A Thesis for the Degree of Master

Design of Secure and Efficient E-commerce Protocols Using Cryptographic Primitives

Wooseok Ham

School of Engineering

Information and Communications University

2003

Design of Secure and Efficient E-commerce Protocols Using Cryptographic Primitives

Design of Secure and Efficient E-commerce Protocols Using Cryptographic Primitives

Advisor : Professor Kwangjo Kim

by

Wooseok Ham
School of Engineering
Information and Communications University

A thesis submitted to the faculty of Information and Communications University in partial fulfillment of the requirements for the degree of Master of Science in the School of Engineering

Daejeon, Korea
Dec. 27. 2002
Approved by

(signed)

Professor Kwangjo Kim
Major Advisor

Design of Secure and Efficient E-commerce Protocols Using Cryptographic Primitives

Wooseok Ham

We certify that this work has passed the scholastic standards required by the Information and Communications University as a thesis for the degree of Master

Dec. 27. 2002

Approved:

Chairman of the Committee
Kwangjo Kim, Professor
School of Engineering

Committee Member
Myungchul Kim, Associate Professor
School of Engineering

Committee Member
Choonsik Park, Ph.D
NSRI

M.S. Wooseok Ham
2002140

Design of Secure and Efficient E-commerce Protocols Using Cryptographic Primitives

School of Engineering, 2003, 53p.
Major Advisor : Prof. Kwangjo Kim.
Text in English

Abstract

With the advance of the Internet and development of information technologies, many conventional off-line services such as banking, mailing and governmental affairs are migrating to on-line ones. Currently, building information-oriented company comes to be not only scientific technology but also business strategy to acquire competitive power. E-commerce, which treats commercial activities by on-line, is the most prominent example and intimately associated with our real life.

However, people are still hesitant about using such convenient tools. This is originated from uncertainties on safety of their information. Inherent weaknesses of the Internet and trade-offs between performance and security increases users' distrust. Large number of communications which contains user's confidential message are confronted with malicious behaviors. Thus it is obvious that we should devote ourselves to designing secure E-commerce applications but not compromising efficiency.

Through this thesis, we propose two secure and efficient E-commerce protocols: mobile payment system and on-line sealed-bid auction. Two protocols are based on number-theoretic hard problems like DLP and use cryptographic hash function and digital signature as major primi-

tives.

In case of mobile payment, it computes only two modular multiplications, one modular inversion and two hashes by the customer to pay using two public key pairs and keyed hash function. These low computation makes the protocol loaded and run in mobile devices. Nevertheless, it satisfies general electronic payment requirements; unforgeability and double spending prevention.

Two strong sealed-bid auction protocols are presented, which arebased on RSA problem and Discrete Logarithm Problem, respectively. Main characteristics of the protocols are non-repudiation of winner(s) but keeping anonymity during the bidding process. And the computational complexity to decide winner(s) is reduced to $\mathcal{O}(n \log_2 P)$, where n is the number of bidders and P is the number of possible bidding prices.

Furthermore, we analysis proposed protocols in terms of security and performance and give comparisons to others.

Contents

List of Tables

List of Figures

List of Abbreviations

DLP Discrete Logarithm Problem

E-commerce Electronic commerce

E-government Electronic government

E-office Electronic office

E-learning Electronic learning

M-commerce Mobile commerce

WAP Wireless Application Protocol

SET Secure Electronic Transaction

SSL Secure Socket Layer

CA Certification Authority

Algo. Algorithm

List of Notations

PK Public Key

SK Secret Key

PKS Public Key Set

$\mathcal{KG}$ Key Generation algorithm

$\mathcal{H}$ Collision-resistant one-way hash function

KH Keyed hash chain

DSig Digital signature

ID_X Identity of X

C Customer

V Vendor

B Bank

Conf Confirmation receipt

B Bidder

A Auctioneer

P The number of possible bidding prices

Bp Bidding price

I. Introduction

1.1 E-commerce and Security

Rapid improvement of information technologies and widespread diffusion of communication networks via the Internet have being changed our daily lives in a radical and electronic way. E-commerce, E-government, E-office, E-learning, *etc.*, those terms have been newly introduced to illustrate the impacts and changes of our social and cultural environments from them. Also, it is obvious that these trends toward electronic world would be increasingly accelerating.

Among them, E-commerce is the most pervasive and prominent area. E-commerce is the business process of selling and buying the products, goods and services by on-line communications. It can be highly beneficial in reducing business costs and in creating opportunities for new or improved customer services: customers feel convenience to order and are able to collect plenty of information to compare analogous products which are manufactured from the different vendors, vendors can trade globally and find new market with cut down investment, financial facilities like bank can reduce transaction cost.

Regardless of those motivating benefits, some barriers interrupt the promotion of E-commerce. Those are abuse and misuse of information and failures of systems. The sources of such risks comes from several factors such as malicious attacks exploiting external and internal vulnerabilities, carelessness of users and natural disasters. If those risks are realized, we face several big and small losses: direct financial loss, loss of confidential information, loss of customer confidence, loss of business opportunity, inconvenience, *etc.*.

From above discussions, it is clear that we must pay careful attention to *security* in E-commerce. *Secure E-commerce* generally employs information security functions such as authentication, confidentiality, and data integrity to deal with such risks. Commonly, it implies the use of cryptographic-based technologies such as encryption and digital signatures, especially when valuable or private information is communicated over open systems, or when the potential for repudiation of transactions is unacceptable. As a practical matter, secure E-commerce may come to mean the use of information security mechanisms to ensure the reliability of business transactions over insecure networks.

In addition, secure E-commerce should be *efficient*. We generally regard that adding security technologies to E-commerce applications degrades their performance and increases transaction cost. It is not preferable. Resulting quality of services and total cost after integrating security should be reasonable to the associating parties.

Research Goal

Our research goal through this thesis is to design secure and efficient E-commerce applications. We define secure and efficient E-commerce applications as follows:

Definition 1.1.1 Secure and efficient E-commerce applications *are E-commerce applications that use security procedures and techniques to be secure against potential vulnerabilities and attacks and integrating those procedures and techniques can be easily adopted under the current computing environments with affordable consumption of resources.*

Out of various E-commerce areas, we concentrated on *mobile payment system* and *sealed-bid auction protocol*, since two areas are most newest and highlighting in the literature.

1.2 Our Contribution

In this thesis, we try to review recent active researches on mobile payment and on-line auction systems. We also try to classify their own both security and efficiency requirements and give better design than those of other analogous works. We summarize our contributions as follows:

Mobile Payment System : we present a secure one-way mobile payment system that executes only two modular multiplications, one modular inversion and two hashes by the customer using two public key pairs and keyed hash function. Thus the customer(*i.e.* mobile device) conserves low computation load without any expensive modular exponentiation required for RSA or Diffie-Hellman operation. In addition, only one unilateral communication from the customer to the vendor is sufficient to complete payment. These characteristics will make our scheme easily applicable to the mobile environments. The security of the proposed mobile payment system is proved to be equivalent to the intractability of the discrete logarithm problem.

Sealed-Bid Auctions : we propose two sealed-bid auction protocols that one is based on RSA problem and the other on Discrete Logarithm problem. The peculiar characteristics of new protocols are non-repudiation of bidders preserving their anonymity and the reduced computational complexity to $\mathcal{O}(n \log_2 P)$, where n and P denote the number of bidders and the number of possible bidding prices, respectively. Our protocols have additional characteristics such as privacy, publicly verifiability, fairness and walk-awayness. We claim that this low complexity is preferable in a large scale auction.

1.3 Organization of the thesis

In Chapter II, we briefly introduce basic cryptographic primitives which are fundamentally used for the construction of our E-commerce applications and present related works to our areas. Chapter III describes our secure mobile payment system with security and performance evaluations, in which the limitation of mobile environments and requirements are specified. Chapter IV gives anonymous and non-repudiable sealed-bid auction protocols in analogous points of Chapter III. Chapter V summarizes this thesis and concludes with further works.

II. Preliminaries

In this chapter, we introduce basic cryptographic primitives which would be employed through our protocols and previous works related to ours.

2.1 Cryptographic Primitives

Many cryptographic techniques present the fundamental tools for researchers to construct secure applications. As most of them are based on number-theoretic hard problems, polynomially bounded attackers are defeated in solving those problems or have to consume expensive resources to penetrate the applications in a reasonable time. However, we have to pay close attention to select security parameters of those techniques as computational systems are developing fast and new attack methods on the based problems are continually emerging.

In this chapter, we describes major cryptographic primitives and techniques frequently hired to design E-commerce applications.

2.1.1 Discrete Logarithm Problem

The intractability of the discrete logarithm problem(DLP)[22] is a major primitive to design cryptographic applications. Diffie-Hellman key agreement and ElGamal encryption are its exemplary derivatives.

Definition 2.1.1 *The discrete logarithm problem(DLP) is the following: given a prime p, a generator α of $\mathbb{Z}_p^*$, and an element $\beta \in \mathbb{Z}_p^*$, find the integer x, $0 \leq x \leq p-2$, such that $\alpha^x \equiv \beta \pmod{p}$.*

The size of a prime p is recommended to be bigger than 1,024 bits in order to resistant against malicious attackers under the current computational environment.

2.1.2 Digital Signature

Digital signature associates a message with an originating entity. When a dispute arises as to whether a party signed a document, a mediator can resolve the matter fairly verifying it without accessing secret information of the signer. Digital signature is able to provide authentication, non-repudiation and data integrity. Many applications in information security like the certificate of public keys are adopting this technique to support desired security properties.

Digital signature schemes have several forms(blind, undeniable, proxy, one-time, *etc.*) and each form addresses different goals. We can apply these signature variants according to circumstances to satisfy the requirements of cryptographic applications. Out of several digital signature schemes, blind signature schemes have been widely employed in E-commerce area. We briefly introduce them in the next.

Blind Signature

The concept of a blind signature scheme was introduced by Chaum[5] in 1982. The basic idea is as follows: A sender A blinds an original message(m) and transfers the blinded message(m^*) to a signer B. B signs on m^* and returns to A. Then A can compute B's valid signature on m of his choice. The purpose of a blind signature is to prevent the signer B from observing the message it signs and the signature; hence, it is later unable to associate the signed message with the sender A.

The following three steps shows Chaum's blind signature on a message m in RSA setting. (e, n) and d are public and private key pairs. k

is a randomly selected integer subject to $1 \leq k \leq n-1$.

Blinding. A computes $m^* = mk^e \bmod n$ and sends this to B.

Signing. B computes $s^* = (m^*)^d \bmod n$ and sends this to A.

Unblinding. A computes $s = s^*/k \bmod n$. s is B's signature on m.

2.1.3 Hash Function

The basic operation of hash functions is to map an element of larger domains to an element of smaller domains. This property is utilized in many non-cryptographic computer applications like storage allocation to improve performance. However, cryptographic hash functions (hereinafter, simply hash functions) has more important aspects then conventional ones, which makes them playing a fundamental role in modern cryptography.

The purpose of hash functions in cryptographic sense to provide data integrity and message authentication. For these usage, adopted hash functions($\mathcal{H}$) should satisfy the following requirements:

Compression. Given an input x of arbitrary finite bitlength, $\mathcal{H}(x)$ maps to an output y of fixed bitlength n.

One-wayness. If $y = \mathcal{H}(x)$ is given, it is computationally infeasible to compute x.

Collision-freeness. It is computationally infeasible to find a pair (x, x') satisfying $\mathcal{H}(x) = \mathcal{H}(x')$.

Efficiency. Given an input x, $\mathcal{H}(x)$ is easy to compute.

Hash functions should be resistant against *Birthday* attack [36, 40], which is a powerful method to find colliding input pairs. So it is preferable that the output length of hash is longer than 160 bits under current

computing environments. We assume that hash functions in our protocols are secure and satisfy all above requirements.

The details on collision-freeness and one-wayness of hash functions are appeared in [7].

Hash Chain

Hash chain is a variant of hash functions and utilized in various areas: authentication[17], micropayment[28] and auction[33], *etc.*

The generation of hash chain is done as follows:

$$\begin{aligned} \text{Seed}: &\quad s_0 \\ \text{1st round}: &\quad \mathcal{H}^1 = \mathcal{H}(s_0) \\ \text{2nd round}: &\quad \mathcal{H}^2 = \mathcal{H}(\mathcal{H}(s_0)) \\ \ldots: &\quad \ldots \\ \text{n-th round}: &\quad \mathcal{H}^n = \mathcal{H}(\mathcal{H}^{n-1}). \end{aligned}$$

The use of hash chain values is in reverse order, *i.e.* from $\mathcal{H}^n$ to $\mathcal{H}^1$. From the one-wayness of hash functions, no one can predict the next value from the current value except only one has the knowledge on the seed.

2.2 Related works

2.2.1 Mobile Payment System

Since the mobile device first introduced in the world, there has been rapid development of new functions, improvement of services, and the enhancement of the computing power of mobile devices make M-commerce more profitable and promising.

But, there is still antipathy from the public to buy products or services on-line and pay for them also on the Internet. The main problem is that almost all Internet users are aware of credit card fraud committed by hackers during transmission over the communication channel. Therefore, on-line mobile service provider has to figure out how to pay out without any dispute through every transactions and even, when disputes take place, how the system to resolve them without losing fairness. Similarly, mobile payments face with a number of problems from not only performance but also security points of view.

To the best of our knowledge, two approaches have been done to the mobile payment systems up to date. One is to exploit the properties of mobile agent to reduce the computational overhead by a customer. The other is to make use of mobile devices as an authentication tool.

By Mobile Agent.

Since mobile devices possess limited communicational and computational capacity, mobile payment systems proposed in [20, 29, 38] utilize the idea of "mobile agent" which has autonomity and migration capability. The schemes in [29, 38], employing mobile agent techniques, has accommodated SET[32] protocol in which several public key computations are followed for payment. On behalf of a customer, the mobile agent performs all processes necessary in SET with the customer's confidential data. Lee *et al.*[20] also made use of SET but combined it with *Millicent*[21] in order to make the micropayment system available in mobile devices.

However, the critical problem is the prevention against a malicious host. To execute a purchase transaction, payment agent has to bring the customer's confidential data to the designated host. A random symmetric key used to encrypt payment instruction is generated in the

merchant host. Thus, as all computations like encryption and signature verification are performed in the merchant host, we have to guarantee the correctness of computation and confidentiality of customer information which the mobile agent accompanies. Although a few methods like [19, 30] to protect the mobile agent have been initiated, they increase the total computation adopting additional functions such as proxy signature or proxy certificate. In fact, no known general solution exist against a malicious host.

By Mobile Device.

Instead of using the mobile agent, a different approach has been introduced to the financial industry for the mobile payment systems such as Paybox.net[25] in Germany and Mobilix[23] in Denmark, *etc.* They simply use the mobile device as an authentication tool to confirm customer's payment information and approval by sending secret short code(*e.g.* password) over the air.

Even though they support customer's identification through his/her mobile phone number, the session among whole participating parties such as the bank, the vendor and the customer should be kept online during the transaction to check the validity and fairness of the payment. Without customer's prompt confirmation, the transaction will not able to be completed. Furthermore, the inherent problem of the mobile networks, the existence of mobile gateway, can be exploited to obtain customer's private information like account number or password by an attacker.

2.2.2 Sealed-Bid Auction

On-line auction is an efficient method to buy and sell the items on the Internet. In the cryptographic literature, auction is also an at-

tractive topic for the researchers to design a secure and practical protocol employing cryptographic primitives. To date, many researchers have studied and published various and outstanding auction protocols [1, 3, 4, 14, 12, 15, 16, 18, 24, 26, 31, 33, 34]. As there are a variety of auction styles such as *English, Dutch, Sealed-Bid, Vickrey, and M+1, etc.*(refer to [2] for details) whose rules are quite different, each protocol has distinctive goals and decision strategies depending on its own style. Our target among the auction styles is to design*Sealed-Bid auction* in which a bidder commits his bid with which he is willing to pay on the items without disclosure of the bidding price then, after the bidding session, the auctioneer opens the received bids and declares the highest bid as the winning price and the winner who sent the highest bid.

Here, we limit our observation on the previous works to *Sealed-bid* auction only for simplicity.

Franklin and Reiter [10] presented a sealed-bid auction protocol based on threshold secret sharing of bidding price. Their scheme also used verifiable signature sharing to prohibit a bidder from repudiating but doesn't protect the privacy of losers and losing bidders. Naor, Pinkas and Sumner [24] proposed privacy preserving auction with secure function evaluation and proxy oblivious transfer, which is improved by Juels and Szydlo [14] to address security bleaches by introducing verifiable proxy oblivious transfer. However, the scheme is not publicly verifiable. Cachin [4] used homomorphic encryption with the Φ-hiding assumption and an oblivious third party. The main problem of this scheme is that it cannot resolve the winning price except the winner, and also doesn't support non-repudiation. Suzuki, Kobayashi and Morita [33] proposed an efficient scheme adopting distribution of hash chain results to auctioneers. But, for anonymity, each bidder should register to a registration center and get a suitable pseudonym from the

center. Kikuchi, Harkavy and Tygar[15] explored the property of polynomial interpolation. Interpolation with the winning polynomial results in the identity of the winner. However, their scheme cannot resolve tie-breaking and increases the number of auctioneers proportional to the possible bidding range.

$(M+1)$ auction schemes proposed by Abe and Suzuki [1] and Kikuchi [16] can be converted to fit in sealed-bid auction with small variation. However, Abe and Suzuki's scheme seems to be inefficient in a sense that it makes use of bidder's several proofs and mixing. Kikuchi's protocol addressed non-repudiation, but, as in [15], large number of auctioneers are necessary for the wide range of bidding prices.

III. Secure Mobile Payment System Keeping Low Computation in Mobile Devices

3.1 Requirements

We first look into the general security and performance limitations in the mobile environments more closely to recognize what we can or can't do with mobile devices then define requirements.

Security Limitations: Most severe problem in security of the mobile networks comes from the inherent property of using proxy, *i.e.* mobile gateway, to communicate. Conventional mobile communication such as WAP[37] use two-tier transport layer: user and proxy, proxy and service provider, because each tier follows different protocol, the proxy always plays as a converter in transferring messages to fit on each protocol thus it can get full information on the messages. Thus we need strong trust on that this proxy works properly in secure way to transfer data through the wireless networks. Furthermore, the proxy should not save data in transaction to its storage or memory and must guarantee that authority only can get access to manage it.

Another problem arises when public key cryptosystems, based on exponent computation, are used for the security and availability. For example, in order to assure security against cryptonalysis and various attacks, RSA must use 1024-bit modulus. This will be a big burden on mobile devices with respect to computation as they have limited re-

sources.

Performance Limitations : Although wireless businesses are gaining popularity more and more, there exist performance limitations comparing to desktop environment that severely restricting what we can do in the mobile payment systems. In terms of hardware devices, the widely acknowledged performance limitations[37] are: *low CPU power, less memory (ROM and RAM), restricted power consumption, small display, and a variety of different input devices, etc.*

The problem is how we can transform currently available applications in the wired environments into the wireless environments without degrading the underlying properties. To make more convenient use of mobile devices, the size of a mobile device is getting smaller. Therefore, the available resources in desktop environment are not equally given when designing or implementing in the mobile environment. Obviously, above limitations prohibit mobile devices from heavy computations to require for conventional electronic commerce applications like SET. Furthermore, the serious concern is how to work well with a lot of different mobile devices.

In addition, forwarding message requires reliable network bandwidth. But we also have network limitations with the mobile communications[37] such as: *less bandwidth, more latency, unstable connection, and high error rate.*

Following above discussions, we classifies the basic security and performance requirements[11] of the electronic payment systems that must be fulfilled in the mobile payment systems as follows:

Unforgeability. Only authorized entity can issue coins.

Double-spending prevention. One issued coin cannot be used more than once.

Efficiency. The system must be efficient in terms of storage, communication, and computation.

Although there are many other requirements from the viewpoint of security and availability like anonymity, atomicity, divisibility, unlinkability, and transferability, *etc.*, the requirements to be addressed rely on the specific situation where the payment system runs. In our contribution, we design a mobile payment system satisfy only the general requirements listed above as a first step.

As in the real life, there must be a way of controlling money. If anyone can mint money freely, it's totally unnecessary for us to secure the money because if you need money, you can make it whenever you want. In electronic payment, more attention should be paid for the fact that digital script can be easily copied. Therefore, only authorized entity (*i.e.* Bank) should issue money and the issued money should be hard to counterfeit.

In electronic payment system, there must a manner to prevent from spending the same money script more than once. If this happens, a lot of disputes will occur in the payment system and the bank can be cheated by the customer or the vendor. In fact, double spending is a weak forgery of the payment script. Preventing double spending is very important in the electronic payment system.

From the performance limitations described earlier, reducing computation and communication load without compromising other existent characteristics is necessary in the generic mobile environments. We also focus on providing better efficiency with basic security requirements that listed above in constructing payment systems.

3.2 Proposed Scheme

3.2.1 Our Model and Assumptions

There are various operational models for M-commerce composed of three entities; the customer(C), the vendor(V), and the bank(B). However we specify the connections among three entities for our scheme as Fig.3.1.

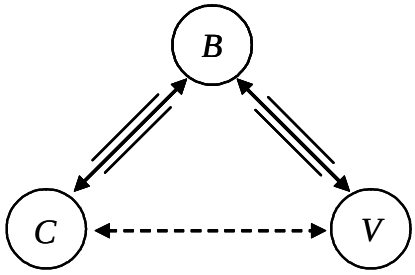

Figure 3.1: Operation model

Here, note that only the connection between the customer and the vendor(*i.e.* purchase) is set up through the wireless channel denoted as the dotted arrow. The customer has a mobile device like a mobile phone and the vendor provides corresponding mobile services.

When the customer or the vendor interacts with the bank, sensitive information such as account information or password and even money itself are transferred into the other end. In that case, a reliable and secure connection against passive and active attacks is mandatory. So, the other two connections(*i.e.* withdrawal and deposit) depicted as the solid arrows are assumed to be established through the *secure* wired channel by using the well-known security protocol like SSL[35]. In consequence, we mainly focus on the design of the mobile payment system which is secure against various attacks under the wireless network between the customer and the vendor.

Our approach on the payment is to utilize *off-line digital money* directly, which means the bank need not be *on-line* in the payment processing for goods under the above operational model. This is different from the two previous approaches where the bank should be always *on-line* to mediate the payment.

$\mathcal{H}$ is a collision resistant one-way hash function which takes an arbitrary string and outputs a uniformly distributed random string of a fixed size. Symbol $\|$ denotes concatenation of two strings. $\mathsf{DSig}_{\mathsf{X}}^{\mathsf{n}}(\cdot)$ is the digital signature generated by entity X_n on $(\cdot)$. These notations are used identically through this thesis.

3.2.2 Overview

In our scheme, we use a public key cryptosystem based on the discrete logarithm problem and a keyed hash function without adopting a mobile agent technique.

Our scheme consists of three protocols: **Withdrawal**, **Purchase**, and **Deposit**. Before describing the entire protocol, note that both Withdrawal and Deposit protocols are performed over a secure wired channel and only Purchase protocol is done through the mobile networks as stated before. Each protocol has the following main functions:

Withdrawal (C $\leftrightarrow$ B) : C requests setting-up for the mobile payment to B, who decides and publishes initial value observing the predetermined rule and issues the confirmation receipt on the request.

Purchase (C $\leftrightarrow$ V) : C constructs the payment script as much as the negotiated price of a product and pays it to V, who verifies the correctness of the payment script.

Deposit (V $\leftrightarrow$ B) : V asks deposit on the payment script received

from C then B reimburses the appropriate amount of money to V after verification.

Before our proposed scheme starts, we assume that two public keys of C, y_1 and y_2 have published in advance to the public key repository which permits anyone to access and retrieve the relevant information. The corresponding secret keys, x_1 and x_2 are preserved by C in secret.

3.2.3 Withdrawal Protocol

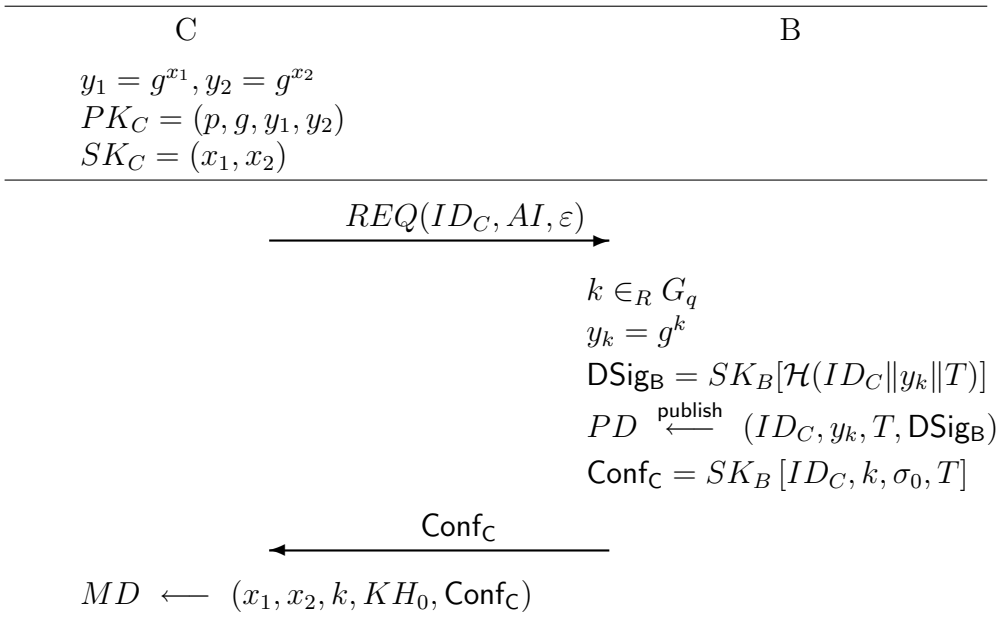

Figure 3.2: Withdrawal

C sends initiative request for setting up the mobile payment to B with his identity ID_C, account information AI and additional information ε to confirm C's account and identity. If C's account is good then B selects a random number k from the subgroup G_q and calculates y_k. B

associates k with C's account and keeps it in safe. This value will be used for the verification of the payment script from V in Deposit protocol and guarantee the prevention of double spending. B also decides the limitation of withdrawal σ_0 subject to the condition of C's account including some additional credit-loan. B publishes ID_C and y_k with T, issue and duration time of k, to PD so they become publicly accessible. Duration time is predetermined among participating entities. Whenever B releases the data to PD, for the integrity of PD, he generates digital signature $\mathsf{DSig_B}$ on the data and posts to PD to convince who gets the data.

B generates a confirmation receipt $\mathsf{Conf_C}$ on the received values using his secret key SK_B and returns $\mathsf{Conf_C}$ as the receipt to C through the secure wired channel. $\mathsf{Conf_C}$ makes B not to repudiate his agreement on the initiation of the mobile payment with C and the correctness of k. In fact, $\mathsf{Conf_C}$ is not used any more in other protocols. Its role is just an evidence about the generation of k when disputes happens among entities. C decrypts $\mathsf{Conf_C}$ with PK_B and checks the lifespan of committed k and the available amount of money σ_0. Then C stores securely his own secret keys x_1 and x_2, committed values k, initial keyed hash output $KH_0(=\mathcal{H}(k))$, and $\mathsf{Conf_C}$ to his mobile device.

For the security enhancement, we may employ smart card which has the tamper-proof property as an alternative to store those confidential data. In terms of anonymity, we can substitute ID_C for pseudonym which is issued by B, for instance, in the similar way that generates KH. B and C perform Withdrawal protocol whenever the following cases take place:

- The lifetime of k expires or the value of k is compromised.
- The summation of payments exceeds σ_0.
- Public key pair of B should be regenerated.

- C wants to alternate his account.

3.2.4 Purchase Protocol

C	V
$r \in_R G_q$	
$\mathcal{H}(ID) = \mathcal{H}(ID_C \| ID_V \| ID_B)$	
$KH_i = \mathcal{H}(k \| KH_{i-1})$	
$\alpha = x_1 + r$	
$\beta = k + KH_i$	
$\omega = r^{-1}\left[\sigma x_1 + x_2 + KH_i + \mathcal{H}(ID)\right]$	
$\xrightarrow{ID_C, ID_B, (\alpha, \beta, \omega)}$	
	Get y_1, y_2, y_k
	$P1)$ $\mathcal{H}(ID) = \mathcal{H}(ID_C \| ID_V \| ID_B)$
	$P2)$ $\frac{g^\alpha}{y_1} = g^r (= y_r)$
	$P3)$ $\frac{g^\beta}{y_k} = g^{KH_i}$
	$P4)$ $y_r^\omega \stackrel{?}{=} \left[\, y_1^\sigma y_2 g^{KH_i} g^{\mathcal{H}(ID)}\right]$

Figure 3.3: Purchase

This protocol is carried out between C and V over the wireless channel. Assume that the negotiated price of a product is σ which denotes the i-th payment of C for the sake of simplicity. For constructing payment script, C first picks up r at random from the subgroup G_q and hashes the concatenated identities of C, V, and B resulting in $\mathcal{H}(ID)$. Then C calculates keyed hash chain output KH_i that indicates the i-th payment and yields α, β, and ω as in Fig.3.3. C sends ID_C, ID_B, and (α, β, ω) as payment to V. By using two public keys and attaching a random number to the message, these messages reveal no information

on secrets x_1 and x_2. We will discuss this more later.

After receiving the payment script, V gets y_1 and y_2 from the public key repository and the public value y_k from PD using ID_C. V proceeds equations from $P1$ to $P4$ in order to verify the correctness of the received payment script. First, V computes $\mathcal{H}(ID)$ and then extracts $g^r (= y_r)$ from α and g^{KH_i} from β, using y_1 and y_k , respectively. V finally performs equation $P4$ to ascertain the payment script representing the price σ. If it passes, V keeps ID_C and g^{KH_i} for the period of lifespan of k to prevent double spending of the script. When V finds the same g^{KH_i}, by comparing with the stored values, he denies the payment script and closes the protocol. Thus C can't buy any goods of V with the same script due to the KH_i in β. The verification process, equation $P4$, could be justified as follows:

$$\begin{aligned} y_r^{\omega} &= g^{rr^{-1}[\sigma x_1 + x_2 + KH_i + \mathcal{H}(ID)]} \\ &= g^{\sigma x_1 + x_2 + KH_i + \mathcal{H}(ID)} \\ &= y_1^{\sigma} y_2 g^{KH_i} g^{\mathcal{H}(ID)}. \end{aligned}$$

If this fails, V may claims the payment script is invalid or forged and terminates the transaction. Notice that only the person who knows x_1, x_2, and k can compose legitimate messages α, β, and ω. In addition, since the hashed value of identities $\mathcal{H}(ID)$ is included in ω, even though an attacker intercepts the message, he cannot request deposit on the payment script and cannot insist that the script belongs to himself.

3.2.5 Deposit Protocol

Deposit protocol occurs between V and B through a secure wired channel. In order to get refund on the payment script, V needs to be authenticated by B that he is indeed the right merchant who sold his goods to C and received the valid payment script from the designated C. V

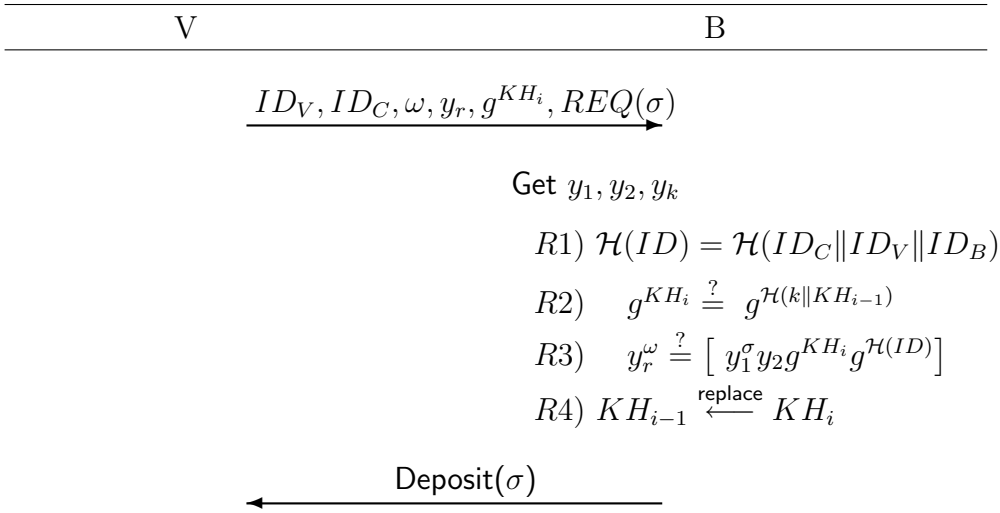

Figure 3.4: Deposit

sends the request tuple, (ID_V, ID_C, ω, y_r, g^{KH_i}, $REQ(\sigma)$), to B. B gets C's public keys (y_1, y_2) and retrieves k and KH_{i-1} corresponding to the ID_C's account which is maintained by B in safe.

B yields $\mathcal{H}(ID)$ and KH_i then confirms the transferred message through equations $R2$ and $R3$. First, B certifies the uniqueness of the script by equation $R2$ with calculated KH_i then checks the correctness of the payment script by equation $R4$. We omit justification of equation $R4$ because it is the same with $P4$ in Purchase protocol. If equations $R2$ and $R3$ are held, B replaces the value of KH_{i-1} with KH_i for the next verification. Finally, B reimburses the appropriate amount of money Deposit(σ) back to V. During the confirmation, if the accumulated of money spent by C exceeds σ_0, B eliminates g^k of C from PD then C must initiate Withdrawal protocol again to pay further payment.

3.3 Evaluation

We evaluate our scheme with respect to security and performance in detail.

3.3.1 Security

We may think three types of attack to forge the payment script due to the combination of entities: Vendor-only, Bank-only, and Vendor-Bank-together. Among them, the most powerful condition is the collusion of Vendor and Bank since they can obtain all messages during the transaction. So, we only consider this type of attack here.

Unforgeability.

To prove unforgeability of the payment script, we use the following lemma:

Lemma 3.3.1 *If solving the discrete logarithm problem is hard under a group, for the given pair* (g^x, μ) *where* x *is the secret key and* g^x *is the public key, finding random element* τ *of the group satisfying such that* $\mu = x + \tau$ *is equivalent to the difficulty of the discrete logarithm problem.*

Proof: It is straightforward to show the proof. Since we don't know the value of x from g^x by the intractability of DLP under a group which is solving DLP is hard in the polynomial time and τ is a random group element, the only way to get τ is to find the discrete logarithm of (g^μ / g^x) such that $g^\tau = (g^\mu / g^x)$. It leads to solve DLP again. □

In case of the collusion of V and B, β is meaningless from the view of B. So, we keep an eye on the values α and ω. V and B have knowledge on

σ, KH_i, and $H(ID)$ from the received messages and the keeping values k and KH_{i-1} to B. However, given α, they hardly obtain the random number r by Lemma 1. In addition, they don't know $(\sigma x_1 + x_2)$ as it is difficult to get x_1 and x_2 due to DLP. Without knowing r, it is impossible to compute the inverse r^{-1} which is multiplied to each term in ω. Therefore V and B cannot generate another payment script from the messages.

Here, the reason why we make use of two public key pairs to improve security becomes obvious. Assume that we use one public key pair (x_1, y_1). By changing of variable, x_1 into $(\alpha - r)$ in ω, B is able to get r^{-1} since B knows all other values; $\alpha, \omega, \sigma, KH_i,$ and $\mathcal{H}(ID)$. It leads to the disclosure of the secret key x_1, since it is simple to know r and x_1 from r^{-1}.

Although V and B have plural payment scripts on the same amount of money σ from the identical customer, they only can get the following values:

$$\begin{aligned} \alpha - \alpha' &= r - r'. \\ \omega - \omega' &= (r^{-1} - r'^{-1})(\sigma x_1 + x_2 + KH_i + \mathcal{H}(ID)). \end{aligned}$$

With the value of subtraction of α and α', they cannot induce r or r', which are chosen at random from the subgroup G_q, and cannot discover any helpful relationship between $(r - r')$ and $(r^{-1} - r'^{-1})$. The unique way to know both r and r' is to calculate the discrete logarithm of g^r or $g^{r'}$ on each.

As a result, there is no way to generate an eligible payment script by V and B without changing α and ω, even though V and B conspire together, as far as resolving DLP is intractable.

Double Spending.

The result of keyed hash chain KH_i makes sure the uniqueness of the current payment protocol run. Only the customer, who has k which comes from B in a secure way, can compute eligible KH_i by the underlying property of keyed hash function. So, when the same g^{KH_i} is returned to the bank, the bank can easily identify double spending. Furthermore, since only the owner of public key pairs y_1 and y_2 can make a legitimate payment script, the customer cannot repudiate double spending of the payment script when the bank informs the customer of the fact of double spending. One-wayness of hash function also makes an attacker is unable to find the pre-image of the result of hash.

From these discussions, we get the following theorem:

Theorem 3.3.2 *Our scheme satisfies the basic security requirements; unforgeability and double spending prevention, of the mobile payment system.*

Remark.

We are able to imagine various other attacks by the participating parties themselves to cheat other parties. Money forgery by any party is unattainable as described previously except the customer who is in the possession of secret keys (x_1, x_2).

The customer may try to send his same payment script to other vendors or even the same vendor to other goods. But both cases are not permitted. Since ID of the vendor who sold the goods to the customer is included in ω, the same payment script fails to the verification process by the other vendor in the former case. The latter case is also easily detected as the vendor is supposed to hold ID_C and g^{KH_i} during the lifespan of y_k that is certified by observing the bank's publication.

We might imagine another possibility of only substituting ID_V to other vendor's ID preserving the other values as the same by the customer. However this misbehavior will be detected by the bank from the index KH_i during Deposit protocol and the vendor who received invalid payment script will not be paid. In that case, the vendor can accuse the customer of cheating in a court of law or to an arbitrator with the payment script as an evidence.

We skip the details on security against the other possible attacks by the vendor or the bank since the proposed system can protect or solve with similar reasons above without losing confidentiality and fairness of the payment script.

3.3.2 Performance

We discuss the performance of our scheme in terms of computation and communication which are main interests in the mobile applications.

Computation.

In general, B and V are supposed to have enough powerful computational resources to execute several exponentiations. So we only take into consideration on the C's computational capability. Since Withdrawal protocol is carried out through the wired channel and C can use high-performance computation equipment like personal computer for withdrawal, one exponentiation to verify $\mathsf{Conf_C}$ is not expensive. During Purchase protocol, C executes only *two modular multiplications, one modular inversion, and two hashes* to constitute the payment script α, β and ω. As addition operation is negligible, we don't take into account on it. Furthermore, C need not participate in Deposit protocol again. As a result, the total computation, two modular multiplications, one modular inversion, and two hashes are so reasonable that we are

able to operate several times on mobile devices with little consumption of power and memory and less CPU power.

Communication.

Withdrawal and Deposit protocols are executed through the wired channel fewer than Purchase protocol. Thus let's consider only the communication taking place in Purchase protocol. Suppose that the size of each party's ID is 20 bit and q is 160 bit and SHA-1 is used for the hash. When a customer pays out to a vendor in Purchase protocol, the customer sends the following message to the vendor as shown in Fig.3.3:

$$ID_C, ID_B, (\alpha, \beta, \omega).$$

The total size of the message is

$$20 + 20 + (3 \times 160) = 520 \text{ bit},$$

since α, β, and ω are computed on modulo q. Message transfer from the customer to the vendor occurs only *once* during Purchase protocol. As a result, the total size, *520 bit*, is quite reasonable to transmit through the wireless networks with the limited bandwidth.

3.4 Summary

We suggested a secure and efficient mobile payment system that calculates only two modular multiplications, one modular inversion, and two hash computations, using two public key pairs and keyed hash function. Thus the customer(*i.e* mobile device) conserves low computation exempting from any modular exponent computation which is generally used in other electronic payment systems. Even one unilateral small message transfer from the customer to the vendor is sufficient to

complete payment. This characteristic is very effective in the wireless networks with the limited bandwidth. The security of our scheme is based on the intractability of the discrete logarithm problem and the one-wayness of hash function.

IV. Non-repudiable and Anonymous Sealed-bid Auctions

4.1 Requirements

From the previous researches, we have figured out there exist two problems which deteriorate the security and efficiency of the auction.

One is *to identify the winner explicitly by the auctioneer alone*. Otherwise, the winner can repudiate his bidding since he feels the winning price is too high to buy the items even if he casted at the winning price. In addition, a bidder can conspire with other bidders to decrease the winning price by not engaging in the winner identification. So, the auctioneer must have the ability to authenticate real or equivalent identity of the winner without assistance of him. Some works[15, 16, 33] treated non-repudiation as a mandatory requirement. But, [33] does not meet anonymity so that these protocols raise privacy problem. [15] cannot resolve tie-breaking which compromises non-repudiation. In others [1, 4, 14, 16, 18, 24, 31, 34], they seems to be anonymous in that only the indices of the winner are revealed to the auctioneer at the end of protocol. However, inevitably, the auctioneer must perform supplementary communications with the winner, namely who is placed in the winning indices, to confirm the fact that he committed the winning bid.

Definition 4.1.1 *Anonymous and non-repudiable auction* is the bid is committed to the auctioneer anonymously, however,the winner is explicitly identified without bidder's aid at the end of the auction.

The other problem is *to reduce the computational complexity to the size* $\log P$ in the winner resolution, where P is the number of possible

bidding prices. Abe and Suzuki have stated this issue as well in [1]. If the complexity of an auction protocol is proportional to $\log P$, the protocol is able to achieve much higher efficiency as the bid range increases. Naor, Pinkas and Sumner [24] introduced a protocol proportional to $\log P$ in a rough estimation, but a bidder's *on-line* communication load is very high to proceed bit by bit oblivious transfer.

As a result, in order for an auction protocol to provide both security and efficiency, we take into account the following requirements:

Privacy Losing bidders and bids should be kept in secret even to the auctioneer except the winning bid and the winner.

Anonymity No one can identify the bidder and the bidding price from a bid.

Non-repudiation The winner cannot repudiate his bidding at the winning price.

Publicly Verifiability Any one can verify the winning price and the winner which are decided correctly.

Fairness The protocol run is terminated in the predefined period and all accepted bids is dealt with in a fair way.

Walk-Awayness A bidder doesn't need to do any other action after bidding.

Efficiency The protocol should be efficient from the viewpoints of computation and communication.

Definition 4.1.2 If a sealed-bid auction accomplishes all requirements listed above, we say a *strong sealed-bid auction.*

4.2 Proposed Protocol 1

4.2.1 Our model and Assumptions

We focus ourselves to sealed-bid auction which can be modelled as consisting of three main phases: BID, Opening and Announcement. There are n bidders($= B_1, \ldots, B_n$), one master auctioneer(A_M), and m sub-auctioneers($= A_1, \ldots, A_m$). The role of master auctioneer is to organize each auction run and announces the bid result(*i.e.*, the winner and the winning price) at the end of the protocol run. He receives bids from bidders in a predetermined form (BID phase) and distributes the bids to m sub-auctioneers for the selection of the winning price and winner (Opening phase) then published the result(Announcement phase). The channels between master auctioneer and sub-auctioneers are supposed to be secure and reliable, which means all messages transferred between two entities finally reach to the communicating party without compromising. We suppose that auctioneers doesn't collude each other and misbehave. Our scheme is based on a public key cyrptosystem, namely every entity has its own secret key(SK) and public key(PK). Here, note that PK and SK of A_M are not static keys but ephemeral keys and he reveals SK after the bid. This assumption is given to provide publicly verifiability of our schemes. Bidding price(Bp) is denominated as a binary string of size m. Each sub-auctioneer represents each bit in bidding price, *i.e.*, bidders have 2^m possible bid choices. Notice that we allow a sub-auctioneer to know a little information such as bid statistics at his position.

$\mathcal{KG}$ is the key generation algorithm that takes a random string 1^k as an input, where k is a security parameter, and returns a *key* pair depending on the underlying encryption system.

4.2.2 Initialization

This protocol is based on RSA problem. n is a composite number subject to $n = pq$, where p and q are sufficiently large distinct primes and Euler phi function $\phi(n) = (p-1)(q-1)$. We suppose that all bidders and auctioneers have the key generation algorithm $\mathcal{KG}$. Master auctioneer calls $\mathcal{KG}$ and receives his RSA key tuple (n_A^M, e_A^M, d_A^M), where each term denotes a modulus, public and secret keys, respectively. $d_A^M (= SK)$ is kept in safe. m sub-auctioneers execute $\mathcal{KG}$ and each sub-auctioneer gets his own key tuple (n_A^j, e_A^j, d_A^j) on the condition that $n_A^M < n_A^1 < \ldots < n_A^m$. PK set of all auctioneers $\mathsf{PKS} = \{(n_A^M, e_A^M), (n_A^1, e_A^1), \ldots, (n_A^m, e_A^m)\}$ is announced in public by master auctioneer before the bid runs. Each bidder B_i obtains his RSA key tuple (n_B^i, e_B^i, d_B^i) from $\mathcal{KG}$, where $n_B^i < n_A^M$. PK of each bidder, (n_B^i, e_B^i), is publicly known. In order to protect our protocol against unexpected system failure in any sub-auctioneer, we may adopt key distribution method working in a threshold manner among auctioneers.

4.2.3 BID Phase

The i-th bidder B_i carries out the following steps to commit his bid:

B1. Gets PKS.

B2. Decides his bidding price $\mathsf{Bp} = (b_m \ldots b_1) \in_R \{0,1\}^m$.

B3. Executes the following B–Compute algorithm:

Algo. B–Compute $(\mathsf{PKS}, d_B^i, \mathsf{Bp})$
$EID \leftarrow [\mathcal{H}(ID_B^i \| Seq)]^{d_B^i} \mod n_B^i$
$\sigma_M^i \leftarrow [ID_B^i \| EID]^{e_A^M} \mod n_A^M$
$\sigma_0^i \leftarrow \sigma_M^i$
for $1 \le j \le m$

if $b_j = 1$ then

$\sigma_j^i \leftarrow \left[\sigma_{j-1}^i + 1\right]^{e_A^j} \bmod n_A^j$

$S_j^i \leftarrow \mathcal{H}(\sigma_{j-1}^i \| e_A^j)$

else

$\sigma_j^i \leftarrow \left[\sigma_{j-1}^i\right]^{e_A^j} \bmod n_A^j$

$S_j^i \leftarrow \mathcal{H}(\sigma_{j-1}^i)$

$S_M^i \leftarrow \mathcal{H}(\sigma_M^i \| \sigma_m^i \| Seq)$

Returns $(\sigma_m^i, S_M^i, S_m^i, \ldots, S_1^i)$.

B4. Sends $(\sigma_m^i, S_M^i, S_m^i, \ldots, S_1^i)$ to A_M.

Seq is the unique number of the participating auction, which works as a nonce to ensure uniqueness. Notice that b_1 is the least significant bit(LSB) in Bp and computed from LSB in B–Compute algorithm. Whenever a bit in Bp equals 1, PK of the sub-auctioneer at that place is powered to the previous result. S_j^i will play a role of indicator to verify whether or not the bid is committed at this bit. The communication between a bidder and master auctioneer occurs just *once* to transfer the encoded bid tuple.

4.2.4 Opening Phase

When the bidding time is over, master auctioneer closes the bidding session and collaborates with sub-auctioneers to resolve the winner and the winning price. Only auctioneers communicate according to the order in this phase. Observing the following steps, each sub-auctioneer A_j publishes the winning bit b_j^W(1 *or* 0) and transfers returned results to A_{j-1}. Running steps are as follows:

O1. A_M publishes all (σ_m^i, S_M^i) with $\mathsf{DSig}_A^M(\mathcal{H}(\sigma_m^1 \| S_M^1 \| \ldots \| \sigma_m^n \| S_M^n))$.

O2. A_M distributes all S_j^i to each A_j, for $1 \leq i \leq n$.

O3. A_j publishes all received S_j^i with $\mathsf{DSig}_A^j(\mathcal{H}(S_j^1 \| \ldots \| S_j^n))$.

O4. A_M transfers $(\beta_m^1 (= \sigma_m^1), \ldots, \beta_m^n (= \sigma_m^n))$ to A_m.

O5. From A_m to A_1, each sub-auctioneer A_j runs A–Resolution algorithm and forwards the returned value to the next sub-auctioneer A_{j-1} at the end of the algorithm:

Algo. A–Resolution $(d_A^j, (\beta_j^i, \ldots, \beta_j^n), (S_j^1, \ldots, S_j^n))$
for $1 \leq i \leq n$
$\beta_{j-1}^i \leftarrow ([\beta_j^i]^{d_A^j} - 1) \bmod n_A^j$
$S_j^{i'} \leftarrow \mathcal{H}(\beta_{j-1}^i \| e_A^j)$
if any $S_j^i = S_j^{i'}$ then announces $b_j^W = 1$
else $b_j^W = 0$
for $1 \leq i \leq n$
$\beta_{j-1}^i \leftarrow \beta_{j-1}^i + 1$
Publishes $\mathsf{DSig}_A^j(\mathcal{H}(\beta_{j-1}^1 \| \ldots \| \beta_{j-1}^n \| 1 \; or \; 0))$
Returns $(\beta_{j-1}^1, \ldots, \beta_{j-1}^n)$.

O6. A_1 transfers all β_0^i subject to $S_1^i = S_1^{i'}$, if $b_1 = 1$; otherwise, computes $S_1^{i'} \leftarrow \mathcal{H}(\beta_0^i)$ for $1 \leq i \leq n$, then sends all β_0^i subject to $S_1^i = S_1^{i'}$ to A_M.

Note that the order of opening is consecutive from the m-th sub-auctioneer to the 1st sub-auctioneer. In A–Resolution algorithm, the j-th sub-auctioneer first decrypts all bids and examines if at least one bidder bid at the j-th bit by matching $S_j^{i'}$ to S_j^i which is delivered from A_M in advance. Provided that bidders and auctioneers work correctly, at least one β_0^i that is sent to A_M will have the identical form of σ_M^i in BID phase. In terms of the signature scheme, several provably secure signature schemes (refer to [27] for details) could be a candidate to sign the message in our protocol. The last concatenated bit (1 *or* 0)

in DSig_A^j is the winning bit b_j^W announced by the j-th sub-auctioneer. DSig_A^j in A–Resolution algorithm will be an evidence when any disputes happens related to auctioneers's malfunctioning. We assume that all digital signatures are generated using the signer's static private key.

Note that a few trivial modifications and policies can enhance performance of our algorithm.

4.2.5 Announcement Phase

In this phase, A_M performs alone the following steps to officially announce the winning bid and authenticate the winner(s):

A1. Aggregates all winning bits announced by each sub-auctioneer, $\mathsf{Bp}^{Win} = (b_m^W \dots b_1^W)$.

A2. Decrypts all β_0^i received from the 1st sub-auctioneer, $\beta_M^i \leftarrow [\beta_0^i]^{d_A^M} \bmod n_A^M$, and extracts (ID_B^i, EID) from β_M^i.

A3. Gets PK of ID_B^i from the public key repository and computes $\beta_m^{i'}$ using β_M^i as σ_m^i in B–Compute algorithm taking PKS and Bp^{Win} as inputs.

A4. Verifies winner(s): $\mathcal{H}(ID_B^i \| Seq) \stackrel{?}{=} EID^{e_B^i} \bmod n_B^i$ and $S_M^i \stackrel{?}{=} \mathcal{H}(\beta_M^i \| \beta_m^{i'} \| Seq)$.

A5. Announces the winner(s) ID_B^W and Bp^{Win} with DSig_A^M on them and publishes his ephemeral secret key d_A^M together.

At A2, note that β_0^i is identical to σ_M^i in BID phase.

4.3 Proposed Protocol 2

4.3.1 Initialization

This protocol makes use of the intractability of discrete logarithm problem. Each bidder B_i has his static key tuple $(p, g, x_B^i, y_B^i (= g^{x_B^i}))$. A_M and every A_j execute $\mathcal{KG}$ and receive their ephemeral key tuple $(p, g, x_A^M, y_A^M (= g_1^{x_A^M}))$ and $(p, g, x_A^j, y_A^j (= g^{x_A^j}))$, respectively. x and y denote SK and PK, on each. These tuples match to the ElGamal encryption[8] setting, although we don't make use of ElGamal encryption through the paper. Here, another constraint is given in generating secret keys of all entities: $\gcd(SK, \phi(p)) = 1$. This policy should be embedded in $\mathcal{KG}$ beforehand. Note that auctioneers' keys are not static but ephemeral. $\mathsf{PKS} = \{y_A^M, y_A^m, \ldots, y_A^1\}$ and all PK of bidders, y_B^i, are published as in our protocol 1. We regard PK of bidder represents his identity assuming PK_B is certified by the certification authority(CA), *i.e.* $PK_B = ID_B$.

When the above initialization is completed, BID, Opening, and Announcement phases are almost same as in our protocol 1. Hence, we just describe main steps without further details. Note that some small variations in algorithms and steps are done.

4.3.2 BID Phase

In order to bid, a bidder B_i performs the following steps:

B1. Gets PKS.

B2. Decides his bidding price $\mathsf{Bp} = (b_m \ldots b_1) \in_R \{0, 1\}^m$.

B3. Executes the following B–Compute algorithm:

Algo. B–Compute $(\mathsf{PKS}, x_B^i, \mathsf{Bp})$

$a, b \in_R \mathbf{Z}_p^*$
$\tau_B^i \leftarrow (Seq + x_B^i)/a$
$\sigma_M^i \leftarrow \left[y_A^M\right]^a$
$\alpha_B^i \leftarrow g^b$
$\sigma_0^i \leftarrow \sigma_M^i$
for $1 \le j \le m$
  $\delta_j^i \leftarrow \left[y_A^j\right]^b$
  if $b_j = 1$ then
    $\sigma_j^i \leftarrow (\sigma_{j-1}^i + 1)\delta_j^i$
    $S_j^i \leftarrow \mathcal{H}(\sigma_{j-1}^i \| \delta_j^i)$
  else
    $\sigma_j^i \leftarrow \sigma_{j-1}^i \delta_j^i$
    $S_j^i \leftarrow \mathcal{H}(\sigma_{j-1}^i)$
$S_M^i \leftarrow \mathcal{H}(\sigma_M^i \| \sigma_m^i \| \tau_B^i \| \alpha_B^i \| Seq)$
Returns $(\sigma_m^i, \tau_B^i, \alpha_B^i, S_M^i, S_m^i, \ldots, S_1^i)$.

B4. Sends $(\sigma_m^i, \tau_B^i, \alpha_B^i, S_M^i, S_m^i, \ldots, S_1^i)$ to A_M.

4.3.3 Opening Protocol

O1. A_M publishes all $(\sigma_m^i, \tau_B^i, \alpha_B^i, S_M^i)$ with DSig_A^M $(\mathcal{H}$ $(\sigma_m^1$ $\|\tau_B^i$ $\|\alpha_B^i$ $\|S_M^1\| \ldots \|\sigma_m^n$ $\|\tau_B^n$ $\|\alpha_B^n \| S_M^n))$.

O2. A_M distributes all (S_j^i, α_B^i) to each A_j, where $1 \le i \le n$.

O3. A_j publishes all received (S_j^i, α_B^i) with DSig_A^j $(\mathcal{H}$ $(S_j^1\|$ $\alpha_B^1\|$ $\ldots \|$ $S_j^n\|$ $\alpha_B^n))$.

O4. A_M transfers $(\beta_m^1 (= \sigma_m^1), \ldots, \beta_m^n (= \sigma_m^n))$ to A_m.

O5. From A_m to A_1, each sub-auctioneer A_j runs A–Resolution algorithm and forwards the returned value to the next sub-auctioneer A_{j-1} at the end of the algorithm:

Algo. A–Resolution $(x_A^j, (\beta_j^1, \ldots, \beta_j^n), ((S_j^1, \alpha_B^1), \ldots, (S_j^n, \alpha_B^n)))$

for $1 \leq i \leq n$

$\lambda_j^i \leftarrow [\alpha_B^i]^{x_A^j}$

$\beta_{j-1}^i \leftarrow (\beta_j^i / \lambda_j^i) - 1$

$S_j^{i'} \leftarrow \mathcal{H}(\beta_{j-1}^i \| \lambda_j^i)$

if any $S_j^i = S_j^{i'}$ then announces $b_j^W = 1$

else $b_j^W = 0$

for $1 \leq i \leq n$

$\beta_{j-1}^i \leftarrow \beta_{j-1}^i + 1$

Publishes $(\lambda_j^1, \ldots, \lambda_j^n)$ and DSig_A^j $(\mathcal{H}(\beta_{j-1}^1 \| \lambda_j^1 \| \ldots \| \beta_{j-1}^n \| \lambda_j^n \| 1 \; or \; 0))$

Return $(\beta_{j-1}^1, \ldots, \beta_{j-1}^n)$.

O6. A_1 transfers all β_0^i subject to $S_1^i = S_1^{i'}$, if $b_1 = 1$; otherwise, computes $S_1^{i'} \leftarrow \mathcal{H}(\beta_0^i)$, where $1 \leq i \leq n$, and sends all β_0^i subject to $S_1^i = S_1^{i'}$ to A_M.

4.3.4 Announcement Protocol

A1. Aggregates all announced winning bits by each sub-auctioneer, $\mathsf{Bp}^{Win} = (b_m^W \ldots b_1^W)$.

A2. Computes $\beta_m^{i'}$ using β_M^i as σ_m^i in B–Compute algorithm taking $\mathsf{PKS}, \lambda_j^i$ and Bp^{Win} as inputs.

A3. Verifies winner(s): $S_M^i \stackrel{?}{=} \mathcal{H}(\beta_M^i \| \beta_m^{i'} \| \tau_B^i \| \alpha_B^i \| Seq)$.

A4. Announces the winner(s) $ID_B^W (= \left[\beta_M^{i\,(\tau_B^i / x_A^M)}\right] / g^{Seq} = g^{x_B^i})$ and Bp^{Win} with DSig_A^M on them and publishes his ephemeral secret key x_A^M together.

We claim that our two protocols can be extended to other public key cryptosystems with small modifications.

Remark 1. We may construct sub-auctioneers using tamper-proof devices which have $\mathcal{KG}$, $\mathcal{H}$, sufficient computation capability and ephemeral key agreement techniques. In that case, trusted third party(TTP) is responsible for the construction and supervision of these devices. Thus the computation and communication in sub-auctioneers can be done in secure and efficient way.

Remark 2. We can reduce the number of sub-auctioneers adopting some useful bid policies such as:

- A_j is in charge of 2 bits within P so he has 2 public key pairs. But, this policy has to sacrifice more information leakage such as bidding statistics on 2 bits to the auctioneer. This policy is preferable in non-critical auctions(for instance, in case that auction reveals bid statistics for the interest and motivation of participants).

- If we suppose the minimal bidding price is \$ 1,000(2^{10}, approximately), the maximal is \$1 million(2^{20}, approximately) and raised by \$ 1,000 then we are able to discard 10 bits from LSB of the maximal value. So 10 bits(=10 sub-auctioneers) are enough to bid up to \$ 1 million.

4.4 Evaluation

In this section, we give justifications on the requirements as stated before to show that our schemes are secure and efficient. We represents our proposed protocol 1 as P1 and protocol 2 as P2, respectively.

4.4.1 Security

PrivacyDuring Opening and Announcement phase, only winning σ_m^i takes off the exponent at the place of each auctioneer; otherwise, becomes a garbage value. So it is obvious that as far as at least one sub-auctioneer is honest, no one can learn any knowledge on losers' identities and bidding prices from the losing bids.

Anonymity Multiple encryptions with PK of sub-auctioneers in P1 randomize the bidder's identity. In P2, no information could be obtained from τ_B^i. Collision-free hash function makes sure no relationship between S_M^i and B_i as well.

Non-repudiation In Announcement phase of P1, the verification step A4 requires the winner's public key. This process seems to be equivalent to the verification of digital signature signed by the winner so that the winner cannot repudiate the fact of his bidding. In P2, non-repudiation is achieved by both the properties of hash function and Diffie-Hellman problem. Hashed value of correct $(\sigma_M^i, \sigma_m^i, \tau_B^i, \alpha_B^i, Seq)$ only maps to S_M^i and the one who knows x_B^i and a is able to compute τ_B^i.

From above security discussions, we can get:

Theorem 4.4.1 *Our proposed protocols,* P1 *and* P2 *are anonymous and non-repudiable sealed-bid auctions.*

Publicly Verifiability This is straightforward as SK of master auctioneer and Bp^{Win} would be opened.

Fairness By checking the issuing time of DSig_A^M issued by master auctioneer, fair termination can be observed. Fairness in the operations

done by sub-auctioneers could be observed from the correctness of DSig_A^j, Bp^{Win} and published values. Any loser is able to claim, if Bp^{Win} is lower than his bid.

Walk-Awayness This property is explicit in our protocols. The winner is identified by interactions among auctioneers only.

From above discussions and our embedded system setting, we can induce the following security for the bidding message:

Theorem 4.4.2 *As far as $\mathcal{KG}$ works with a sufficient security parameter as input and $\mathcal{H}$ generates collision-free outputs, bidding messages in both* P1 *and* P2 *are secure.*

Proof(Sketch). Our two key generation algorithms are based on RSA problem and DLP, respectively. We, in general, believe that an attacker bounded to the polynomial time has negligible probability($\leq \frac{1}{2^k}$, approximately) in solving those two hard problems given a sufficient security parameter. In addition, $\mathcal{H}$ with collision-free and one-way properties makes the attacker not to recover or find the original value from the output. So,we can conclude that our schemes are equivalent to those two hard problems and secure.

Security comparison with some sealed-bid auction protocols is shown in **Table 4.1**. For consistency, we consider 1st-price auction in [16] and method 2 in [33]. Note that [33] provides the privacy of bids not bidders.

4.4.2 Performance

Remind that P is the number of possible bid choices, n is the number of bidders and $m(= \log_2 P$, in our setting) is the number of auctioneers.

Table 4.1: Security comparison

	[16]	[24]	[33]	Our protocols
Privacy	△	O	△	O
Anonymity	O	O	X	O
Non-repudiation	X	X	O	O
Publicly Verifiability	X	X	O	O
Fairness	O	O	O	O
Walk-Awayness	X	X	O	O

We feel that $m = 20$(over than 1 million dollars) is enough for the general auctions.

Computation

In terms of computation, we ignore computation overhead of master auctioneer, since it is quite small and dominated by that of sub-auctioneers. A bidder's computation in Bp phase also is not expensive as it can be performed in *off-line* and computed with a little consumption of resources under the current computing power. Main computation overhead takes place in each sub-auctioneer to resolve the winner and the winning price: n RSA decryptions and hashes with two digital signatures(P1) and n modular exponentiations, n modular divisions and hashes with two digital signatures(P2). These computations dominate that in master auctioneer. Consequently, we can represent our computational complexity as asymptotically $\mathcal{O}(n \log_2 P)$ with small constant.

Table 4.2 shows performance comparison of main computation and communication overhead of the various auction protocols.

Table 4.2: Performance comparison

	Computation (Opening)	Communication ($B_i \rightarrow A$)
[16]	$\mathcal{O}(P)$	$(P+t)\mathsf{M}$
[24]	$\mathcal{O}(n \log_2 P)$	$(\log_2 P + 2)\mathsf{M}$
[33]	$\mathcal{O}(mnP)$	$m\mathsf{M}$
Our protocols	$\mathcal{O}(n \log_2 P)$	P1: $1\mathsf{M} + (\log_2 P + 1)\mathsf{H}$ P2: $3\mathsf{M} + (\log_2 P + 1)\mathsf{H}$

Communication

Only *one* transmission from B_i to A_M is enough to finish bid commitment. Each sub-auctioneer has *one* communication with A_M and A_{j-1}, individually, except that A_m and A_1 have one more with A_M.

In communication comparison of **Table 2** , M and H denote the output size of modulo operation and hash, respectively. For the sake of consistency, we set log operation with the base 2. In [16], t is is a number of faulty auctioneers. We concentrates on the communication, a bidder to an auctioneer only, since this is a main bottleneck in protocols. We regard all protocols are able to use a master auctioneer as a proxy to communicate with other auctioneers since it is more practical and rational. Notice that [16] and [33] should encrypt the transferring messages in that case. We claim that the sizes of message from B_i to A_M in our protocols are not serious under the current network environment.

From the the above security and performance discussions, we can get:

Theorem 4.4.3 *Our proposed protocols,* P1 *and* P2*, are strong sealed-*

bid auctions.

4.5 Summary

We proposed two secure and efficient sealed-bid auction protocols based on the intractability of RSA problem or DLP with collision-free and one-way hash function. One of main achievements is non-repudiation of winners keeping anonymity during the bidding phase. Another one is the computational complexity reduction to $\mathcal{O}(n \log_2 P)$ even if it's not the first scheme that works with this complexity, but in a different way. Furthermore, the inner communication and computation among auctioneers are not expensive. We believe that this low complexity makes our proposed protocols fit in a large scale auction with respect to both the number of bidders and possible choices.

V. Conclusions and Further Works

Through this thesis, we have studied on secure and efficient designs of E-commerce applications using cryptographic primitives. For the concrete design, we reviewed previous related works and pointed out their problems and weaknesses. And then we have suggested two improved protocols to address those problems and weaknesses considering other requirements.

Firstly, we have designed secure mobile payment system which can be easily adopted in the mobile environments. This is because the computation and communication loads in the customer side are reasonably low in our suggestion, but anyone neither can forge the payment script by DLP nor doubly spend by the characteristic of hash chain. Proposed mobile payment protocol requires only two modular multiplications, one modular inversion and two hashes to the customer. For the simple composition of the protocol, we employed two public key pairs and keyed hash function as basic cryptographic primitives.

Secondly, we have showed yet another strong sealed-bid auctions based on the intractability of RSA problem and DLP, respectively. Their main achievements are non-repudiation keeping anonymity of bidder and the computational complexity reduction to $\mathcal{O}(n \log_2 P)$. Iterative encryptions(P1) or modular multiplications(P2) hides a bidder's identity(*i.e* anonymity) and the winner's identity is revealed after winner decision processes by auctioneers(*i.e* non-repudiation). Furthermore, our protocols satisfies other basic requirements such as publicly verifiability and fairness.

Through our evaluations in terms of security and performance, we have explained that two proposed protocols are secure and efficient and

have more advantages compared to previous works. However, we have to consider more factors to use them in the real field. For example, suitable payment system should be combined for the auction protocols and digital signature schemes are chosen for both protocols.

As further works, it is necessary to prove our protocols' security in the sense of provable security. And real estimation of resource consumption and transaction time from implementation is meaningful to consolidate our efficiency.

In addition, it is valuable to find other solutions which reduce the computation and communication complexities preserving all requirements.

암호 기술을 이용한 안전하고 효율적인 전자 상거래 프로토콜 설계

함우석

컴퓨터 기술의 발달과 인터넷 보급의 확산은 전통적으로 오프라인(off-line) 형태로 행해져오던 은행, 우편, 관공서 업무 같은 처리들을 컴퓨터 네트워크를 중심으로 하는 온라인(on-line) 형태로 급속하게 변화시키고 있다. 더불어, 기업에서의 정보화(digitization)은 신 기술을 도입한다는 의미뿐만 아니라, 기업이 경쟁력을 갖기 위한 전략으로 등장하고 있다. 전자 상거래는 그 가운데서 가장 실생활에 밀접한 관련을 가지고 있으며, 신속한 발전을 보여주는 대표적인 예이다.

이러한 전자 상거래의 편리성과 효율성에도 불구하고, 전송 정보에 대한 안전성에 대한 불안감은 더 넓은 확산을 방해하는 요소로 작용하고 있다. 기본적으로 인터넷의 개방성과 성능과 보안 기능과의 절충(trade-offs)으로 오는 취약성은 중요한(confidential) 메시지들이 악의적인 공격 행위에 무방비로 노출되게 한다. 따라서, 전자 상거래 어플리케이션을 설계함에 있어 안전성을 보장하는 것은 필수적이다. 더불어, 설계된 프로토콜은 효율성을 함께 가져야 한다.

이러한 목적을 달성하기 위한 방법론으로 본 논문에서는, 안전하고 효율적인 이동 지불 시스템(mobile payment system)과 비공개 경매(sealed-bid aution) 프로토콜을 제안한다. 제안된 프로토콜들은 이산대수 문제(discrete logarithm problem)와 같은 정수론(number theory)적으로 어려운 문제들에 기반하고 있으며, 암호학적 해쉬(hash) 함수와 전자 서명(digital signature)를 주요 원천 기술로 사용하였다.

제안된 이동 지불 시스템은 두 개의 공개키(public key) 쌍과 키(key) 사용 해쉬 함수를 이용함으로써, 사용자는 대금 지불을 위해 한 번

의 모듈라(modular) 곱셈과 한 번의 모듈라 역원연산(inversion), 두 번의 해쉬 값 계산만을 수행하면 된다. 이러한 적은 연산은 제안된 프로토콜이 원활하게 이동 환경에 적용될 수 있도록 하며, 그럼에도 불구하고, 전자 지불 시스템의 기본적인 요구사항인 위조불가능성(unforgeability)와 이중 사용 방지(double spending prevention)을 만족 시킨다.

본 논문에서 제안된 두 개의 비공개 경매 프로토콜들은 RSA 문제와 이산대수 문제에 기반하고 있다. 제안된 경매 프로토콜들의 주요한 특성은 경매에 참여할 시에는 익명성(anonymity)을 보장하면서도 결정된 당첨자의 부인을 방지(non-repudiation)을 달성할 수 있다는 점이다. 더불어, 당첨자 결정을 위한 계산 복잡도를 $\mathcal{O}(n \log_2 P)$로 낮추었다. 여기서 n은 참여자의 수, P는 선택할 수 있는 경매 값들의 수를 나타낸다.

기본적인 프로토콜 소개와 함께, 제안된 프로토콜을 안전성과 성능면에서 분석을 실시하였으며, 유사한 다른 프로토콜과의 비교를 통해 제안된 프로토콜의 우수성을 보여준다.

References

1. M.Abe and K.Suzuki, M+1-st Price Auction Using Homomorphic Encryption, *Public Key Cryptography '02*, LNCS 2274, pp.115-124, Springer-Verlag, 2002.

2. C.Boyd and W.Mao, Security Issues for Electronic Auctions, Hewlett Packard, HP Technical Report HPL-2000-90, 2000.

3. O.Baudron and J.Stern Non-interactive Private Auctions, *Financial Cryptography '01*, LNCS 2339, pp.364-377, Springer-Verlag, 2001.

4. C.Cachin, Efficient Private Bidding and Autions with an Oblivious Third Party, *6th ACM conference on Computer and Communications Security*, pp.120-127, 1999.

5. D.Chaum, Blind Sigantures for Untraceable Payments,*Advances in Cryptology-Crypto '82*, pp.199-203, Plenum, 1983.

6. C. Kaufman, R. Perlman, and M. Speciner, Network Security PRIVATE Communication in a PUBLIC World, Prentice Hall, New Jersey, NJ, 1995.

7. I.B.Damgård, Collision Free Hash Functions and Public Key Signature Schemes, *Advances in Cryptology-Eurocrypt '87*, LNCS 0304, pp.203-216, Springer-Verlag, 1988.

8. T.Elgamal, A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, *Advances in Cryptology- Crypto '84*, LNCS 0196, pp.10-18, Springer-Verlag, 1985.

9. ERICSSON, Mobile Virtual Private Network, available at http://www.ericsson.com.

10. M.K.Franklin and M.K.Reiter, The Design and Implementation of a Secure Auction Service, *IEEE Trans. on Software Engineering*, Vo.22(5), pp.302-312, 1996.

11. H. Kim, M. Seo, J. Baek and K. Kim, Requirements and Comparision of the Existing Electronic Cash Protocols, *WISC '99*, pp.231-251, 1999.

12. M.Harkavy, J.Tygar and H.Kikuchi, Electronic Auctions with Private Bids, *3rd USENIX Workshop on Electronic Commerce*, pp.61-74, 1998.

13. NTT DoCoMo, available at http://www.nttdocomo.com.

14. A.Juels and M.Szydlo, A Two-Server, Sealed-Bid Auction Protocol, To appear in *Financial Cryptography '02*, Springer-Verlag, 2002.

15. H.Kikuchi, M.Harkavy and J.D.Tygar, Multi-round Anonymous Auction Protocols, *1st IEEE Workshop on Dependable and Real-Time E-Commerce Systems*, pp.62-69, 1998.

16. H.Kikuchi, (M+1)st-Price Auction Protocol, *Financial Cryptography '01*, LNCS 2339, pp.291-298, Spring-Verlag, 2001.

17. L.Lamport, Password Authentication with Insecure Communications, *Communications of ACM*, Vol.24, No.11, pp.770-772, 1981.

18. H.Lipmaa, N.Asokan and V.Niemi, Secure Vickrey Auctions without Threshold Trust, To appear in *Financial Cryptography '02*, Springer-Verlag, 2002.

19. B. Lee, H. Kim and K. Kim, Secure Mobile Agent using Strong Non-designated Proxy Signature, *ACISP '01*, LNCS 2119, pp.474-486, Springer-Verlag, 2001.

20. T. Lee, Y. Yip, C. Tsang, and K. Ng, An Agent-Based Micropayment System for E-Commerce, *E-commerce Agents*, LNAI 2033, pp.247–263, Springer-Verlag, 2001.

21. M.Manasse, The Millicent Protocol for Electronic Commerce, *The 1st USENIX Workshop on Electronic Commerce*, 1995. available at http://www.millicent.org/works/details/papers/mcentny.htm.

22. K.S.McCurley, The discrete logarithm problem, *Symposia in Applied Mahematics*, Vol.42, pp.49-74, American Mathematical Society, 1990.

23. MobiliX, available at http://mobilix.org.

24. M.Naor, B.Pinkas, and R.Summer, Privacy Preserving Auctions and Mechanism Design, *ACM conference on E-commerce*, pp.129-139, 1999.

25. Paybox.net, available at http://www.paybox.net.

26. K.Peng, C.Boyd, E.Dawson and K.Viswanathan, Robust, Privacy Protecting and Publicly Verifiable Sealed-Bid Auction, *ICICS '02*, LNCS 2513, pp.147-159, Springer-Verlag, 2002.

27. D.Pointcheval and J.Stern, Security Arguments for Digital Siganture and Blind Signatures, *Jouranl of Cryptology*, Vol.13, No.3, pp.361-396, 2000.

28. R.Rivest and A.Shamir, PayWord and MicroMint: Two Simple Micropayment Schemes, *International Workshop on Security Protocols*, LNCS 1189, pp.69-87, Springer-Verlag, 1996.

29. A.Romao, and M.Silva, An Agent-Based Secure Internet Payment System for Mobile Computing, *TREC '98*, LNCS 1402, pp.80-93, Springer-Verlag, 1998.

30. A.Romao, and M.Silva, Secure Mobile Agent Digital Signatures with Proxy Certificates, *E-Commerce Agents '01*, LNAI 2033, pp.206-200, Springer-Verlag, 2001.

31. K.Sako, An Auction Protocol Which Hides Bids of Losers, *Public Key Cryptography '00*, LNCS 1751, pp.422-432, Springer-Verlag, 2000.

32. The SET Standard Book 1 Business Description, SETCo, available at http://www.setco.org.

33. K.Suzuki, K.Kobayashi and H.Morita, Efficient Sealed-bid Auction using Hash Chain, *ICISC '00*, LNCS 2015, pp.183-191, Springer-Verlag, 2000.

34. S.G.Stubblebine and P.F.Syberson, Fair On-line Auctions Without Special Trusted Parties, *Financial Cryptography '99*, LNCS 1648, pp.230-240, Springer-Verlag, 1999.

35. Netscape Communications, The SSL Protocol, Version 3.0, available at http://wp.netscape.com/eng/ssl3/ssl-toc.html.

36. D.Wagner, A Generalized Birthday Problem, *Advances in Cryptology-Crypto '02*, LNCS 2442, pp.288-303, Springer-Verlag, 2002.

37. WAP Forum, available at http://www.wapforum.org.

38. X. F. Wang, K. Y. Lan, and X. Yi, Secure Agent-Meditated Mobile Payment, *PRIMA '98*, LNAI 1599, Springer-Verlag, pp 162–173, 1999.

39. K. Wrona, M. Schuba, and G. Zabagli, Mobile payments-State of Art and Open problems, *Welcome '01*, LNCS 2232, pp 88-100, Springer-Verlag, 2001.

40. G.Yuval, How to swindle Rabin, *Cryptologia*, Vol.3, pp.187-190, 1979.

Acknowledgement

First, I would like to express my sincere gratitude to Prof. Kwangjo Kim, my academic advisor, for his constant direction and support. He always has shown his consistent affection and encouragement for me to carry out my research and life in ICU. Special thanks also goes to Prof. Myungchul Kim and Dr. Choonsik Park for their generosity and agreeing to serve as committee members of my thesis.

I also would like to thanks to all members of cryptology and information security laboratory: Jongseong Kim, Hyunrok Lee, Hyungki Choi, Kyusuk Han, Byunggon Kim, Songwon Lee, Hwasun Chang, Jaehyrk Park, Soogil Choi, Juhyung Lee, Vo Duc Lim from Vietnam, Yan Xie and Zhang Fang Guo from Chaina, and Divyan from India, for giving me lots of interests and good advices during the course of my study.

In addition, I appreciate to the graduates, Manho Lee in KFTC and Myungsun Kim, for their everlasting guidance in life and study of ICU and I want to present my sincere gratitude to Jungyeon Lee of C Lab. and my fellow students Ingul Lee and Kihui Kim of Dongguk Univ.. It has been a precious time due to their being.

Most of all, I should mention my father and mother for their endless concerns and devotional affection. I cannot forget their trust and encouragement on me. My brother, his wife and my sister also have given me warmhearted concerns. I hope God bless my family and to be happy.

Finally, I will always remember the life of ICU. It filled up my poor knowledge and made me a grown-up person.

Curriculum Vitae

Name : Wooseok Ham

Date of Birth : Feb. 06. 1974

Sex : Male

Nationality : Korean

Education

1992.3–2000.2 Management Information System
Dongguk University (B.A.)

2001.2–2003.2 Cryptology and Information Security, Engineering
Information and Communications University (M.S.)

Career

2002.9–2002.12 Graduate Teaching Assistant
ICE615 Network Security
School of Engineering, ICU

2002.4–2002.12 Graduate Research Assistant
Research on Easy Security Technology
Electronics and Telecommunications Research Institute(ETRI)

2001.2–2002.7	Graduate Research Assistant Development of Electronic Voting System for World-Cup 2002 Information Research center for Information Security, ICU
2002.2–2002.6	Graduate Teaching Assistant ICE625 Computer Security School of Engineering, ICU
2002.1–	Graduate Research Assistant Middleware(8) Electronics and Telecommunications Research Institute(ETRI)
2001.7–2001.8	Apprentice Researcher Information Sharing Platform Laboratories(ISL), NTT, Japan
2001.2–	Graduate Research Assitant Cultivation of Top Level IT Security Manpower The Ministry of Information and Communications(MIC)
2001.2–2001.8	Graduate Research Assitant Study on Enabling Technologies for Next Generation Public Key Infrastructure SECUi.COM

Academic Experience

2002.4–	KIISC student member

Publications

(1) 2003.1 Wooseok Ham and Kwangjo Kim, Yet Another Strong Sealed-bid Auctions, *To appear in the 2003 Symposium on Cryptography and Information Security*, Hamamatsu, Japan.

(2) 2002.8 Wooseok Ham, Hyungki Choi, Yan Xie, Misung Lee and Kwangjo Kim, Secure One-way Mobile Payment System Keeping Low Computation in Mobile Devices, *The 3rd International Workshop on Information Security Applications*, pp.287-301, Jeju, Korea

(3) 2002.11 Wooseok Ham and Kwangjo Kim, A Secure On-line Lottery using Bank as a Notary, 2002년도 한국정보보호학회 학술대회, pp.219-224, 항공대학교, 한국

(4) 2002.11 함우석, 김종승, 이송원, 박재혁, 최수길, 김광조, 김숙연, 남택용, QoSS의 연구 동향과 적용, 2002년도 한국정보보호학회 학술대회, pp.352-355, 항공대학교, 한국