and Information Security[2]
Guangxi Key Lab of Multi-source Information Mining & Security[3]
School of Electronic and Information Engineering, China Communication University[4]
(Received June 15, 2018; Revised and Accepted Nov. 2, 2018; First Online July 8, 2019)

Abstract

Due to decentration and anonymity, "bitcoin" cryptocurrency is widely paid attention. However, it only furnishes pseudo-anonymity instead of authentic anonymity. As an anonymous technique, ring signature is a candidate to provide authentic anonymity in cryptocurrency [2]. However, in most of existing ring signatures, the length of signature grows linearly with the size of the ring. To construct constant-size signature, Qin *et al.* recently proposed a practical constant-size ring signature scheme, and claimed that their scheme can provide unforgeability and anonymity which are two basic security requirments of ring signature. Unfortunately, in this letter, we show that their scheme is insecure against unforgeability attack and anonymity attack. Finally, the two detail attacks are given.

Keywords: Anonymity; Constant-Size Ring Signature; Security Attack; Unforgeability

1 Introduction

As a decentralized distributed public ledger, blockchain can furnish trust to operations between unrelated parties, without requiring the collaboration of a trusted third party. However, the public verifiability and decentration of blockchain transaction often do not provide the strong security and privacy properties required by the users. It can only provide pseudonymity instead of true anonymity. In cryptocurrency, ring signature, stealth address, and zero-knowledge proof are several cryptographic techniques which can achieve privacy protection for transaction entities.

In particular, ring signature can often been used to protect the identity anonymity of the user. To provide real anonymity of transaction party, ring signature is applied to conceal the origin of a transaction in the 'Monero' cryptocurrency. For the perspective of the outsider, it can not distinguish who is the actual sponsor of transaction.

In 2001, a novel anonymous signature conception [14] was invented by Rivest, Shamir and Tauman, it was named as **ring signature** since the structure of signature generation looks like a ring. Like group signature, a user is allowed to produce a signature on behalf of the whole group without the cooperation of the other users of the group in ring signature. Anonymity and unforgeability [1, 3, 7, 9, 11] are two basic properties of a secure ring signature. Anonymity can guarantee the identity privacy of the actual signer. Unforgeability can ensure the security of signature algorithm since it can prevent an adversary from forging a signature of new message m. These properties make that ring signature has very important application such as anonymous authentication in VANET. However, most of the existing ring signature schemes exist a common flaw: "*The length of ring signature is linear with the size of the ring.*". Therefore, the larger the ring size is, the longer the length of ring signature is.

To reduce the length of ring signature, Chandran *et al.* presented the first sub-linear length ring signature [5] in the standard model by utilizing private information retrieval technique. However, the security of their scheme builds on the composite-order bilinear group which is inefficient. Later on, Ghadafi constructed a sub-linear size ring signature in the prime-order setting [8].

There only exist two constant-size ring signature schemes so far. One is Dodis *et al.*'s anonymous identification scheme [6] which is based on strong RSA problem, the other is Bose *et al.*'s ring signature [4] which is based on two Diffie-Hellman assumption. Nevertheless, the two scheme is inefficient in practice since the scheme in [6] makes use of the strong RSA based instantiation which uses quite complex Σ- protocols, and the scheme in [4] is based on the q-strong Diffie- Hellman

Table 1: Notions

Notions	Implication
$\mathbb{G}$,$\mathbb{G}_T$:	two groups with the same order q
e :	A bilinear pairing map
PPT:	Probability polynomial time
DLP:	discrete logarithm problem
ACC :	A dynamic accumulator
PPT:	the probabilistic polynomial time
H_0, H_1:	two cryptographic hash functions
(x_i, PK_i):	public-private pair of user i

(q-SDH) assumption and the symmetric external Diffie Hellman (SXDH) assumption via the Boneh- Boyen signature scheme. These techniques make the two schemes less efficient in practice.

Our contributions: Recently, based on the discrete logarithm problem (for short, DLP) assumption, Qin *et al.* proposed a practical constant-size ring signature scheme [17]. Their construction is very simple and efficient. And they claimed that their scheme can provide security proof of unforgeability and anonymity. Unfortunately, in this work, we find that their scheme is insecure by analyzing the security of Qin et al.'s scheme. Their scheme suffers from two security flaws. One is that it exists universal forgeability, namely, any one can forge a signature on arbitrary a message; The other is that it can not achieve anonymous prevention of real signer's identity. Finally, the detail attacks are given.

2 Preliminaries

In this work, we make use of bilinear map technique to construct our scheme. To make our paper self-contained, we will introduce some necessary cryptographic background which is related to bilinear map . And they are also the basis of achieving the security of our proposed scheme.

2.1 Bilinear Maps [10, 15, 16]

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of the same prime order q, g be a generator of group $\mathbb{G}_1$. A bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is a map that for all $g, h \in \mathbb{G}_1$ and $a, b \in_R Z_q^*$, $e(g^a, h^b) = e(g, h)^{ab}$. And there exists a computable algorithm that can efficiently compute e and $e(g, g) \neq 1$.

2.2 Accumulator

Definition 1. *An accumulator is a tuple (X_χ, F_χ), where $\chi \in N$, N is the set of positive integers. X_χ is called the value domain of the accumulator, and F_χ is a collection of pairs of functions such that each $(f, y) \in F_\chi$ is defined as $f : U_f \times X_f \to U_f$ for some $X_\chi \subseteq X_f^{ext}$, and $y : U_f \to U_y$ is a bijective function where U_f and U_y denote the value domain of functions f and y respectively. In addition, an accumulator should satisfy the following properties.*

1) Efficient generation. There exists an efficient algorithm that takes as input a security parameter χ and outputs a random element $(f, y) \in_R F_\chi$, possibly together with some auxiliary information. And in the following sections, we denote the algorithm by ACC.Gen.

2) Quasi-commutativity. For every $\chi \in N$, $(f, y) \in F_\chi, u \in U_f$, $x_1, x_2 \in X_\chi$: $f(f(u, x_1), x_2) = f(f(u, x_2), x_1)$. For any $\chi \in N$, $(f, y) \in F_\chi$ and $X = \{x_1, x_2, \cdots, x_n\} \subset X_\chi$, we call $y(f(f(u, x_1), \cdots, x_n))$ the accumulated value of the set X over u. Due to quasi-commutativity, the value $y(f(f(u, x_1), \cdots, x_n))$ is independent of the order of x_i and is denoted by $f(u, X)$.

3) Efficient Evaluation. For every $(f, y) \in F_\chi, u \in U_f$ and $X \subset X_\chi$ with polynomially-bound size: $y(f(u, X))$ is computable in time polynomial in χ, and we use ACC.Eval to represent the process of computing the accumulated value. Also, there is a witness w meaning that some variable x has been accumulated within $v = f(u, X)$ iff $f(w, x) = v$, and we use ACC.Wit to denote computing the witness w.

For simplicity, in the context, we adopt the accumulator in [13]. Namely, the functions (f, y) is defined as follows:

$$f : (\beta, x) \to \beta(x + d), y : x \to x$$

where $d \in Z_q$ and $\beta \in Z_q$, and $y(x) = x$ is an identity function. For this accumulator, we have $f(\beta, X) = \beta(x_1 + d) \cdots (x_n + d)$ for a set $X = \{x_1, x_2, \cdots, x_n\}$.

Obviously, all the above three properties in Definition 1 are satisfied.

2.3 Ring Signature Definition

A ring signature scheme is a tuple (Setup, Gen, Sign, Verify) of PPT algorithms, where each of them means generating a key pair, signing a message, and verifying the signature for the message using the corresponding public keys, respectively. Formally they are described as follows.

1) Setup(1^l). It takes as input a security parameter l, and outputs the system parameters *params*.

2) Gen(*params*, i). It inputs the identity information i of a user and *params*, and outputs a public key PK_i and a private key SK_i for each member i.

3) $Sign(SK, m, R)$. The signer outputs a signature δ on a message m with respect to a ring R using the signing key SK_i. We assume that the number of public keys in the ring $|R| > 2$, and there is exactly one public key in R corresponding to the signing key SK_i, and all the keys in R are generated by Gen.

4) Verify(δ, m, R). The verifier can be anyone, including the adversary, who verifies the signature on a message m with respect to the ring R. If the verifier accepts the signature, then the algorithm returns 1; otherwise, returns 0.

3 Reviews of Qin *et al.*'s Constant-Size Ring Signature

To achieve constant-length ring signature, Qin *et al.* proposed a practical constant-size ring signature (for short, PCRS) in [17]. The PCRS scheme consists of four algorithms. We will briefly review their algorithms. Please the interesting readers refer to [17] for the detail. For convenience, the used notations in the following parts are summarized in Table 1.

3.1 System Initialization

Taking a security parameter λ as input, it outputs two cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_T$ with the same prime order p, and the discrete logarithm problem in $\mathbb{G}_1$ is hard. Let g be a generator of group $\mathbb{G}_1$ and $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ be a bilinear pairing map. H_0 and H_1 are hash functions which satisfy $H_0 : \{0,1\}^* \to Z_q$ and $H_1 : \{0,1\}^* \to Z_q$, where q is also a prime which satisfies $q|p-1$. For each user $i \in \{1, \cdots, n\}$, it chooses x_i as its private key and calculates public key $PK_i = g^{x_i}$. Let R denote the public key list of n users,namely $R = \{PK_1, \cdots, PK_n\}$. Let AAC denote an accumulator, refer to [17] for the detail. And then we invoke AAC.Gen to produce $f : (\beta, x) \to \beta(x+d)$ and the corresponding $P_{pub} = g^d$. Finally, the system parameter $Param = \{\mathbb{G}_1, \mathbb{G}_T, e, p, q, g, f, P_{pub}, H_0, H_1\}$.

3.2 Signing

Let $L = \{H_0(PK_i)\}_{i=1}^n$ and $L' = \{H_0(PK_i)\}_{i=1, i\neq s}^n$. Given a public key list $R = \{PK_1, PK_2, \cdots, PK_n\}$ and system parameter $Param$, to produce a signature on message m, a user s with public-private key pair (x_s, PK_s) executes the following steps:

1) First, it calculates $V = ACC.Eval(f, L)$ and $W = ACC.Wit(f, L')$. Note that $V = W(H_0(PK_s) + d)$.

2) Next, it uniformly samples $r \in Z_q$ to calculate $U = W + r$ and $P_U = g^U$.

3) Pick $k_1, k_2 \in Z_q$ at random to calculate $\prod = e(g, P_U)^{-k_1} e(P_{pub}, g)^{k_2} e(R', g)$ where $e(R', g) = \prod_{i=1, i\neq s}^n e(PK_i, g)$.

4) And it calculates

$$c = H_1(m||V||P_{pub}||P_U||\prod||R)$$

where $R = \{PK_1, \cdots, PK_n\}$.

5) Finally, it computes

$$\begin{aligned} s_1 &= k_1 + cH_0(PK_s) \\ s_2 &= k_2 + cr \\ s_3 &= crH_0(PK_s) - x_s. \end{aligned}$$

6) The resultant ring signature is $\delta = (c, P_{pub}, P_U, V, s_1, s_2, s_3)$

3.3 Verifying

Given a message m and its ring signature $\delta = (c, P_{pub}, P_U, V, s_1, s_2, s_3)$ as well as public key list R, a verifier can conduct the following procedure. First, it calculates

$$\begin{aligned} \Pi' &= e(g, P_U)^{-s_1} e(P_{pub}, g)^{s_2} e(g, g)^{s_3} e(P_{pub}, P_U)^{-c} \\ &\quad \cdot e(g, g^V)^c e(R, g). \end{aligned}$$

And then it checks whether $c \stackrel{?}{=} H_1(m||V||P_{pub}||P_U||\Pi'||R)$. If it holds, then the signature is accepted; otherwise, refuse it.

4 Security Analysis

Recently, based on the DLP problem, Qin *et al.* proposed a PCRS scheme. Their scheme is more efficient than the existing two constant size ring signature schemes in terms of computational cost. At the same time, they also claim that their scheme can achieve anonymity of the signer's identity and provide perfect zero knowledge for the verifier. By analyzing the security of their scheme, we find that their scheme does not achieve anonymity. Given a ring signature δ, a verifier can know which signer produce the signature δ. Furthermore, given two ring signatures δ and δ', the verifier can know whether the two ring signatures are from the same signer. The detail attacks are given as blow.

4.1 Attack on Anonymity

Given a ring signature $\delta = (c, P_{pub}, P_U, V, s_1, s_2, s_3)$ on message M, an attack A first computes

$$\begin{aligned} \Pi &= e(g, P_U)^{-s_1} e(P_{pub}, g)^{s_2} e(g, g)^{s_3} \\ &\quad \cdot e(P_{pub}, P_U)^{-c} e(g, g^V)^c e(R, g) \\ c &= H_1(m||V||P_{pub}||P_U||\Pi') \end{aligned}$$

For $j = 1$ to n
{

1) It computes $\bar{k}_1 = s_1 - c \cdot H_0(PK_j)$;

2) And then it computes $s_2 H_0(PK_j) - s_3 = k_2 H_0(PK_j) + x_s$;

3) And it checks

$$\begin{aligned} & e(P_{pub}, PK_j) \cdot \left(\frac{\Pi' \cdot e(PK_j, g)}{e(R, g)} \cdot e(g, P_U)^{\bar{k}_1}\right)^{H_0(PK_j)} \\ & \stackrel{?}{=} e(P_{pub}, g)^{s_2 H_0(PK_j) - s_3} \end{aligned} \quad (1)$$

4) If Equation (1) holds, it breaks it.

}

If $j \leq n$ then it outputs the identity index j of the real signer. Otherwise, it outputs False.

We will show that our attack is valid since if the real signer's public key is PK_s, we can obtain the following relation

$$\begin{aligned} k_1 &= s_1 - c \cdot H_0(PK_s) \quad (2) \\ s_2 H_0(PK_s) - s_3 &= k_2 H_0(PK_s) + cr H_0(PK_s) \\ &\quad -cr H_0(PK_s) + x_s \\ &= k_2 H_0(PK_s) + x_s \quad (3) \end{aligned}$$

Thus, we have

$$\begin{aligned} e(P_{pub}, g)^{s_2 H_0(PK_s) - s_3} &= e(P_{pub}, g)^{k_2 H_0(PK_s) + x_s} \\ &\Updownarrow \\ e(P_{pub}, g)^{s_2 H_0(PK_s) - s_3} &= e(P_{pub}, g)^{k_2 H_0(PK_s)} e(P_{pub}, PK_s) \\ &\Updownarrow \\ e(P_{pub}, g)^{s_2 H_0(PK_s) - s_3} &= \left(\frac{\Pi \cdot e(g, P_U)^{k_1}}{e(R', g)}\right)^{H_0(PK_s)} \\ &\quad \cdot e(P_{pub}, PK_s) \\ &\Updownarrow \\ e(P_{pub}, g)^{s_2 H_0(PK_s) - s_3} &= \left(\frac{\Pi \cdot e(g, P_U^{k_1} \cdot PK_s)}{e(R, g)}\right)^{H_0(PK_s)} \\ &\quad e(P_{pub}, PK_s). \end{aligned}$$

It means that the real signer's public key PK must satisfy Equation (1). Thus our attack is valid.

The reason to produce such attack is that random number k_1 in signing phase can be recovered by making use of the hash value of $m||V||P_{pub}||\Pi||R$ and the actual signer's identity PK_s. At the same time, s_2 and s_3 in the ring signature exist a certain relevance. It makes that the relation $s_2 H_0(PK_s) - s_3 = k_2 H_0(PK_s) + x_s$ holds. Thus, it reveals the relevant identity information of the actual signer.

4.2 Attack on Unforgeability

For a ring signature, unforgeability should be a very important property, namely, it is difficulty to forge a ring signature on a new message m^*. The property ensures that ring signature can provide stronger unforgeability. Unfortunately, by analyzing the security of their scheme, we show that Qin *et al.*'s ring signature scheme also does not satisfy unforgeability. The detail attack is given as below.

1) Let m^* be a forged message. $L = \{PK_1, \cdots, PK_n\}$ is a public key list.

2) First, the attacker picks a random number $\alpha \in Z_q$ to calculate $P_U^* = g^\alpha$.

3) Then, it chooses three random numbers $r_1, r_2, r_3 \in Z_q^3$ to calculate

$$\Pi^* = e(g, P_U^*)^{-r_1} e(P_{pub}, g)^{r_2} e(g, g)^{r_3} e(R, g)$$

where $e(R, g) = \prod_{j=1}^{n} e(PK_j, g)$.

4) Next , it randomly selects $v \in Z_q$ to set $V^* = v$, and computes $c^* = H_1(m^*||V^*||P_{pub}||P_U^*||\Pi^*||R)$, where $R = \prod_{j=1}^{n} PK_j$.

5) Subsequently, the attacker sets $s_1^* = r_1$, $s_2^* = r_2 + \alpha \cdot c^*$ and $s_3^* = r_3 - V^* \cdot c^*$.

6) Finally, the resultant ring signature on message m^* is $\delta^* = (c^*, P_{pub}, P_U^*, V^*, s_1^*, s_2^*, s_3^*)$.

In the following, we show that the above forged signature δ^* is valid, that is to say, it can pass the verification checking. Because

$$\begin{aligned} \Pi' &= e(g, P_U^*)^{-s_1^*} e(P_{pub}, g)^{s_2^*} e(g, g)^{s_3^*} e(P_{pub}, P_U^*)^{-c^*} \\ &\quad e(g, g^{V^*})^{c^*} e(R, g) \\ &= e(g, g^\alpha)^{-r_1} e(P_{pub}, g)^{r_2 + \alpha c^*} e(g, g)^{r_3 - V^* c^*} \\ &\quad e(P_{pub}, g^\alpha)^{-c^*} e(g, g^{V^*})^{c^*} e(R, g) \\ &= e(g, g^\alpha)^{-r_1} e(P_{pub}, g)^{r_2} e(g, g)^{r_3} e(R, g) \\ c^* &= H_1(m^*||V^*||P_{pub}||P_U^*||\Pi^*||R). \end{aligned}$$

It means that our forged ring signature is valid since it can pass the verification equation. Thus, Qin *et al.*'s scheme is insecure. Any one can forge a ring signature on arbitrary a mesasage. Thus, our attack is valid.

In the following, we analyze the reason to lead to forgery attack. For the verification algorithm of the ring signature, we can find that any one can calculate Π' by random choosing $(s_1, s_2, s_3, c, P_U, V)$. Essentially, Π' is irrelevant to the hash value c since Π' can be computed without c by choosing the appropriate (s_1, s_2, P_U, V, s_3). Thus, the main reason to produce such attack is that the generation of Π is independent of hash value c. c can not restrain the generation of Π.

5 Conclusions

In this letter, we analyze the security of *Qin et al.*'s PCRS scheme, and show that their scheme is insecure. It can not achieve two security properties of ring signature: **unforgeability** and **anonymity**. In their scheme, any one can forge a ring signature on arbitrary a message, and given a ring signature δ, the verifier can know who is the actual signer. Finally, our analysis is confirmed thought two concrete attacks, the corresponding reasons to produce such attacks are given. It is our future work how to design a secure and practical ring signature scheme with constant-size.

Acknowledgments

This research was was supported by Beijing Municipal Natural Science Foundation (Nos.4162020), Guangxi Key Laboratory of Cryptography and Information Security (No.GCIS201710) and Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (No.MIMS16-01).

References

[1] M. Bellare and G. Neven, "Multi-signatures in the plain publickey model and a general forking lemma," in *Proceedings of the 13th ACM Conference Computer and Communications Security*, pp. 390-399, 2006.

[2] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Proceedings of IEEE Symposium on Security and Privacy (SP'14)*, pp. 459–474, May 2014.

[3] D. Bleichenbacher and U. Maurer, "On the efficiency of one-time digital signatures," in *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'96)*, pp. 176–180, June 2008.

[4] P. Bose and C. P. Rangan and D. Das, "Constant size ring signature without random oracle," in *Proceedings of the 20th Australasian Conference Information Security and Privacy*, pp. 230–247, July 2015.

[5] N. Chandran, A. Sahai and J. Groth, "Ring signatures of sublinear size without random oracles," in *International Colloquium on Automata, Languages, and Programming (ICALP'07)*, pp. 423–434, vol. 4596, June 2007.

[6] Y. Dodis, A. Kiayias, A. Nicolosi, and V. Shoup, "Anonymous identification in ad hoc groups," in *Proceedings International Conference Theory and Application Cryptographic Techniques*, pp. 609–626, 2004.

[7] R. Ehmet, L. Z. Deng, Y. Y. Zhang, and J. W. Zeng, "Multi-proxy multi-signature without pairing from certificateless cryptography," *International Journal of Network Security*, vol. 20, no. 3, pp. 403–413, 2018.

[8] E. M. Ghadafi, "Sub-linear blind ring signatures without random oracles," in *Proceedings of the 14th IMA International Conference Cryptography and Coding (IMACC'13)*, pp. 304–323, Dec. 2013.

[9] S. Goldwasser, C. Rackoff and S. Micali, "The knowledge complexity of interactive proof systems," in *SIAM Journal on Computing*, pp. 186–208, 1989.

[10] M. S. Hwang and I. C. Lin, *Introduction to Information and Network Security (6ed, in Chinese)*, Taiwan: McGraw Hill, 2017.

[11] M. S. Hwang and C. Y. Liu, "Authenticated encryption schemes: Current status and key issues," *International Journal of Network Security*, vol. 1, no. 2, pp. 61–73, 2005.

[12] M. S. Hwang, S. F. Tzeng, C. S. Tsai, "Generalization of proxy signature based on elliptic curves", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 73–84, 2004.

[13] L. Nguyen, "Accumulators from bilinear pairings and applications," in *The Cryptographers Track at the RSA Conference (CT-RSA'05)*, pp. 275–292, 2005.

[14] R. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 552–565, vol. 2248, June 2001.

[15] J. H. Zhang and Q. C. Dong, "Efficient id-based public auditing for the outsourced data in cloud storage," *Information Sciences*, vol. 343-344, no. 2, pp. 1–14, 2016.

[16] J. H. Zhang, P. Y. Li, and M. Xu, "On the security of a mutual verifiable provable data auditing in public cloud storage," *International Journal of Network Security*, vol. 19, no. 4, pp. 605–612, 2017.

[17] Y. L. Zhao, M. J. Qin and Z. J. Ma, "Practical constant-size ring signature," *Journal of Computer Science and Technology*, vol. 33, no. 3, pp. 533-541, 2018.

Biography

Jianhong Zhang received his Ph.D. degrees in Cryptography from Xidian University, Xian, Shanxi, in 2004 and his M.S. degree in Computer Software from Guizhou University, Guiyang, Guizhou, in 2001. He was engaging in postdoctoral research at Peking University from October 2005 to December 2007. He has been an Assistant Processor of College of Sciences, North China University of Technology, Beijing China, since 2001. His research interests include computer networks, cryptography, electronic commerce security, computer software.

Wenle Baig received his Ph.D. degrees in Communication Networking from Beijing University of Posts and Telecommunications; , Beijing, in 2006 and his M.S. degree in Computer Software from Beijing University of Posts and Telecommunications,Beijing in 2001. He has been an Assistant Processor of College of Sciences, North China University of Technology, Beijing China, since 2001. His research interests include computer networks, cryptography, electronic commerce security, computer software.

Zhengtao Jiang received his Ph.D. degrees in Cryptography from Xidian University, Xian, Shanxi, in 2005 and his M.S. degree in School of Mathematics from Central South University, Changsha, Hunan, in 2002. He was engaging in postdoctoral research at Beihang University from October 2008 to December 20010. He has been an Assistant Processor of School of Computer Science, China Communication University, Beijing China, since 2001. His research interests include computer networks, cryptography, electronic commerce security, computer software.