THEORETICAL EXAMINATION AND PRACTICAL IMPLEMENTATION ON CRYPTOGRAPHY ALGORITHMS, DIGITAL MONEY PROTOCOLS AND RELATED APPLICATIONS

By

Shek WONG

A Thesis

submitted in partial fulfillment of the requirements

for the Degree of Master of Philosophy

Division of Information Engineering

The Chinese University of Hong Kong

December 1997


香港中文大學
圖書館系統
16 JUL 1999
UNIVERSITY
THE CHINESE UNIVERSITY OF HONG KONG

Acknowledgement

I would like to express my sincere gratitude to my thoughtful advisor, Prof. Victor Keh-Wei Wei, who introduced me to the world of cryptography; devoted outstanding guidance and constant encouragement for keeping my perspective research objectives and motivation; helped me bring my research to fruition; provided me the best environment for research; and offered me a good job. Without his guidance and support, I would not be able to write this work and to learn so much invaluable and fruitful knowledge, which is crucial to my future development. Also, the "brain" exercises at the beginning of some of our discussion sessions stimulate the inspirations.

Special thanks to Prof. Raymond Wai-ho Yeung, who give me support and valuable suggestions for my further studies in this field. I have also been very happy and privileged to be affiliated to the Information Integrity Laboratory, both past and present. I formed a close friendship with all the members of this laboratory. I would like to thank all of them and many other friends of mine for their help and shouts of encouragement.

Finally, a heartfelt thanks to my parents, my sister, Ida and my loved, Phoebe. They provide me a nurturing and enriching home and always give me continued love, care and the best advice.

Abstract

In this report, two new fundamental concepts on untraceable divisible off-line electronic cash are proposed. The first concept is to allow each divisible electronic coin to bear a daily or monthly spending limit, which confines the spending of this particular coin to an amount not over the limit within one day or one month. The second concept is to impose an interest rate on the balance of each divisible electronic cash. These two concepts are also realized into two compatible and practical schemes.

Another research result I presented is about improvements of efficiency on storage and transmission of divisible untraceable off-line binary tree electronic coins. An abacus type electronic cash scheme is described and combinations of abacus type data structure with multi-tree based coins are discussed.

Besides attaining these results, a general introduction to electronic cash; a brief description of the terminologies and a collection of remarkable electronic cash schemes are presented. I appraise them to be some valuable contributions for further studies and researches. Finally, an overview on contemporary account card based Internet payment systems is given.

簡介

在當今的電子錢幣研究中，有一個分支叫做 “Untraceable Divisible Off-line Electronic Cash”。 在這個分支裏，本文提出了兩個全新的概念。第一個概念是讓每一個 “電子硬幣” (Electronic Coin) 在一天或一個月裏的使用不能超過其預定的限額。 第二個概念是要讓電子錢幣有利息。 而這兩個概念是用互相兼容的方法來實現的。

除此之外，本文還會介紹另一個研究結果。 它就是建立了一個新的電子錢幣的 “數據結構” (data structure)。 此結構叫 Abacus 。 這個結構可結合 “二叉樹” (binary tree) 或 “多叉樹” (multi-tree) 數據結構來達到電子錢幣的 “可除性” (divisibility)。

另外，本文還會對電子錢幣的發展做一個概述並分析它的結構。 同時也介紹一些重要的文獻以供參考。 最後，對一些流行於 Internet 上的銀行卡或信用卡的 “付款系統” (payment systems) 作一個簡介。

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Ever since the dawn of Mankind, people have been exchanging goods and services. During the late Pleistocene Age, Neanderthal men used their chipped stone tools to exchange animal flesh in barter. The obvious limitations of this kind of exchange led to the use of proxies for value. Shells and gems were used to perform the primitive functions of money as bearer instruments. Later on, pieces of precious metal came to be shaped into coins. The use of paper notes has also evolved with the development of printing. Economic historians often suggest that the switch from barter to the use of money allowed for more rapid economic growth of the man's history. We conclude all kinds of trades and business activities that are carried out in our society as *commerce*.

According to Miller [Mil88], money has four traditional functions: medium of exchange, unit of accounting, store of value and standard of deferred payment. Nowadays, human interactions are increasing at a good pace with the advancements of technologies. Our society is getting more complicated and divergent than ever before. As a natural extension, the world is now migrating into a new

era of 'cyber'. More and more information and valuable documents are handled in electronic forms. Many trades and business activities are also being carried out by exchanging of electronic signals. A new direction in the development of commerce has already been emerged. We called it - **Electronic Commerce**.

The development of electronic commerce relies heavily on the advancements of computer-based technologies. Computer-based technologies are mainly referred to the connections of thousands and millions of computers around the world. These globally connected computer networks are bringing new impact on our traditional society. *Cybernate* society has already been established which is furnished by inter-connected computers and advanced electronic equipment. Not only that, many types of *money* or payment means are also adopted. Examples of these new payment means include Electronic Cash, various kinds of credit card based Internet Payment Systems and Prepaid Smart Cards. These new payment means are different from our old acquaintances such as paper bank notes and nickel coins. They perform credit or debit via electronic signals rather than through the use of paper memo. Paper bills, currencies and checks are no longer involved any more.

To look into the future, new prospects are continually opening to this field without any sign of shrinking. More payment means are going to be divulged. Many of them may even become tomorrow's payment standards.

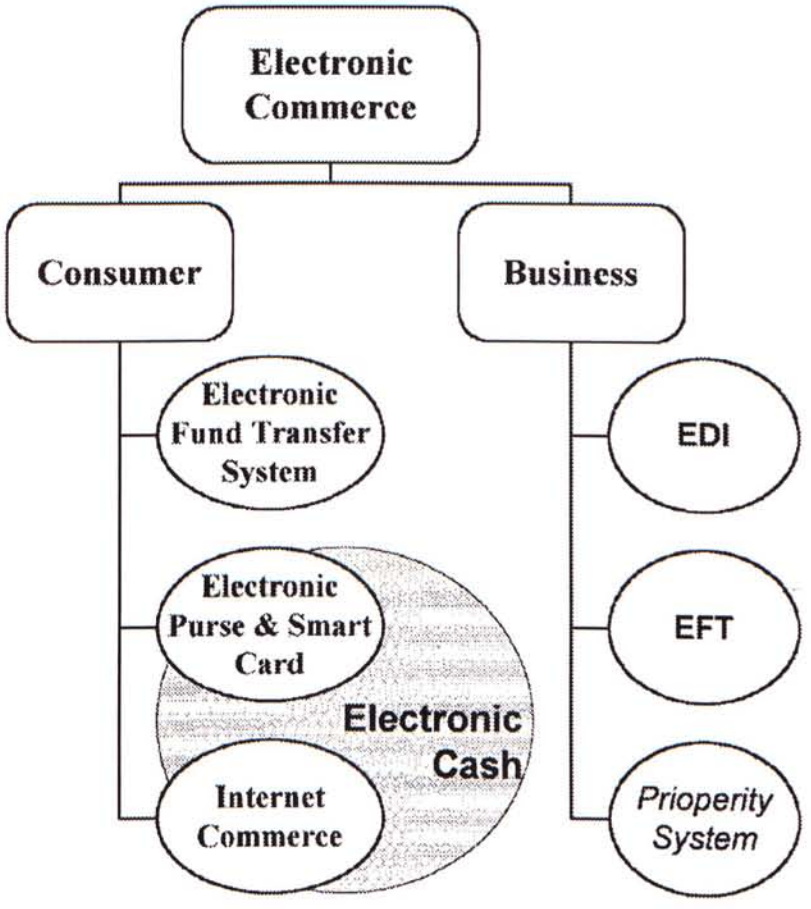

Figure 1.1: Components of Electronic Commerce

1.1 Electronic Commerce

A cashless society is emerging; electronic funds are traveling to-and-fro through computer networks and links of electronic equipment. Trades and business activities carried out in this environment are concluded as **Electronic Commerce**. There are several books [Kos97, For97, Lyn96, ECo] and online resources regarding this new emerging field can be consulted.

Figure 1.1 shows the major components which construct the electronic commerce. Basically, it can be divided into two aspects: **Business-to-business**

aspect and **Consumer** aspect. In business-to-business aspect, it is essential to work out standards and to let the involved companies to follow and to comply with. As early as 1960s, large business enterprises have already begun to conduct their business-to-business transactions using **Electronic data interchange** (EDI) [Kim91, MC93] on private computer network called **Value Added Network** (VAN). EDI is a matured general-purpose system which is standardized by the industries. EDI allows companies to exchange business documents in a standardized form and the documents are transmitted securely and efficiently on the private networks at most of the time. For the bank systems, they have been using dedicated networks for **electronic funds transfer** (EFT) almost as long. Furthermore, with the increased awareness and popularity of the Internet, these systems are no longer sticking on the pure private networks only. Actually, both EDI and EFT can be conducted over the Internet. Recently, EDI service providers are trying to incorporate Internet services in the existing EDI systems. Furthermore, new series of specifications for handling EDI on the Internet have already been begun to work out.

In consumer aspect, things are getting divergent. Not only divergent, but also inter-crossed. First of all, there are many **electronic funds transfer systems** being built up. Terms that we are familiar with including Electronic Banking, Phone Banking, ATM, etc. All of them are *checkless money transfer systems.* Money is transferred from one account to another by means of electronic signals. In this way, individual bank accounts are automatically credited and/or debited without any physical involvement of paper bills. These systems are the most matured ones in the consumer aspect of electronic commerce.

Next, we notice that there are various kinds of plastic cards in our pockets nowadays. Some are prepaid telephone cards and some others are common stored value transportation cards. You may also have some kinds of **Electronic Purses** such as Visa Cash[1] and Mondex[2]. These are all the payment means emerged from the advancements of smart card technologies. A smart card [ZO94], credit card sized plastic card, but has a chip embedded which equips hard-wired computing power and has storage capacity. Smart card architecture provides *physical security* to prevent forgery. It takes strict control of chip manufacturing, supplying, personalization and card issuing. Also, various protocols are required for conducting secure card transactions.

Besides electronic fund transfer systems and electronic purses, Internet Commerce is another important field which experiences remarkable development in recent years. Internet are becoming popular and *user-friendly* with the thanks to World Wide Web (WWW). More and more people today are able to browse the web sites on Internet easily for obtaining abundant information. More information people acquired, hungrier they are. In the meantime, business activities and trades are also carried out on Internet for cost effectiveness and flexibility. Therefore many *cyber* shops and on-line shopping malls have been built up on the Internet. Hundreds more of cyber-shop web sites are creating every day around the world. Those cyber-shops and shopping malls[3] provide a wide range of choices for their customers from daily commodities to luxury collections. In order to run business on Internet, we need appropriate payment means and some

[1] www.visa.com/cgi-bin/vee/pd/cash/main.html?2+0

[2] www.mondex.com

[3] Examples such as www.amazon.com, www.montywong.com, www.arkadia.com.hk, www.virtualvin.com, imsp001.netvigator.com/shopping/wellcom/index.html, www.llbean.com, etc.

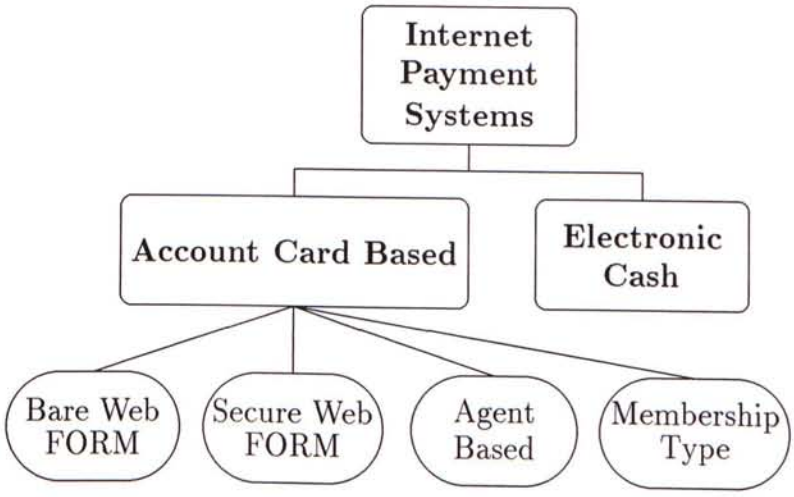

Figure 1.2: Various Internet Payment Systems

convenient payment systems. As this is a new industry, no single payment system has ever been dominating the market. Instead, a lot of different kinds of payment systems are built up to compete for a considerable market share.

Among the available Internet payment systems, two main approaches have got the leading positions: **Account Based approach** and **Electronic Cash approach** as shown in Figure 1.2. Account cards include credit cards and bankcards. One common characteristic of these cards is that a cardholder should first obtain an account. The account can be obtained from a bank, an agent or simply from a card issuer. During payment, money is withdrawn from the holder's account automatically, if it is a bankcard, or a proceed of loan to the card holder by the issuer, if it is a credit card. There are many other types of payment systems operating on the Internet. Since there isn't any standard to confine the protocols as well as the practices of the implementation of them, a

lot of different kinds of payment systems have been established. Even worse, almost all of these systems are different from each other in certain extent and they are also incompatible with each other. Consequently, cyber shops have to accept at least one or several of such systems as an interim or even a permanent solution.

In Figure 1.2, it also shows five main approaches of account card based payment systems. They are selected because all of them have gained certain extent of popularity on the Internet. In Appendix A, I will describe these payment systems in details for ease of reference.

Now, we stop stalking forward for a while and give a way back to think about if there is any relationship between electronic purse and Internet commerce. We notice that there are many remarkable achievements obtained on purse cards and there are also many innovative applications dug out at the same time. These have already brought a tremendous impact on Internet commerce. One of the examples is to use smart card to store the electronic coins. In this way, a user can use the withdrawn electronic coins to buy things from both traditional shops in the street and cyber shops on the Internet.

1.2 Electronic Cash

Finally, we still have one oval in Figure 1.1 which hasn't been described yet. This oval resides and overlaps between Internet Commerce and Electronic Purses. It's called **Electronic Cash**. Electronic cash is analogous to the conventional paper bank notes or nickel coins in the sense of certain properties. The difference is that electronic cash is used in electronic media.

Traditional physical cash has certain properties of an anonymous medium. However, its anonymous use is significantly constrained by some considerations. Firstly, it's bulky. Large amounts of money take up a certain amount of space. Secondly, it's palpable. Physical cash cannot be transferred over a computer network, and transferring it securely to a remote payee takes time and resources that may render the process somewhat visible. Thirdly, it's traceable. If law enforcement authorities know the serial numbers of bills being tracked, financial institutions may be able to help identify the next person who deposits them.

Electronic cash, on the other hand, outstrip the above issues for physical cash. Furthermore, it can strike a balance between anonymity and traceability. A system of electronic cash consists of "**Electronic Coins which are numbers**". This concept is first described by David Chaum in [Cha83]. Since the electronic coins are represented by just some strings of numbers, they can be stored in any kinds of physical media. For example, they can be stored in smart cards or computer storage media such as RAMs, hard drives, tapes, etc. Even broader perception, they can also be stored remotely at an agent's site on any connected communications network such as Internet. This concept stimulates new developments on maintaining the integrity and security of the electronic cash systems. It also raises research interests on improving the efficiency of such systems. Many cryptographic techniques have been employed and a handful of inspirational schemes have been proposed. In addition to that, it's also shown that the electronic commerce will outstrip our traditional commerce in the next ten years. In chapter 3, more detailed materials will be elaborated.

1.3 What This Report Contains

This thesis addresses the topic of electronic cash — An aspect dwells on employing various kinds of cryptographic techniques and schemes to realize and to conduct anonymous electronic payments in an environment of mutual mistrust among the bank and the system users.

In my studies, intensive readings and abundant survey of the state-of-the-art electronic cash schemes have been made throughout the whole research period. In order to get a panorama of electronic cash first, a general introduction to electronic cash is given in chapter 3. Afterwards, a brief description of the terminologies and a collection of remarkable schemes are described. In chapter 2, some cryptographic techniques and number theoretic properties are stated. To acquaint with the mathematical background is very useful for understanding the electronic cash schemes' cryptographic protocols and for conducting researches in this field.

In this report, I devote most of the efforts to present new research results obtained. With the crucial inspirations and involvement of Prof. Wei, we propose two new fundamental concepts on untraceable divisible off-line electronic cash in chapters 4 and 5. The first concept is to allow each divisible electronic coin to bear a daily or monthly spending limit. It confines the total amount spent of a particular coin to the specified limit within one day or one month. The second concept is to impose an interest rate on the balance of each divisible electronic coin.

Currently proposed electronic cash schemes are more or less the debit type. Electronic coins are minted by the bank and issued to the user after an equivalent

amount of money is deducted from the user's account. The user, on the other side, uses his coins to buy goods or to pay for services until all of them have been used up. There isn't any interest bore to the balance of electronic coins at any time. In our conceptual scheme, we will show how to impose an interest rate on the balance of electronic coins at the end of a certain period. For positive interest rate, the user's electronic coins are just the money stored in his bank savings account and interest will be added to the balance of electronic coins. There are many other applications you can think about by employing these two concepts. In section 5.6, I allot a few words to introduce some intuitional applications.

To realize these two new concepts, we use a *single term direct construction technique* at withdrawal protocol. This technique provides higher efficiency than that of the *cut-and-choose methodology* (section 2.13). Also, we employ a *secret sharing line technique* for detecting double-spending at payment protocol.

In chapter 6, another research result is presented. This result improves the efficiency on storage and transmission of divisible untraceable off-line binary tree electronic coins. An abacus type electronic cash scheme is described and combinations of abacus type data structure with multi-tree based coins are discussed.

In the annex, an overview on contemporary account card based Internet payment systems is also provided.

In this report, there are a handful of references and worthy resources that can be obtained on the Internet. To ease reading, I jot them down as footnotes with hypertext links and FTP site addresses wherever they are referred.

Chapter 2

Cryptographic Background

Cryptographic systems have been used for centuries by military and diplomatic organizations to keep messages secret and to authenticate the involved parties. Over the last decade, cryptology has been emerged from art to science and was incurred drastic evolution with the fast-growing global communications needs and commercial applications. In the field of electronic cash, we heavily employ dozens of cryptographic techniques and methodologies in order to attain the stringent requirements listed for an electronic cash system.

In this section, we'll review some points in number theory and some important cryptographic techniques which will be employed in the later parts of this thesis. Few related topics of cryptography and number theory will be covered in this chapter that are applicable to following chapters. Detailed mathematical text in number theory can be consulted from many textbooks. [Niv72] and [Sha78] are my favorite. Also, many cryptographic resources are collected in [MVOV97, Sch94, Sim92, Sti95, Tsi97, CLS94].

2.1 Euler Totient Function

If consider arithmetic modulo n, then a **reduced set of residues** is a subset of the complete set of residues modulo n which are relatively prime to n. The **Euler totient function**, also called the **Euler phi function** and written as $\phi(n)$, is the number of elements in the reduced set of residues. In short, $\phi(n)$ is the number of positive integers less than n that are relatively prime to n. This was first described by the mathematician Leonhard Euler (pronounced "Oiler") in the eighteenth century.

If n is prime, then $\phi(n) = n - 1$. If $n = p \times q$, where p and q are prime, then $\phi(n) = (p-1)(q-1)$. These numbers appear frequently in the public-key algorithms.

2.2 Fermat's Little Theorem

If m is prime, a is not a multiple of m, then **Fermat's little theorem** implies that

$$a^{m-1} \equiv 1 \pmod{m}. \tag{2.1}$$

2.3 Quadratic Residues

If p is prime and a is less than p, then a is a **quadratic residue** modulo p if

$$x^2 \equiv a \pmod{p}, \quad \text{for some } x. \tag{2.2}$$

There are exactly $(p-1)/2$ quadratic residues modulo p (denoted as QR_p) and the same number of quadratic nonresidues modulo p (denoted as QNR_p). Also,

if a is a quadratic residue modulo p, then a has exactly two square roots: one between 0 and $(p-1)/2$, and the other between $(p-1)/2$ and $(p-1)$. One of these two square roots is also a quadratic residue modulo p and it is called the **principle square root.**

If n is the product of two primes, p and q, there are exactly $(p-1)(q-1)/4$ quadratic residues modulo n.

2.4 Legendre Symbol

Definition Let p be an odd prime and a be an integer. The Legendre symbol $(\frac{a}{p})$ is defined to be

$$(\frac{a}{p}) = \begin{cases} 0, & \text{if } p|a, \\ 1, & \text{if } a \in QR_p, \\ -1, & \text{if } a \in QNR_p. \end{cases} \tag{2.3}$$

An easy way to calculate $(\frac{a}{p})$ is

$$(\frac{a}{p}) = a^{(p-1)/2} \bmod p. \tag{2.4}$$

Properties The properties of Legendre symbol can be summarized as below. Let p be an odd prime and $a, b \in \mathbb{Z}$.

(i) $(\frac{1}{p}) = 1$ and $(\frac{-1}{p}) = (-1)^{(p-1)/2}$. Hence $-1 \in QR_p$ if $p \equiv 1 \pmod 4$, and $-1 \in QNR_p$ if $p \equiv 3 \pmod 4$.

(ii) $(\frac{ab}{p}) = (\frac{a}{p})(\frac{b}{p})$. Hence if $a \in \mathbb{Z}_p^*$, then $(\frac{a^2}{p}) = 1$ and $(\frac{a^2b}{p}) = (\frac{b}{p})$.

(iii) If $a \equiv b \pmod p$, then $(\frac{a}{p}) = (\frac{b}{p})$.

(iv) $(\frac{2}{p}) = (-1)^{(p^2-1)/8}$. Hence $(\frac{2}{p}) = 1$ if $p \equiv 1$ or $7 \pmod 8$, and $(\frac{2}{p}) = -1$ if $p \equiv 3$ or $5 \pmod 8$.

(v) If q is an odd prime distinct from p, then

$$(\frac{p}{q}) = (\frac{q}{p})(-1)^{(p-1)(q-1)/4}$$

This is **law of quadratic reciprocity**. $(\frac{p}{q}) = (\frac{q}{p})$ unless both p and q are congruent to 3 modulo 4, in which case $(\frac{p}{q}) = -(\frac{q}{p})$.

2.5 Jacobi Symbol

The **Jacobi symbol** $(\frac{a}{n})$ is a generalization of the Legendre symbol for $n \geq 3$ with odd prime factorization $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$.

Definition It is defined to be:

$$(\frac{a}{n}) = (\frac{a}{p_1})^{e_1}(\frac{a}{p_2})^{e_2} \cdots (\frac{a}{p_k})^{e_k}$$

Properties Let $m \geq 3$, $n \geq 3$ be odd integers, and $a, b \in \mathbb{Z}$. Then, the Jacobi symbol has the following properties:

(i). $(\frac{a}{n}) = 0, 1,$ or -1. Moreover, $(\frac{a}{n}) = 0$ if and only if $\gcd(a, n) \neq 1$.

(ii). $(\frac{ab}{n}) = (\frac{a}{n})(\frac{b}{n})$. Hence if $a \in \mathbb{Z}_n^*$, then $(\frac{a^2}{n}) = 1$.

(iii). $(\frac{a}{mn}) = (\frac{a}{m})(\frac{a}{n})$.

(iv). $(\frac{1}{n}) = 1$.

(v). $(\frac{-1}{n}) = (-1)^{(n-1)/2}$. Hence $(\frac{-1}{n}) = 1$ if $n \equiv 1 \pmod 4$, and $(\frac{-1}{n}) = -1$ if $n \equiv 3 \pmod 4$.

(vi). $(\frac{2}{n}) = (-1)^{(n^2-1)/8}$. Hence $(\frac{2}{n}) = 1$ if $n \equiv 1$ or $7 \pmod 8$, and $(\frac{2}{n}) = -1$ if $n \equiv 3$ or $5 \pmod 8$.

(vii). $(\frac{m}{n}) = (\frac{n}{m})(-1)^{(m-1)(n-1)/4}$. In other words, $(\frac{m}{n}) = (\frac{n}{m})$ unless both m and n are congruent to 3 modulo 4, in which case $(\frac{m}{n}) = -(\frac{n}{m})$.

(viii). $(\frac{a}{b}) = (\frac{a \bmod b}{b})$.

Note that $(\frac{a}{n})$ *does not* reveal whether or not a is a quadratic residue modulo n. It is indeed that if $a \in QR_n$, then $(\frac{a}{n}) = 1$. However, if $(\frac{a}{n}) = 1$, it is not necessary that $a \in QR_n$. Let $J_n = \{a \in \mathbb{Z}_n^* \mid (\frac{a}{n}) = 1\}$. The set of **pseudosquares** modulo n, denoted $\overline{QR_n}$, is defined to be the set $J_n - QR_n$.

For a specific dwelling on, let p and q are odd primes, $n = p \times q$ and $z \in \mathbb{Z}_n^*$. Then, z is a quadratic residue modulo n if and only if $z \in QR_p$ and $z \in QR_q$ (use Legendre symbol to check them).

$|QR_n| = |\overline{QR_n}| = (p-1)(q-1)/4$; that is, half of the elements in J_n are quadratic residues and the other half are pseudosquares.

Now, if $z \in QR_n$, there are exactly 4 square roots of the form $\{x, n-x, y, n-y\}$ in $\mathbb{Z}_n^*$. And one square root is also a quadratic residue and is called **principle square root** of z modulo n.

If $z \in QR_n$ and p, q are known, then the square roots of z modulo n can be found in probabilistic polynomial time. Refer to **§3.5 Computing square roots** in $\mathbb{Z}_n$ of [MVOV97] for details.

$(\frac{a}{p})$	$(\frac{a}{q})$	$\mathbb{Z}_{(i,j)}$	$(\frac{a}{n})$
1	1	$\mathbb{Z}_{(1,1)}$	1
1	-1	$\mathbb{Z}_{(1,-1)}$	-1
-1	1	$\mathbb{Z}_{(-1,1)}$	-1
-1	-1	$\mathbb{Z}_{(-1,-1)}$	1

Table 2.1: 4 Classes for $a \in \mathbb{Z}_n^*$

2.6 Blum Integer

Definition If p and q are two primes, and both are congruent to 3 modulo 4, then $n = p \times q$ is called a **Blum integer**.

If n is a Blum integer, and let $a \in QR_n$, a has exactly four square roots, one of which is also a square (i.e. exactly one of which is also in QR_n); this is the **principle square root**.

Properties Let there is a Blum integer $n = pq$ where p and q are distinct primes each congruent to 3 modulo 4.

(i) $(\frac{-1}{n}) = (\frac{-1}{p})(\frac{-1}{q}) = 1$.

(ii) $(\frac{-x}{n}) = (\frac{x}{n})$ since $(\frac{-x}{p}) = -(\frac{x}{p})$ and $(\frac{-x}{q}) = -(\frac{x}{q})$.

(iii) If $x, y \in \mathbb{Z}_n^*$ and $x^2 \equiv y^2 \pmod{n}$, $\pm x, \pm y$ are 4 square roots where $|\pm x| \neq |\pm y|$. Also, $(\frac{x}{n}) = -(\frac{y}{n})$.

(iv) If $a \in \mathbb{Z}_n^*$, $\mathbb{Z}_n^*$ can be classified into 4 classes as shown in Table 2.1 Clearly, only $\mathbb{Z}_{(1,1)}$ denotes the set of quadratic residue integers in $\mathbb{Z}_n^*$. We can denote $\mathbb{Z}_{(1,1)}$ as QR_n and denote the other 3 classes as QNR_n.

(v) If $z \in QR_n$, z has exactly one square root in each $\mathbf{Z}_{(i,j)}$ for $i = \{0,1\}$ and $j = \{0,1\}$. These 4 square roots are relatively prime square (mod pq) as $[a,b]$, $[-a,-b]$, $[a,-b]$ and $[-a,b]$ where a is a quadratic residue of p and b is a quadratic residue of q.

Extended Properties

(i) For $n = p \times q$ where p and q are primes each congruent to 3 modulo 4. Let $x \in QR_n$. Then, for any integer t $(1 \leq t)$, there are exactly 4 square roots of x — y_1, y_2, y_3 and y_4 such that

$$y_i^{2^t} \equiv x \pmod{n}$$

(ii) $y_1 \in \mathbb{Z}_{(1,1)}$, $y_2 \in \mathbb{Z}_{(1,-1)}$, $y_3 \in \mathbb{Z}_{(-1,1)}$ and $y_4 \in \mathbb{Z}_{(-1,-1)}$.

(iii) $y_1 \equiv -y_4 \pmod{n}$, and $y_2 \equiv -y_3 \pmod{n}$.

(iv) $(\frac{y_1}{n}) = (\frac{y_4}{n}) = 1$ and $(\frac{y_2}{n}) = (\frac{y_3}{n}) = -1$.

The above proposition immediately implies that four values y_1 to y_4 of 2^t-th root y of x can be uniquely determined by knowing the following two information:

1. one root yields whether $(\frac{y}{n}) = 1$ or -1, and
2. one root gives whether $y < n/2$ or not.

This is because when $y < n/2$, there are two values of y: one of which has $(\frac{y}{n}) = 1$ and the other has $(\frac{y}{n}) = -1$.

Also, to obtain these 4 roots ($x^{1/s^t} \pmod{n}$) can be *efficient* from x, p and q in expected polynomial time. Details of the solving algorithm can be referred to

§3.44 Algorithm of [MVOV97]. However, if you don't know the prime factors p and q of n, to compute the 4 square roots solely from x and n is as difficult as factoring n.

2.7 Williams Integer

Definition A **Williams integer** [Wil80] is a composite integer of the form $n = p \times q$ where p and q are distinct primes where $p \equiv 3 \pmod 8$ and $q \equiv 7 \pmod 8$.

Properties From the definition, it can be derived that $n \equiv 5 \pmod 8$. Also, the Williams integer is a specific type of Blum integer. So, Williams integer has *all properties* of the Blum integer.

Properties include:

(i) $(\frac{-1}{p}) = -1$, $(\frac{-1}{q}) = -1$, $(\frac{2}{p}) = -1$, and $(\frac{2}{q}) = 1$.

(ii) $(\frac{2}{n}) = -1$.

(iii) $(\frac{-1}{n}) = (\frac{-1}{p})(\frac{-1}{q}) = 1$.

(iv) For any $x \in \mathbb{Z}_n^*$, either **one** of $x, -x, 2x$ or $-2x$ is in QR_n. Any relatively prime residue x (i.e. $x \in \mathbb{Z}_n^*$) will be of type 1: ($\mathbb{Z}_{(1,1)}$ or $\mathbb{Z}_{(-1,-1)}$) if and only if $2x$ is not.

(v) When $ax \in QR_n$ where a is either 1, -1, 2 or -2, then $bx \in QNR_n$ for $b \neq a$, and b is either 1, -1, 2 or -2.

From the above properties, it is easy for us to find a value which belongs to QR_n. The procedure is as below:

a) choose any $x \in \mathbb{Z}_n^*$,

b) find out the value of a for $a = 1, -1, 2$ or -2 such that $(\frac{ax}{p}) = 1$ and $(\frac{ax}{q}) = 1$.

Then, this $ax \in QR_n$. This is easy provided that we know p and q.

However, if we only know x and n without obtaining p and q, then we can only know that

1. whether $(\frac{x}{n})$ is equal to 1, and

2. if $(\frac{x}{n}) = 1$, then x is in type 1: ($\mathbb{Z}_{(1,1)}$ or $\mathbb{Z}_{(-1,-1)}$).

Thus, we have $1/2$ chance to guess correctly that whether x belongs to QR_n or not.

Quadratic Residues, Legendre Symbol, Jacobi Symbol, Blum Integer and Williams Integer are useful for reading chapter 6.

2.8 The Quadratic Residuosity Problem

Definition The Quadratic Residuosity Problem (QRP) is defined as given an odd composite integer n and $a \in J_n$, decide whether or not a is a quadratic residue modulo n.

If n is prime, it is easy to solve by checking with Legendre symbol that if $(\frac{a}{n}) = 1$. If n is a product of 2 primes: p and q, then $a \in QR_n$ if and only if $a \in QR_p$ and $a \in QR_q$. However, if the factorization of n is unknown, then there is no efficient procedure known for solving QRP, other than guessing the answer. If $n = pq$, then the probability of a correct guess is $\frac{1}{2}$ since $|QR_n| = |\overline{QR}_n|$. It is believed that QRP is as difficult as the problem of factoring integers.

2.9 The Factorization Problem

Definition Given a positive integer n, find its prime factorization; that is write $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ where the p_is are pairwise distinct primes and each $e_i \geq 1$.

The factoring problem is one of the oldest in number theory and the difficulty of factoring large numbers are employed to the security of many cryptographic algorithms. Protocols that are based on the difficulty of factoring usually require that the prime factors are at least 512 bits and preferably larger than 1024 bits[1], since factoring a lower order of the numbers is well within the reach of today's computing machinery. For textbook treatment of efficient factoring algorithms, see [Rie85] and [MVOV97].

2.10 The Discrete Logarithm Problem

Definition 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 $\beta = \alpha^x$. This can be easily generalized to the problem in any finite cyclic group G [MVOV97].

It is a hard problem to obtain the discrete logarithm of β when the order is large. In particular, there is no known polynomial-time algorithm for the discrete logarithm problem (DLP). Methods for solving the DLP in one group do not necessarily apply in other groups. Therefore, the difficulty of DLP depends upon the group used for arithmetic.

The security of many cryptosystems depends on the intractability of the discrete logarithm problem in a multiplicative group $\mathbb{Z}_p^*$. Examples include El-Gamal encryption and signature scheme [ElG85], Digital Signature Algorithm

[1] www.rsa.com/rsalabs/newfaq/q48.html

(DSA)[2] and Diffie-Hellman key agreement [DH76]. There are some other public-key cryptosystems whose security rests on discrete logarithm problem in elliptic curve groups (ECDLP) [Mil86, MV90]. In these papers, no new cryptographic algorithm was invented, but they implemented existing public-key algorithms, such as Diffie-Hellman and ElGamal, in elliptic curves over finite fields.

An Elliptic Curve Cryptosystem (ECC) is elliptic curve analogues of cryptosystems based on the DLP in $\mathbb{Z}_p^*$. This analogue cryptosystem is just like its counterparts in $\mathbb{Z}_p^*$, but with all computations performed in an elliptic curve group instead of in $\mathbb{Z}_p^*$.

ECDLP appears to be more difficult to solve than in the group $\mathbb{Z}_p^*$ (DLP). Detailed description and implementation analysis can be consulted from [Men93].

2.11 One-way Functions

Definition Given x, it is *easy* to compute $f(x)$; but given $f(x)$, it is *computationally infeasible* to find x.

Basically, it is an invertible function. Also, the above definition can be made more precise by requiring *easy* to mean computable in polynomial time while *computationally infeasible* to mean requiring exponential time. We refer *computationally infeasible* to a problem which would take millions of years to compute even if all the computers in the world were assigned to it. At present, it is not known whether one-way functions exist, although there are several candidate one-way functions.

[2]NIST FIPS 186 (csrc.nist.gov/fips/fips186.ps)

A **trap-door one-way function** is a one-way function with additional property that given some extra information called *trapdoor*, it becomes feasible to find x from given $f(x)$.

2.12 Blind Signature

We are quite familiar with the term **Digital Signature** (refer to [Sch94] **§2.6 Digital Signatures**). One main feature holds for all digital signature protocols is that the signers *always* know what they are signing. On the contrary, there are times when we want a **signer** to sign a document for a **sender** but prevent the signer to observe the message it signs. Hence, it is later unable to associate the signed message with the sender. This technique is dreadfully used in electronic cash scheme to provide user anonymity since the introduction by David Chaum in [Cha83].

The scheme is established based on the assumption of existence of a commutative style **public key system**. Also, an untraceable payment system was built as an application by the presented blind signature technique. Chaum further developed blind signature protocols in [Cha88] based on RSA [RSA78]. Many other references including [Cha83, Cha85, Cha88] are worthy to be consulted. Another typical application of blind signature technique on untraceable electronic cash system could later be found in [CFN90]. An online literature [Cha92][3] can also be found.

The concept of a blind signature can be illustrated by an example taken from the familiar world of paper documents. The paper analog of a blind signature

[3]http://ganges.cs.tcd.ie/mepeirce/Project/Chaum/sciam.html

can be implemented with carbon paper lined **envelopes**. Writing a signature on the outside of such an envelope leaves a carbon copy of the signature on a slip of paper within the envelope. This example is first referred by David Chaum in [Cha85].

About implementation, the process consists of three basic transformations:

1) **blinding** by the sender,

2) **signing** by the signer, and

3) **unblinding** by the sender.

Cite Chaum's blind signature protocol as an implementation example, we assume there is a sender, Alice requests a signature of signer, Bob on a blinded message, m. It uses the RSA algorithm. Let B has a public key, e, a private key, d, and a public modulus, n.

Step 1 Alice randomly chooses a value k called **blinding factor** which satisfies $1 < k < n$ and $\gcd(n, k) = 1$. Alice computes $m^* = k^e m \bmod n$ and sends this to Bob.

Step 2 Bob computes $s^* = (m^*)^d \bmod n$ and sends back to Alice.

Step 3 Alice computes $s = k^{-1}s^* \bmod n$ and the result should be $s = m^d \bmod n$. This is Bob's signature on m.

The protocol described above is called **Completely Blind Signature** and can only be operated under a mutual trust environment.

Problem In an application of electronic cash scheme, a customer requests the bank to blindly sign a number of his coin, which is defined to denote a million dollars, but the customer claims that it's the number for denoting 10 dollars only. Since the bank cannot observe the number and doesn't sure whether the number is the claimed one, the customer can play such trick successfully.

The possibilities are endless and the completely blind signature protocol isn't particularly useful in the field of electronic cash. To provide solution against this problem, we need other techniques to incorporate with the blind signature protocol. One of those techniques is called **cut-and-choose methodology**.

2.13 Cut-and-choose Methodology

The technique, **cut-and-choose** is named because of its similarity to a classic protocol for dividing anything fairly:

1) Alice cuts a cake in half.

2) Bob chooses one of the halves for himself.

3) Alice takes the remaining half.

In many electronic cash scheme, cut-and-choose methodology is mainly used to solve the problem stated in last section. The heart of this methodology can be explained by the following example:

1) Alice prepares ten messages, each message denotes the same denomination that she wants to withdraw but with a different serial number inside it.

2) She blinds each of these messages with a different *blinding factor*. And she sends the blinded messages to the bank, Bob.

3) Bob *randomly* chooses nine blinded messages and asks Alice for the corresponding blinding factors.

4) Alice sends Bob the appropriate blinding factors for each of those chosen messages.

5) Bob removes the blinding factors from these nine messages and checks if the denominations are all consistent and correct. Also he makes sure that serial numbers are different from each other and do not exist in bank's database of used serial numbers.

6) Bob then signs the remaining unrevealed message and sends it back to Alice.

7) Alice removes the blinding factor and uses this signed coin for purchasing.

This protocol is secure against Alice cheating. For her to cheat, she would have to predict accurately which message Bob would not examine. The odds of her doing this are 1/10. To have a better feel of confidence that the odds of Alice's being able to find such a pair to be negligibly small, the number of messages that Alice sends to Bob can be increased, say 100 messages. Then, the possibility that Alice can cheat is down to 1/100.

Chapter 3

Anatomy and Panorama of Electronic Cash

3.1 Anatomy of Electronic Cash

The user of credit cards today is an act of faith on the part of all concerned. Each party is vulnerable to fraud by the others, and the cardholder in particular has no protection against surveillance. On Internet, the problem is even worse because no card is required to show up physically during purchasing. Purchasing by card is just a providing of card number and card expiration date on Internet. Therefore, security is one of the main issues of using credit card. In addition, plethora of Internet payment systems have been built based on account-base approach (Appendix A) but none of them can provide **privacy** — the way to keep buyer's identity secret from the merchants. If we look at our society, to keep user's anonymity is one of the significant advantages of the physical cash over credit cards. This advantage is also one of the main reasons to explain why

the physical cash is still an unbeatable mean of money even the society is rapidly migrating to a cashless one. Today, people need more freedom and more self-determining power. On the contrary, autocrats and large business enterprises are becoming more powerful than ever before. They are able to collect a lot of personal information of the people and make use to the information in many other aspects in which they are interested.

Consequently, privacy is progressively viewed as an attractive feature in economic transactions. Philosophically, the rights of individuals to privacy dictate it. On a more pragmatic side, massive compilation and misuse of personal data is likely to lead to a general understanding of the danger to privacy originating from the lack of anonymity. To give regards to these concerns, we need a payment mean which can protect the individuals' privacy from those "big brothers" during purchasing in electronic commerce just like what physical cash does for us today. This is also one of the basic criteria of constructing an electronic cash system. In simplicity, we can consider electronic cash is analogous to physical cash but it is an electronic form used electronically on computer communications networks.

Before going into a deep discussion of electronic cash, let's see what other denotations and nicknames that "electronic cash" has. There is an adage that the more up-to-date thing, the more likely it is to be changed. This is also reinforced when we try to give a name to this new type of money. Within these few years, a heap of synonyms come out for naming it. Early names were replaced by later refined names; and many more names are coexisting. Some of them are *electronic money*, *digital money*, *e-cash*, *digi-cash*, or some combinations of the words. Over here, I use **electronic cash** because its crucial feature, *user*

anonymity is analogous with physical cash, in otherwise, its "**coins which are numbers**" which travel through electronic links and computer networks. In addition, we called the coins or tokens in an electronic cash system as electronic coins.

3.1.1 Three Functions and Six Criteria

In its basic form, an electronic cash scheme [Cha85] is a set of cryptographic protocols for

Function 1 a **customer** to withdraw electronic coins from a **bank** or a **broker**;
(This function is carried out by *Withdrawal Protocol.* In some schemes [OO90, OO92, Oka95], an *Opening Protocol* is first invoked before any withdrawal protocols are being carried out.)

Function 2 the customer to use the money to purchase something from a **shop** or a **vendor**;
(This function is carried out by *Payment Protocol.*)

Function 3 the vendor to deposit the money in its bank account or its broker account.
(This function is carried out by *Deposit Protocol.*)

These three functions delineate the basic working flows of an electronic cash scheme and the protocols residing in, which protect the security interests of the three parties involved. To construct an ideal electronic cash scheme, the following six requirements [OO92] have to be fulfilled:

(a) *Independence* — The security of electronic cash cannot depend on any physical condition. Then the cash can be transferred through networks.

(b) *Security* — The ability to copy (reuse) and forge the cash must to be prevented.

(c) *Privacy (Untraceability)* — The privacy of the user should be protected. That is, the relationship between the user and his purchases must be untraceable by anyone.

(d) *Off-line payment* — When a user pays electronic cash to a shop, the procedure between the user and the shop should be executed in an off-line manner. That is, the shop does not need to be linked to the host in user's payment procedure.

(e) *Transferability* — The cash can be transferred to other users.

(f) *Divisibility* — One issued piece of cash worth value C (dollars) can be subdivided into many pieces such that each subdivided piece is worth any desired value less than C and the total value of all pieces is equivalent to C.

3.1.2 Untraceable

To further elaborate the six criteria in previous section, we can first consider the paper cash. Paper cash has a significant advantage over credit cards with respect to privacy as mentioned before. Although the serial numbers on cash make it traceable in principle, paper cash is nearly difficult to be traced in practice. This is called **conditionally untraceable**. Some electronic cash schemes proposed [Tsi97] can perform **unconditional untraceability**.

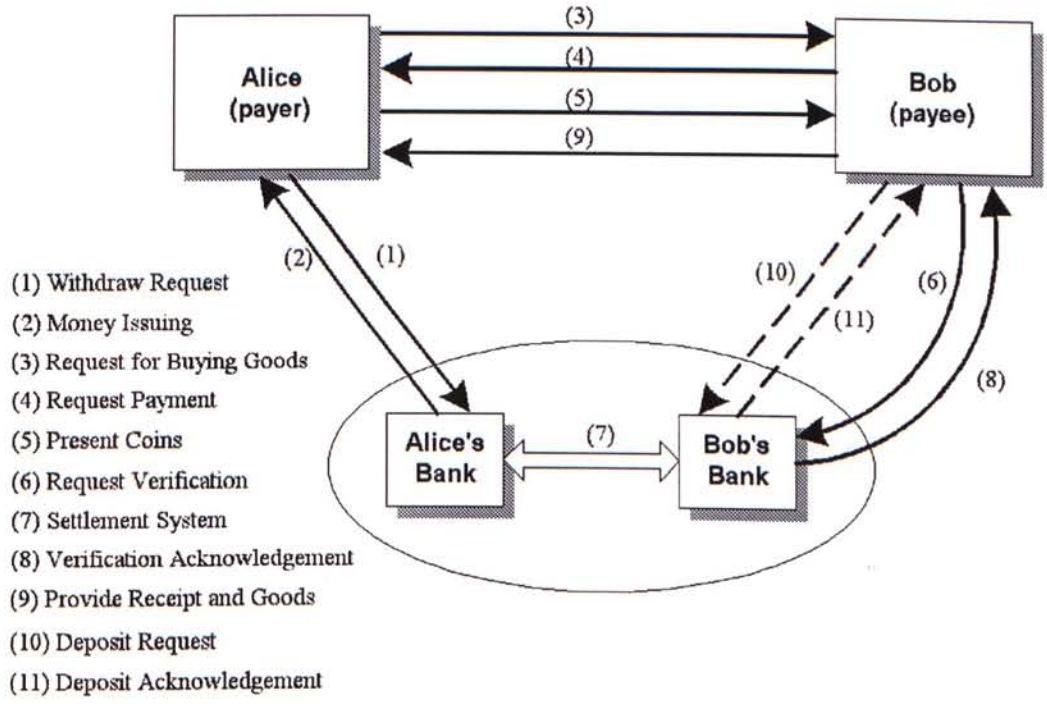

Figure 3.1: Online (Broker type) Electronic Cash System

Unconditional payer untraceability means that even an unlimited powerful bank or an authority cannot identify the identity of a user from provided information. **Computational untraceability** means that the bank or any party cannot identify the user by looking into given information unless it can make a computation, which is though to be infeasible.

3.1.3 Online and Off-line

An electronic cash scheme is said to be "Off-line" [CFN90] if the payment protocol does not involve the bank or a broker; otherwise the scheme is said to be "online" [Cha90, Dam90, PW92, PO95][1]. To lessen the possibility of confusion from computer terminologies, we can also called these scheme as *brokerless* and

[1] www.w3.org/pub/Conferences/WWW4/Papers/228/

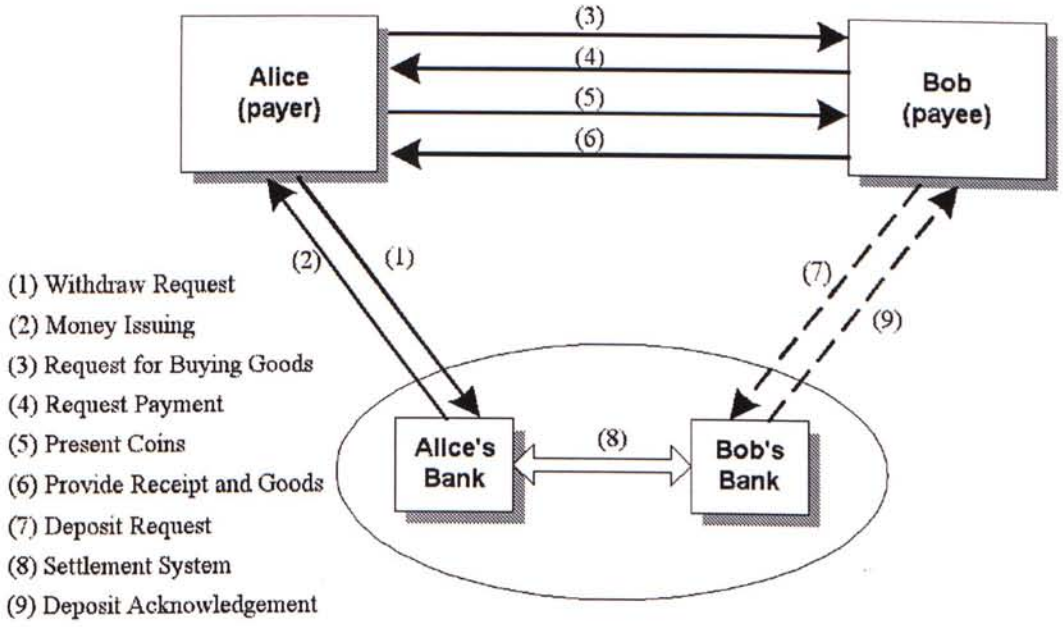

Figure 3.2: Off-line (Brokerless type) Electronic Cash System

broker type respectively. Figure 3.1 and 3.2 show the working principle of these two systems respectively.

One disadvantage of broker type electronic cash system is that the bank needs powerful and expensive computer systems for online verification processes for every transaction. Commonly, brokerless type electronic cash scheme is suitable for daily low value transactions where accountability afterwards is sufficient to deter abuse or where connecting to a broker in real-time is too costly or impossible under some conditions. Therefore, a practical system may usually operates in *dual mode*. That is to verify high-cost transactions online in order to prevent high-volume fraud; while the bulks of small payments are processed in batch mode by the bank. A noticeable online electronic cash system operating on Internet is DigiCash's *ecash*[2] which is founded by David Chaum in Holland.

[2]www.digicash.com

3.1.4 Security

In the issue of security (criteria (b)), what is to prevent anyone from making several copies of an electronic coin and using them at different shops? How can we detect whenever there is a counterfeit coin? Paper banknote doesn't present this problem, since making exact copies of them is though to be infeasible. Nor do credit cards, because their unique identity lets the bank take legal action to regain overdrawn balances, and the bank can add cards to a blacklist. Then, for electronic cash scheme, online clearing is one possible solution. This solution is adopted by credit card system though it's a rather expensive one and it's only applicable to online electronic cash systems. For an off-line scheme, the bank is not consulted during payment, and hence reuse cannot be prevented. However, the customer's identity can be embedded in each coin in a way that is accessible if and only if the same coin is used for more than one purchase — *double-spending* or *over-spending*. Thus, double-spending can be *detected* after the fact — when the copies of the coin are eventually deposited, the bank will learn the identity of the reuser.

Also, on the issue of proving the existence of secure electronic cash, Damgard [Dam90] shows that the security of an *online* electronic scheme using two-party computation protocol and zero-knowledge is provable. Some flaws were identified and corrected later in [PW92]. Pfitzmann and Waidner also showed how to construct a provably secure *off-line* scheme under the general assumption of the existence of trapdoor one-way permutations in a later time.

3.1.5 Transferability

Transferability (criteria (e)) of electronic cash means that the payee in one payment transaction can spend the received money in a later payment to a third person without contacting the bank or another central authority between the two transactions. As online electronic payment systems require communication with a central authority during the payment transaction, transferability is only an issue for off-line systems. Although the ability to transfer "normal" money (physical cash such as coins and notes) is very important and can said to be very convenient in our daily life, this property has only received very little attention in relation to electronic money.

Transferability is a feature that exists in physical cash but has not been applied in electronic cash. The main reason is the danger involved in transferring electronic coins and the subsequent increase in liability for the banks issuing those coins: an over-spending user is only identified when the coins he spent return to the bank; in addition, a single coin can be over-spent by all users from whose hands it has passed. The lack of interest in pursuing research or the inertia of further development in transferability can also be attributed to the facts that

1. depositing an electronic coin is much more efficient than depositing a physical token, hence transferability becomes less of an issue, and

2. a coin grows in size with every transfer, as shown by Chaum and Pedersen in [CP92]. This is because transcripts of the previous payments are appended and must be verified in each payment to prevent double spending.

3. a payer can recognize his money if he sees it later in the chain of payments

— forward traceability.

In [CP92], it not only shows that it is *impossible* to construct an electronic money system providing transferability without the property that the money grows when transferred but also can an unlimited powerful user always recognize his money later. In short, since each transfer requires information to prevent double spending and to be able to reveal the identity of double spender, the entropy of the transferred coin must increase for adding the information of recipient.

3.2 Panorama of Electronic Cash

In the evolution of Electronic Cash, there are many devoted classical papers that contribute to this area a lot.

3.2.1 First Model of Off-line Electronic Cash

First off-line anonymous electronic cash scheme was introduced by Chaum, Fiat and Naor in [CFN90]. Security of their scheme relied on some arbitrary assumptions however no formal proof was attempted. Although hardly practical, their system demonstrated how off-line e-cash can be constructed and laid the foundation for more secure and efficient schemes to follow. Their methodology was conceptually simple. At withdrawal, the bank verifies in a zero-knowledge manner that the user's identity is "embedded" (encrypted) in a randomly created (by the user) coin. Then it provides the user with a blind signature on this coin. At payment, the user provides a distinct "hint" on his identity, such that one hint provides one computationally secure and unconditionally blinded commitment

on the user's identity, whereas any two hints can identify the user. Hints are verifiable by the shop, i.e., a zero-knowledge proof that the hint corresponds to the identity in the coin is given. At deposit, the shop just transfers a payment transcript to the bank. Upon double spending of a coin, the bank identifies the user using the two distinct "hints" on his identity. The zero-knowledge proofs during withdrawal and payment were using the *cut-and-choose* methodology (see section 2.13).

It's the first off-line untraceable electronic cash system which satisfies criteria (a) to (d) stated in section 3.1.1 based on cut-and-choose methodology and a collision free one-way hash function technique [Sch94].

3.2.2 Successors

Evolution of e-cash system [OO90] proposed another similar one, which satisfies criteria (a) to (e). It used disposable zero-knowledge authentication scheme in place of collision free function. They made the first attempt to improve the First Model described in the previous section.

Although the cut-and-choose technique is still employed, the main evolution was to introduce one more step before Withdrawal Protocol, called Opening Protocol (account establishment). This step carries out the most complex part of the functionality of the previous withdrawal protocol, namely the zero-knowledge proof of the user's identity. Opening protocol provides user an untraceable "license". Thus, anonymity is established only once, in the form of a pseudonym, instead of being "refreshed" with every withdrawn coin. However, this scheme does not satisfy *perfect* untraceability, although it satisfies *linkable* untraceability. That is, the relationship between a user and his purchases cannot be traced

by anyone, but a purchase history of an anonymous user can be traced. Such linkability of user's coins may even cause users to be traced with conventional techniques, (i.e., using locality, time, type, size, frequency of payments, or by finding a single payment in which the user identified himself.) Hence, a compromise between the efficiency and the unlinkability of the system can be found by running the account-establishment protocol more than once.

3.2.3 Binary Tree Based Divisible Electronic Cash

In [OO92], Okamoto and Ohta first proposed the six criteria (Criteria (a) to (f)) of an ideal untraceable electronic cash and also their scheme is the first one which satisfies all those six criteria. As claimed by the authors, the total data transfer for a payment is about 20kB, and the protocol can be completed in several seconds, assuming the existence of a Rabin scheme [MVOV97] chip. The security of this scheme relies on difficulty of factoring (section 2.9). Key techniques are the square root modulo N (N is a Williams Integer). Divisibility is achieved by using binary tree structure table and is manipulated in quadratic residues properties.

Okamoto proposed another version in [Oka95] several years later. This scheme adopts **single term** cash scheme which uses **bit commitment** in place of **cut-and-choose methodology**. The divisibility of electronic coin is realized by a similar approach as [OO92].

Chapter 4

Spending Limit Enforced Electronic Cash

In this chapter, a new fundamental concept on untraceable divisible off-line electronic cash is proposed. It is called **Spending Limit Enforced Electronic Cash**[1]. A scheme, for realizing this concept, is also presented.

4.1 Introduction to Spending Limit Enforced Electronic Cash

Among the various kinds of off-line electronic cash schemes proposed since the introduction by Chaum, Fiat and Noar [CFN90], most efforts have been devoted to fulfill the requirements of security, privacy, complexity and coin divisibility. A big heap of improved schemes have been proposed and there are noticeable

[1]This concept and scheme plus another research result called **Interest-bearing Electronic Cash** presented in the next chapter are evolved with the crucial inspirations and involvement of Prof. Wei.

achievements established among these stringent requirements. However, there is no limitation of how many tokens a customer can spend per day or per month. The customer simply spends all the withdrawn tokens until all of them have been used up. There isn't any limitation on daily or monthly total expenditure.

In this report, we are going to change the perception on electronic cash by offering a new fundamental concept.

Concept This is a fundamental concept which imposes a daily or monthly spending limit to each divisible electronic coin. For each of such coins, the amount spent within one day or one month could not exceed the corresponding spending limit.

One application evolved can be delineated as there is a child who gets a sum of electronic cash from his parents at the beginning of each month. However, in each day, he can only spend a small portion of the money. This daily spending limit is enforced by the electronic cash scheme.

Our implementation uses a direct construction of coin numbers instead of cut-and-choose methodology (see section 2.13) during coin withdrawal. Also, a single term challenge-response secret **sharing line technique** [Fer94, Sha79] is used instead of multiple terms challenge-response scheme for detecting double spending during payment for higher efficiency.

The secret sharing line technique is an extension of [Fer94], which uses a 2-dimensional secret sharing line to enforce double spending detection, is first introduced by Shamir in [Sha79]. The central idea of Ferguson's scheme is to ask the user to give a *share* of his identity to the shop at each payment in response to a random challenge. This share is a point on the line $x \mapsto \alpha x + U$ where

α is a constant chosen randomly by the user and is secretly kept as a **secret sharing parameter**. U is the user's identity which is distinct for each coin. If the user spends the same coin twice, he has to reveal a second point on the line — the second share of his identity. With these two points, the values of α and U can be recovered by the banks easily and the double spender's identity, U, is also revealed.

This technique gets rid of large number of challenge-response terms and Ferguson uses this technique to detect double spending. In [Fer94], an algorithm to prevent double spending of an non-divisible coin was shown. Let each coin is represented by 3 **base numbers**: a, b, c. Alice (the user) constructs

$$A = f_a(a) \tag{4.1}$$

$$B = f_b(b) \tag{4.2}$$

$$C = f_c(c) \tag{4.3}$$

where $f_{\{a,b,c\}}$ are appropriate one-way functions. A, B and C are called the fingerprint of the 3 corresponding base numbers of the coin. Let U be Alice's identity and k is generated randomly by Alice and is called **secret sharing line parameter**. Alice gets 2 RSA-signatures from the bank by blind signature scheme (see section 2.12) so that bank don't know the values of her base numbers. The 2 signatures are

$$S_1 = (C^k A)^{1/v} \pmod{\sigma} \tag{4.4}$$

$$\text{and} \quad S_2 = (C^U B)^{1/v} \pmod{\sigma} \tag{4.5}$$

where v and σ are bank's public exponent and modulus respectively. v should be a reasonably large prime (say 128 bits).

During payment, Alice sends the base numbers (a, b, c) to shop. The shop randomly chooses a challenge number, x and sends back to Alice. Alice computes the **share** of her identity, r as

$$r = kx + U \pmod{v} \tag{4.6}$$

and a signature by

$$S = S_1^x \cdot S_2 \tag{4.7}$$

and sends (r, S) to the shop.

The shop constructs back A, B and C from the base numbers and then check

$$S^v \stackrel{?}{=} (C^r A^x B). \tag{4.8}$$

Since the challenge-response pair (x, r) only reveals one point (one *share* of Alice's identity) on the line

$$x \mapsto kx + U, \tag{4.9}$$

the shop cannot know the value of U, Alice's identity. However, if this coin is spent twice, Alice has to reveal another point on the line $x \mapsto kx + U$ provided that the value of x is different from that of the first time. These 2 points let the bank to recover the value of U once these 2 transactions have reached bank from the shops.

In our implementation scheme, we will construct an extension of this technique to realize an **unit of time spending limit enforced electronic cash**.

In the next section, we first describe the brief protocol of a scheme to realize the concept by leaving the detailed techniques untouched till later sections.

4.2 The Scheme

We use the term, **unit of time** spending limit, to replace daily or monthly spending limit for maintaining generality. To ease explanation, we assume the spending limit for each time unit of a coin is constant first. It is easy to extend the scheme to support variable spending limits among different time units and this will be elaborated later in this chapter. Let there is a coin worth N dollars. It can be used over T units of time and its unit of time spending limit is M dollars/unit of time. To make the coin be able to have all its value be spent within the T time units, we assume that $N \leq MT$. We first represent this coin by two sets of **base numbers**:

$$\begin{aligned} \Sigma &= \{e_1, e_2, \cdots, e_{N+1}, e_{N+2}\} \\ \Phi &= \begin{bmatrix} P_1 & P_2 & \cdots & P_T \end{bmatrix}^T \end{aligned}$$

where $P_i = \{l_{ij} \mid j = 1, \cdots, M+2\}$ for $i = 1, \cdots, T$. Σ is called **coin base number set** which is used to preventing coin over-spending and Φ is used to imposing unit time limitation on expenditure and also to enforce coin expiration feature. There are T rows of number sets in Φ, each row P_i is a **time base number set**, used to confine every specific time unit's spending limit to M dollars/unit of time.

Assume there are appropriate one-way functions $f(\cdot)$s for these base numbers such that

$$E_i = f_{e_i}(e_i) \qquad \text{for } i = 1, \cdots, N+2 \tag{4.10}$$

$$L_{ij} = f_{l_{ij}}(l_{ij}) \qquad \text{for } i = 1, \cdots, T;\ j = 1, \cdots, M+2 \tag{4.11}$$

During withdrawal, the user, Alice first obtains two sets of random numbers

as her **secret sharing line parameters**:

$$\{c_i \mid i = 1, \cdots, N\} \quad \text{and}$$
$$\{d_{ij} \mid i = 1, \cdots, T; j = 1, \cdots, M\}$$

and also the corresponding RSA-signatures which are blindly signed by the bank in a direct construction which will be explained in details in section 4.4.

$$S_{c_i} = (E_{N+2}^{c_i} E_i)^{1/v} \quad \text{for } i = 1, \cdots, N \tag{4.12}$$
$$S_{c_{N+1}} = (E_{N+2}^{U_c} E_{N+1})^{1/v} \tag{4.13}$$

$$S_{d_{ij}} = (L_{i(M+2)}^{d_{ij}} L_{ij})^{1/v} \quad \text{for } i = 1, \cdots, T;\ j = 1, \cdots, M \tag{4.14}$$
$$S_{d_{i(M+1)}} = (L_{i(M+2)}^{U_i} L_{i(M+1)})^{1/v} \quad \text{for } i = 1, \cdots, T \tag{4.15}$$

where U_c and U_i are user's identity of the coin and of the time unit i respectively. All computations are done in an RSA system [RSA78] where the bank knows the factorization of modulus σ. The corresponding public exponent of this RSA system is v.

During payment, we first assume Alice only spends 1 dollar on ith time unit to ease description. Herewith, we show a **one-move** protocol for achieving better efficiency (see Figure 4.1).

Alice first generates two random numbers $\alpha_e, \alpha_l \in_{\mathcal{R}} \mathbb{Z}_\sigma^*$ and computes two challenge numbers by

$$x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e) \tag{4.16}$$
$$x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l) \tag{4.17}$$

where $\mathcal{H}_e$ and $\mathcal{H}_l$ are two hash functions; and I_s is the identity of the shop, Bob. Then, she computes two corresponding responses r_e and r_l which are her

Alice **Shop**

$$\xleftarrow{\quad I_s \quad}$$

$\alpha_e, \alpha_l \in_{\mathcal{R}} \mathbb{Z}_\sigma^*$

$x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e)$

$x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l)$

$r_e = c_1 x_e^N + c_2 x_e^{N-1} + \cdots + c_N x_e + U_c$

$r_l = d_{i1} x_l^M + d_{i2} x_l^{M-1} + \cdots + d_{iM} x_l + U_i$

$S_e = S_{c_1}^{x_e^N} S_{c_2}^{x_e^{N-1}} \cdots S_{c_N}^{x_e} S_{c_{N+1}}$

$S_l = S_{d_{i1}}^{x_l^M} S_{d_{i2}}^{x_l^{M-1}} \cdots S_{d_{iM}}^{x_l} S_{d_{i(M+1)}}$

$$\xrightarrow{\quad (\Sigma, P_i), (\alpha_e, \alpha_l), (r_e, r_l), (S_e, S_l) \quad}$$

$E_i = f_{e_i}(e_i) \quad \text{for } i = 1, \cdots, N+2$

$L_{ij} = f_{l_{ij}}(l_{ij}) \quad \text{for } i = 1, \cdots, T;\ j = 1, \cdots, M+2$

$x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e)$

$x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l)$

$S_e^v \stackrel{?}{=} E_{N+2}^{r_e} E_1^{x_e^N} \cdots E_N^{x_e} E_{N+1}$

$S_l^v \stackrel{?}{=} L_{i(M+2)}^{r_l} L_{i1}^{x_l^M} \cdots L_{iM}^{x_l} L_{M+1}$

Figure 4.1: A brief illustration of one-move Payment Protocol.

identity shares and two signatures S_e and S_l corresponding to x_e and x_l as below:

$$r_e = c_1 x_e^N + c_2 x_e^{N-1} + \cdots + c_N x_e + U_c \tag{4.18}$$

$$r_l = d_{i1} x_l^M + d_{i2} x_l^{M-1} + \cdots + d_{iM} x_l + U_i \tag{4.19}$$

$$S_e = S_{c_1}^{x_e^N} S_{c_2}^{x_e^{N-1}} \cdots S_{c_N}^{x_e} S_{c_{N+1}} \tag{4.20}$$

$$S_l = S_{d_{i1}}^{x_l^M} S_{d_{i2}}^{x_l^{M-1}} \cdots S_{d_{iM}}^{x_l} S_{d_{i(M+1)}} \tag{4.21}$$

Alice sends the coin (Σ, P_i), challenge numbers (α_e, α_l), corresponding identity shares (r_e, r_l) and signatures (S_e, S_l) to Bob. When Bob receives, he verifies the consistency of all the parameters. If the transaction requires Alice to spend more than 1 dollar, the above steps are to be repeated for every spending of 1 dollar. Note that Bob has to check that all α_e and α_l are different and uniquely generated for every spend of 1 dollar.

4.3 An Example

For a coin worth \$10 and the minimum divisible unit is \$1. The coin will be expired after 1 year and there is a **monthly** spending limit imposed as \$2 per month. Then, the two sets of **base numbers** that representing the coin would be:

$$\Sigma = \{e_1, e_2, \cdots, e_{11}, e_{12}\}$$

$$\Phi = \begin{bmatrix} P_1 & P_2 & \cdots & P_{12} \end{bmatrix}^T$$

where $P_i = \{l_{ij} \mid j = 1, \cdots, 4\}$ for $i = 1, \cdots, 12$. Also, the following equations are constructed for this coin.

$$E_i = f_{e_i}(e_i) \qquad \text{for } i = 1, \cdots, 12 \tag{4.22}$$

$$L_{ij} = f_{l_{ij}}(l_{ij}) \qquad \text{for } i = 1, \cdots, 12;\ j = 1, \cdots, 4 \tag{4.23}$$

During withdrawal, Alice first obtains her **secret sharing line parameters** as:

$$\{c_i \mid i = 1, \cdots, 10\} \quad \text{and}$$

$$\text{and} \quad \{d_{ij} \mid i = 1, \cdots, 12; j = 1, \cdots, 2\}$$

and also the corresponding RSA-signatures:

$$S_{c_i} = (E_{12}^{c_i} E_i)^{1/v} \qquad \text{for } i = 1, \cdots, 10 \tag{4.24}$$

$$S_{c_{11}} = (E_{12}^{U_c} E_{11})^{1/v} \tag{4.25}$$

$$S_{d_{ij}} = (L_{i4}^{d_{ij}} L_{ij})^{1/v} \qquad \text{for } i = 1, \cdots, 12;\ j = 1, \cdots, 2 \tag{4.26}$$

$$S_{d_{i3}} = (L_{i4}^{U_i} L_{i3})^{1/v} \qquad \text{for } i = 1, \cdots, 12. \tag{4.27}$$

During payment (see Figure 4.2), Alice first generates two random numbers α_e and α_l, and computes the two challenge numbers by $x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e)$ and $x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l)$. Then, she computes two corresponding responses r_e and r_l and two signatures S_e and S_l as below:

$$r_e = c_1 x_e^{10} + c_2 x_e^9 + \cdots + c_{10} x_e + U_c \tag{4.28}$$

$$r_l = d_{i1} x_l^2 + d_{i2} x_l + U_i \tag{4.29}$$

$$S_e = S_{c_1}^{x_e^{10}} S_{c_2}^{x_e^9} \cdots S_{c_{10}}^{x_e} S_{c_{11}} \tag{4.30}$$

$$S_l = S_{d_{i1}}^{x_l^2} S_{d_{i2}}^{x_l} S_{d_{i3}} \tag{4.31}$$

Alice **Bob**

$$\xleftarrow{\quad I_s \quad}$$

$\alpha_e, \alpha_l \in_{\mathcal{R}} \mathbb{Z}_\sigma^*$

$x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e)$

$x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l)$

$r_e = c_1 x_e^{10} + c_2 x_e^9 + \cdots + c_{10} x_e + U_c$

$r_l = d_{i1} x_l^2 + d_{i2} x_l + U_i$

$S_e = S_{c_1}^{x_e^{10}} S_{c_2}^{x_e^9} \cdots S_{c_{10}}^{x_e} S_{c_{11}}$

$S_l = S_{d_{i1}}^{x_l^2} S_{d_{i2}}^{x_l} S_{d_{i3}}$

$$\xrightarrow{\quad (\Sigma, P_i), (\alpha_e, \alpha_l), (r_e, r_l), (S_e, S_l) \quad}$$

$E_i = f_{e_i}(e_i) \quad \text{for } i = 1, \cdots, 12$

$L_{ij} = f_{l_{ij}}(l_{ij}) \quad \text{for } i = 1, \cdots, 12;\ j = 1, \cdots, 4$

$x_e = \mathcal{H}_e(\Sigma \| I_s \| \alpha_e)$

$x_l = \mathcal{H}_l(P_i \| I_s \| \alpha_l)$

$S_e^v \stackrel{?}{=} E_{12}^{r_e} E_1^{x_e^{10}} \cdots E_{10}^{x_e} E_{11}$

$S_l^v \stackrel{?}{=} L_{i4}^{r_l} L_{i1}^{x_l^2} L_{i2}^{x_l} L_3$

Figure 4.2: A monthly spending limit enforced electronic coin.

Alice sends the coin (Σ, P_i), challenge numbers (α_e, α_l), corresponding identity shares (r_e, r_l) and signatures (S_e, S_l) to Bob for verification.

4.4 Techniques

Now, we can delve in the details of withdrawal protocol which uses a technique called **direct construction**. The direct construction which builds the withdrawal protocol for high efficiency. For those base number sets, Σ and P_i $(i = 1, \cdots, T)$, each of them follows a similar procedure of direct construction to obtain the secret sharing parameters $(\{c_i\}, \{d_{ij}\})$ and corresponding signatures $(\{S_{c_i}\}, \{S_{d_{ij}}\})$. The withdrawal protocol (see Figure 4.3) can be concluded as $n + 2$ parallel runs of randomized blind signature scheme for each set of base numbers.

Here below is the direct construction protocol to construct a base number set $\{c_i \mid i = 1, \cdots, n+2\}$, the secret sharing line parameters $\{\alpha_i \mid i = 1, \cdots, n\}$ and corresponding signatures $\{S_i \mid i = 1, \cdots, n+1\}$ (Note that the notations in this section are selected differently from the previous sections for maintaining generality).

Step 1. Alice starts by choosing three sets of random numbers :

1. Alice's contributions to the base numbers

   $c'_1, c'_2, \cdots, c'_{n+2} \in_{\mathcal{R}} \mathbb{Z}^*_\sigma$

2. exponential blinding factors

   $\mu_1, \mu_2, \cdots, \mu_{n+2} \in_{\mathcal{R}} \mathbb{Z}_v$

3. multiplicative blinding factors

Alice **Bank**

$c'_1, c'_2, \cdots, c'_{n+2} \in_{\mathcal{R}} \mathbb{Z}^*_\sigma$

$\mu_1, \mu_2, \cdots, \mu_{n+2} \in_{\mathcal{R}} \mathbb{Z}_v$

$\beta_1, \beta_2, \cdots, \beta_{n+2} \in_{\mathcal{R}} \mathbb{Z}^*_\sigma$

$\{\beta_i^v c'_i g_{c_i}^{\mu_i} \mid i = 1, \cdots, n+2\}$ →

$c''_1, c''_2, \cdots, c''_{n+2} \in_{\mathcal{R}} \mathbb{Z}^*_\sigma$

← $\{c''_i \mid i = 1, \cdots, n\},\ h_{c_{n+1}}^{c''_{n+1}},\ h_{c_{n+2}}^{c''_{n+2}}$

$\alpha'_1, \alpha'_2, \cdots, \alpha'_n \in_{\mathcal{R}} \mathbb{Z}^*_v$

$e_{c_{n+1}} \leftarrow f(h_{c_{n+1}}^{c'_{n+1}c''_{n+1}}) - \mu_{n+1}$

$e_{c_{n+2}} \leftarrow f(h_{c_{n+2}}^{c'_{n+2}c''_{n+2}}) - \mu_{n+2}$

$c_i \leftarrow (c'_i c''_i \cdot f_2(e_{c_{n+1}}, e_{c_{n+2}}))^{\alpha'_i}$ for $i = 1, \cdots, n$

$e_{c_i} \leftarrow \frac{1}{\alpha'_i} f(c_i) - \mu_i$ for $i = 1, \cdots, n$

$\{e_{c_i} \mid i = 1, \cdots, n\},\ e_{c_{n+1}},\ e_{c_{n+2}}$ →

$\overline{C_{n+2}} \leftarrow \beta_{n+2}^v c'_{n+2} g_{c_{n+2}}^{\mu_{n+2}} c''_{n+2} g_{c_{n+2}}^{e_{c_{n+2}}}$

$\overline{C_{n+1}} \leftarrow \beta_{n+1}^v c'_{n+1} g_{c_{n+1}}^{\mu_{n+1}} c''_{n+1} g_{c_{n+1}}^{e_{c_{n+1}}}$

$\overline{C_i} \leftarrow \beta_i^v c'_i g_{c_i}^{\mu_i} c''_i \cdot f_2(e_{c_{n+1}}, e_{c_{n+2}}) g_{c_i}^{e_{c_i}}$ for $i = 1, \cdots, n$

$\alpha''_1, \alpha''_2, \cdots, \alpha''_n \in_{\mathcal{R}} \mathbb{Z}^*_v$

← $c''_{n+1},\ c''_{n+2},\ \{\alpha''_i \mid i = 1, \cdots, n\}$

$\{(\overline{C_{n+2}}^{\alpha''_i} \cdot \overline{C_i})^{1/v} \mid i = 1, \cdots, n\}$

$(\overline{C_{n+2}}^{U} \cdot \overline{C_{n+1}})^{1/v}$

$c_{n+1} = c'_{n+1} c''_{n+1}$

$c_{n+2} = c'_{n+2} c''_{n+2}$

$\alpha_i = \alpha'_i \alpha''_i \pmod{v}$ for $i = 1, \cdots, n$

$C_{n+1} \leftarrow c_{n+1} g_{c_{n+1}}^{f(h_{c_{n+1}}^{c_{n+1}})}$

$C_{n+2} \leftarrow c_{n+2} g_{c_{n+2}}^{f(h_{c_{n+2}}^{c_{n+2}})}$

$C_i \leftarrow c_i g_{c_i}^{f(c_i)}$ for $i = 1, \cdots, n$

$S_i \leftarrow ((\overline{C_{n+2}}^{\alpha''_i} \cdot \overline{C_i})^{1/v} / \beta_{n+2}^{\alpha''_i} \beta_i)^{\alpha'_i}$ for $i = 1, \cdots, n$

$S_{n+1} \leftarrow (\overline{C_{n+2}}^{U} \cdot \overline{C_{n+1}})^{1/v} / \beta_{n+2}^U \beta_{n+1}$

$S_i^v \stackrel{?}{=} C_{n+2}^{\alpha_i} C_i$ for $i = 1, \cdots, n$

$S_{n+1}^v \stackrel{?}{=} C_{n+2}^U C_{n+1}$

Figure 4.3: Withdrawal Protocol Using Direct Construction

$$\beta_1, \beta_2, \cdots, \beta_{n+2} \in_{\mathcal{R}} \mathbb{Z}_\sigma^*$$

Alice then computes $\{\beta_i^v c_i' g_{c_i}^{\mu_i} \mid i = 1, \cdots, n+2\}$ and sends the computation results to the bank where g_{c_i}s are publicly known elements of large order in $\mathbb{Z}_\sigma^*$.

Step 2. The bank chooses its contributions to the base numbers

$$c_1'', c_2'', \cdots, c_{n+2}'' \in_{\mathcal{R}} \mathbb{Z}_\sigma^*$$

It then sends $\{c_i'' \mid i = 1, \cdots, n\}$, $h_{c_{n+1}}^{c_{n+1}''} \bmod p$ and $h_{c_{n+2}}^{c_{n+2}''} \bmod p$ to Alice where p is a prime congruent to 1 modulo σ, and $h_{c_{n+1}}$ and $h_{c_{n+2}}$ are publicly known elements of order σ in $\mathbf{F}_p$.

Step 3. Alice chooses a set of random numbers

$$\alpha_1', \alpha_2', \cdots, \alpha_n' \in_{\mathcal{R}} \mathbb{Z}_v^*$$

and computes the corresponding exponents by

$$e_{c_{n+1}} = f(h_{c_{n+1}}^{c_{n+1}' c_{n+1}''}) - \mu_{n+1} \tag{4.32}$$

$$\text{and} \quad e_{c_{n+2}} = f(h_{c_{n+2}}^{c_{n+2}' c_{n+2}''}) - \mu_{n+2} \tag{4.33}$$

where $f(\cdot)$ is a suitable one-way function mapping $\mathbb{Z}_\sigma^*$ into $\mathbb{Z}_v$. She also computes the set of base numbers as

$$c_i = (c_i' c_i'' \cdot f_2(e_{c_{n+1}}, e_{c_{n+2}}))^{\alpha_i'} \qquad \text{for } i = 1, \cdots, n \tag{4.34}$$

where $f_2(\cdot)$ is another one-way function that makes c_i depend on $e_{c_{n+1}}$ and $e_{c_{n+2}}$. She also computes the corresponding exponents of c_i as

$$e_{c_i} = \frac{1}{\alpha_i'} f(c_i) - \mu_i \qquad \text{for } i = 1, \cdots, n \tag{4.35}$$

Note that all calculations of the exponents are done in modulo v. Alice then sends all the exponents to the bank.

Step 4. The bank computes the *blinded* version of $\{\overline{C_i} \mid i = 1, \cdots, n+2\}$ as

$$\overline{C_{n+1}} = \beta_{n+1}^{v} c'_{n+1} g_{c_{n+1}}^{\mu_{n+1}} c''_{n+1} g_{c_{n+1}}^{e_{c_{n+1}}} \tag{4.36}$$

$$\overline{C_{n+2}} = \beta_{n+2}^{v} c'_{n+2} g_{c_{n+2}}^{\mu_{n+2}} c''_{n+2} g_{c_{n+2}}^{e_{c_{n+2}}} \tag{4.37}$$

$$\overline{C_i} = \beta_i^{v} c'_i g_{c_i}^{\mu_i} c''_i \cdot f_2(e_{c_{n+1}}, e_{c_{n+2}}) g_{c_i}^{e_{c_i}} \quad \text{for } i = 1, \cdots, n \tag{4.38}$$

Then, the bank chooses a set of random numbers $\alpha''_1, \alpha''_2, \cdots, \alpha''_n \in_{\mathcal{R}} \mathbb{Z}_v^*$ and sends c''_{n+1}, c''_{n+2}, $\{\alpha''_i \mid i = 1, \cdots, n\}$, $\{(\overline{C_{n+2}}^{\alpha''_i} \cdot \overline{C_i})^{1/v} \mid i = 1, \cdots, n\}$, and $(\overline{C_{n+2}}^{U} \cdot \overline{C_{n+1}})^{1/v}$ to Alice.

Step 5. Alice computes the two remaining base numbers c_{n+1} and c_{n+2} as $c'_{n+1}c''_{n+1}$ and $c'_{n+2}c''_{n+2}$ respectively. Also, she constructs the secret sharing line parameters $\{\alpha_i \mid i = 1, \cdots, n\}$ as $\alpha_i = \alpha'_i \alpha''_i \pmod{v}$. Then, she can construct the numbers C_{n+1} and C_{n+2} as $c_{n+1} g_{c_{n+1}}^{f(h_{c_{n+1}}^{c_{n+1}})}$ and $c_{n+2} g_{c_{n+2}}^{f(h_{c_{n+2}}^{c_{n+2}})}$ respectively, also C_i as $c_i g_{c_i}^{f(c_i)}$ for $i = 1, \cdots, n$ and the signatures as

$$S_i = ((\overline{C_{n+2}}^{\alpha''_i} \cdot \overline{C_i})^{1/v} / \beta_{n+2}^{\alpha''_i} \beta_i)^{\alpha'_i} \quad \text{for } i = 1, \cdots, n \tag{4.39}$$

$$S_{n+1} = (\overline{C_{n+2}}^{U} \cdot \overline{C_{n+1}})^{1/v} / \beta_{n+2}^{U} \beta_{n+1} \tag{4.40}$$

Alice finally checks that the signatures she received are correct. After withdrawal, Alice obtained a base number set $\{c_i \mid i = 1, \cdots, n+2\}$, secret sharing line parameters $\{\alpha_i \mid i = 1, \cdots, n\}$ and the signatures $\{S_i \mid i = 1, \cdots, n+1\}$.

4.5 Security and Efficiency

In the payment protocol, if Alice has spent η dollars ($1 \leq \eta \leq N$) of a coin which is worth N dollars, Alice should sends to shops η sets of challenge numbers and her identity shares $\{(x_{e_k}, r_{e_k}), (x_{l_k}, r_{l_k}) \mid k = 1, \cdots, \eta\}$ which will eventually be sent to the bank. Now, if the coin is challenged to a total of more than N times, but $(N+1)$ times, in a case of overspending as an example, Alice must provides $(N+1)$ pairs of challenge numbers and identity shares $\{(x_{e_k}, r_{e_k}) \mid k = 1, \cdots, N+1\}$ which reveals $(N+1)$ points on the line $x_e \mapsto \alpha_1 x_e^N + \alpha_2 x_e^{N-1} + \cdots + \alpha_n x_e + U_c$ that immediately allow the bank to determine her identity U_c. Thus, in this scheme, we use $N+2$ coin base numbers, Σ, to represent a coin which is worth N dollars and a corresponding $(N+1)$-dimensional secret sharing line to detect overspending.

In addition, we use another set of base numbers, Φ, which has T rows of time base numbers, to enforce unit of time spending limit on the coin. For each row of Φ, P_i, this particular set of base numbers imposes an expenditure limitation up to M dollars for each time unit by using the same technique. Also, for each unit of time, only a specific row of Φ for that time unit, P_i, can be used. This can be guaranteed by using different bank signing key for each different time unit. Thus, the coin cannot be used after T time units since no P_i for $i > T$ exists. This enforces coin expiration automatically. As noted, Σ and Φ are basically independent and thus they can be implemented by using different schemes. As an example, overspending can be detected by various kinds of divisible electronic cash schemes proposed such as binary-tree divisible electronic cash schemes proposed in [OO92], [EO94] and [Oka95]. In addition,

since each P_i is independent to other P_js for $i \neq j$, the number of time base numbers in P_i is not necessarily equal to the number of all other P_js'. This can be utilized to impose a variable unit of time spending limit to an electronic coin. It also implies that for a coin worth N dollar, the unit of time spending limit for time unit i is M_i and the coin can lasts for T time units. Then

$$N \leq \sum_{i=1}^{T} M_i \tag{4.41}$$

and the time base number set would be modified as $P_i = \{l_{ij} \mid i = 1, \cdots, M_i + 2\}$ for $i = 1, \cdots, T$. The subsequent protocols described in section 4.2 have to be modified accordingly.

If we look into the efficiency, the identification of double spenders requires multiple terms for each coin in most of the earlier schemes. Each term is used to answer one bit of challenge from the shop during payment. If both possible answers for any term is ever given, the user's identity is revealed. To achieve an acceptable probability of detection, a large number of terms is required. In this scheme, only one pair of challenge-response term is performed for spending each dollar. This single term payment scheme is much more efficiency than those previously proposed schemes. In addition, the direct construction methodology employed in withdrawal protocol gets rid of k time cut-and-choose methodology. The degree of efficiency improvement is as high as the factor of k.

Chapter 5

Interest-bearing Electronic Cash

In this chapter, another concept on untraceable divisible off-line electronic cash is proposed. It is called **Interest-bearing Electronic Cash**. This concept is to impose an interest rate on the balance of each divisible electronic cash. It is a sequel of previous chapter in implementation.

5.1 Introduction to Interest-bearing Electronic Cash

Currently proposed electronic cash systems are more or less the debit type without exception — no interest borne on the electronic cash. Debit type systems are easily associated with debit cards including phone cards, transportation cards and bankcards and none of them does bear interest. In essence, those debit cards require the customer to pay the card issuer (e.g. the bank) a sum of money in advance. The issuer takes the money, issues debit card and clears purchases

when they are requested. But at any time, the issuer does not pay any interest onto the balance of the cards. Analogously, if a user withdraws electronic cash from a bank, the bank deducts the amount from his account and issues electronic cash which most often will be spent in token form. The money is deducted from the user's bank account prior any expenditure of the electronic tokens and the bank does not pay any interest in the means of extra tokens to the balance of electronic cash. But if we look at other side by reviewing various kinds of accounts of bank systems, conventional current accounts do not bear interest while savings accounts bear interest. This paper is going to change the perception on electronic cash by offering a new fundamental concept and by presenting a construction of an interest-bearing electronic cash system.

Concept To impose an interest rate on the balance of each divisible electronic coin.

In the new conceptual scheme, interest is imposed on the balance of electronic coins at the end of certain period. For positive interest rate, the user's electronic coins are just like to be stored inside his savings account and there is interest added into his account at the end of each month with regard to his account's balance.

About implementation, this scheme uses the same technique as described in last chapter. It uses a single term, secret sharing line technique, and a direct construction instead of cut-and-choose methodology. Also, a single term secret sharing line scheme is used instead of multiple terms challenge-response scheme for detecting double-spending during payment.

As these two schemes we are using compatible protocols. We will finally

show how to simultaneously impose unit of time spending limit features that we discussed in last chapter with interest-bearing feature on the electronic coins by using a similar direct construction.

5.2 An Example

Herewith, I bring in an example first for describing how an interest-bearing electronic cash system behaves.

For example, Alice withdraws \$100 from the bank where the minimum divisible unit for this system is \$1. In the first month, Alice spends \$40. At the end of the month, there is an interest rate of 10% imposed to her electronic cash balance. Thus, \$6 is added to her balance which is summed up to \$66. During the second month, she awards an increase of the interest rate to 20% due to some sorts of incentive and it is applied to her electronic cash balance by the end of this month. If she only spends \$1 in the second month, her new balance will be \$78 at the beginning of third month. This scheme associates many other applications. Besides the conventional positive interest rate imposed on the balance, a negative interest rate can be applied to stimulate the usage of electronic cash as an example. Below is the description of a scheme which uses the generalized withdrawal and payment protocols presented previously.

5.3 The Scheme

We use same notations as that in the last chapter for easy expression and also declare some new symbols for extra parameters without losing generality. Let

Alice withdraws a coin which is worth W dollars. It is represented by a set of **base numbers** $\Sigma = \{c_i \mid i = 1, \cdots, n+2\}$ where n is pre-defined during the coin withdrawal such that $\frac{n}{W} = \lambda_1$ where λ_1 is the number of secret sharing points should be revealed for spending one dollar in the first time period. Let $C_i = f_{c_i}(c_i)$ $(i = 1, \cdots, n+2)$ where $f_{c_i}(\cdot)$s are suitable one-way functions and C_i is a corresponding fingerprint of c_i. During withdrawal, Alice obtains $(n+1)$ RSA-signatures from the bank, $S_\Sigma = \{S_1, S_2, \cdots, S_{n+1}\}$, where

$$S_j = (C_{n+2}^{\alpha_j} C_j)^{1/v} \quad \text{for } j = 1, \cdots, n \tag{5.1}$$

$$\text{and} \quad S_{n+1} = (C_{n+2}^{U} C_{n+1})^{1/v} \tag{5.2}$$

for $\{\alpha_j \mid j = 1, \cdots, n\}$ is a set of random numbers as her **secret sharing line parameters** and U is Alice's identity. The detailed direct construction technique for forming the base number set Σ, secret sharing parameters $\{\alpha_j\}$ and the signature set S_Σ can be referred to section 4.4.

During the first time period, we assume that Alice pays Bob 1 dollar. Alice needs to reveal λ_1 secret sharing points to Bob and this process can be described as below:

Step 1. The process is started when Alice sends the coin number set

$$\Sigma = \{c_i \mid i = 1, \cdots, n+2\}$$

to Bob.

Step 2. Bob sends back a set of randomly chosen non-zero challenge numbers,

$$\{x_k \mid k = 1, \cdots, \lambda_1\}.$$

This means that Bob requests Alice to reveal λ_1 points on the secret-sharing line for spending of 1 dollar.

Step 3. Alice then replies λ_1 pairs of identity shares and corresponding signatures $\{(r_k, \Gamma_k) \mid k = 1, \cdots, \lambda_1\}$ as

$$r_k = \alpha_1 x_k^n + \alpha_2 x_k^{n-1} + \cdots + \alpha_n x_k + U \tag{5.3}$$

$$\text{and} \quad \Gamma_k = S_1^{x_k^n} S_2^{x_k^{n-1}} \cdots S_n^{x_k} S_{n+1} \tag{5.4}$$

Step 4. Finally, Bob verifies the consistency of the responses.

We let the interest rate be ε_1 at the end of the first time period. Then, the number of secret sharing points per dollar for the second time period, λ_2, will be

$$\lambda_2 = \frac{\lambda_1}{1 + \varepsilon_1}. \tag{5.5}$$

Herewith, we assume that $(1 + \varepsilon_1)|\lambda_1$. If not, we need some rounding policies to round the value of λ_2 to the appropriate quantity of the number of secret sharing points. The above steps are then repeated with λ_2 in place of λ_1 for the second time period and so on.

5.4 Security

The idea of introducing λ_i is to change the number of points required to reveal on the secret sharing line for spending one dollar in each different time period. Since the overall number of points can be revealed is fixed during coin withdrawal and if the remaining amount of the withdrawn coin needs to be changed when the time elapses, we should change the number of secret sharing points for each dollar accordingly — λ_i.

The whole process is carried out to T cycles such that

$$\sum_{i=1}^{T} \lambda_i \omega_i \leq n \tag{5.6}$$

where ω_i is the amount in dollars being spent in ith time period. After T cycles being carried out, there are altogether at most n points on the secret sharing line

$$x \mapsto \alpha_1 x^n + \alpha_2 x^{n-1} + \cdots + \alpha_n x + U \tag{5.7}$$

having been revealed. Alice's identity is protected from being revealed even if there is a collusion of all other parties such as the bank and the shops. However, if Alice overspent, she has to reveal more than n points on the line shown above (equation 5.7), this immediately allow the bank to determine her identity U once all the transactions of this coin have been deposited to the bank from the incurred shops.

In the protocol shown in section 5.3, it shows the **two-move** protocol, which is contrary to **one-move** protocol presented in section 4.2 of last chapter. Actually, the interest-bearing electronic cash scheme can also employ one-move protocol. It is easy to switch this scheme to the previous version by referring to the details of section 4.2.

5.5 An Integrated Scheme

Now, we explain the idea to build a single electronic cash system which equips all three features — interest-bearing, daily or monthly spending limit and coin expiration by associating both interest-bearing and spending limit algorithms.

As we notice from section 5.3, the interest-bearing scheme adopts a dynamic adjustment to the number of secret sharing points being revealed for spending one dollar with regard to the variation of interest rate. While the spending limit algorithm sticks to a static construction of the coins which is set during withdrawal. Thus, in order to maintain the floating interest rate characteristic, we only need to replace Σ and its followings of the spending limit algorithm by that of interest-bearing scheme. The time base number sets P_is in Φ and its following derivations are remained unchanged.

5.6 Applications

For positive interest rate, people would prefer to hold interest-bearing electronic cash instead of paper notes and physical coins to lessen from depreciation due to sharp inflation. In addition, the electronic cash also provides divisibility besides user anonymity. This divisibility feature of electronic cash is even more flexible than that of physical cash. Also, it can provide interest to the balance as our bank savings account. One advantage of it over the bank savings account is that the payment of electronic cash can be performed off-line. Thus, there is a potential for electronic cash to replace physical notes and the speed may be faster when interest-bearing electronic cash systems are employed in practice to such a cashless society.

For negative interest rate, it can stimulate the usage of electronic coins. This feature can be incorporated as an instance into an individual based promotion tactic. For example, there is a bank who issues negative interest rate borne electronic *tokens* to its credit card users as bonus points with regard to the

amount of credit made by the user. The bank encourages its users to use all of the bonus points as quickly as possible with money to buy some promoting products. If a user doesn't use the bonus points at once, the bonus points will decrease automatically with time. Although the bonus will never expire, the quantity will decrease continually with time until vanished at all. This individual based credit card bonus point policy is much more flexible than most existing schemes adopted by the bank.

By employing the daily or monthly spending limit feature to the electronic cash as well, we can further extend the applications. In the consideration of security, we may want to limit the total amount of withdrawn cash in our electronic wallets that can be spent within one day or one month. By using unit-of-time spending limit enforced electronic cash, we can reduce our loss of electronic cash if our electronic wallets were stolen and misused.

Chapter 6

Abacus Type Electronic Cash

In this chapter, I dwells on a new divisible cash structure concept. I name it as **Abacus** because of its similar nature to a kind of Chinese calculating tools, *Abacus*. An Abacus type divisible untraceable off-line (brokerless) electronic cash scheme can be used to improve the efficiency of a conventional binary tree type electronic cash system [DC95, EO94, OO92, Oka95, Pai93] for daily usage. Furthermore, this proposing scheme shows a significant improvement on the part of **Opening Protocol** (refer to section 3.1.1). In this chapter, the concept is first described and then realized. Finally, how such Abacus type electronic cash system be more efficient than the previous versions is discussed.

6.1 Introduction

In recent years, many researches on the aspect of electronic coin divisibility have been conducted by using different cryptographic techniques [Pai93, Bra94,

DC95, EO94, OO90, OO92, Oka95]. Okamoto presented a "single-term divisible" scheme in [Oka95], which was the most efficient one among those papers. A "Single-term" cash scheme [CFN90, EO94, Oka95] means that cash consists of a single term where cut-and-choose method is not used. Divisibility [OO92] refers to a piece of electronic cash in a given amount can be subdivided into many pieces such that each subdivided piece is worth any desired value less than the original cash and the total value of all pieces is equivalent to the original one. The scheme consists of four main protocols:

a) Opening Protocol

b) Withdrawal Protocol

c) Payment Protocol

d) Deposit Protocol

Among these four protocols, the "opening protocol" is very clumsy and it degrades the overall performance of this scheme. Although the opening protocol is claimed to be invoked less frequently, it compromises the security concern definitely by changing the *license* of the user after a longer period of time.

Opening protocol is proceeded when a customer is opening an account. Firstly, Okamoto proposed to use bit commitment scheme in realizing single-term property at this opening stage. This scheme required many repeated challenge-response type data transfer and computation. This awkward "CHECK MOD-MULT" Protocol slowed down the process seriously.

On the other side, Binary Tree approach provide efficient capability of dividing the electronic coins but still, large demanding of memory size to memorize

all used nodes and large data transfer sizes for each transaction hinder its advocation.

Electronic cash system efficiency can be defined as:

Protocol Efficiency a terse protocol provides fast data transfer and short computation lead-time.

Resources Efficiency small electronic cash size requires small memory size to store the coins.

In next section, an **Abacus model** is first set up and it is shown that how an Abacus structure is an optimized adaptation to our daily cash usage. Then, an explanation of how it can improve the opening protocol is given. This solves the shortcomings mentioned above and is more efficient than the previous schemes. The key technique is RSA digital signature and encryption algorithm. The security of this scheme relies on the difficulty of factoring.

6.2 Abacus Model

Spending Behavior For daily use, large quantity of small denomination coins are required for low value transactions. To optimize this kind of usage, we should have adequate "small coins" (coins with small denomination) for each payment.

Concept Abacus approach stores the withdrawn cash in a way that it has already been broken into pieces in an appropriate extent just like "small coins" in your electronic wallets when the cash is withdrawn from a bank and those coins can be used directly for a payment without requiring further subdividing.

In other words, abacus approach stores a withdrawn amount in customer's wallet as a set of small coins which are ready to use without any extra effort to divide it into smaller pieces.

An Abacus is a **coin-rack** with many **shelves** to store the coins with different denominations. It is analogous with an abacus, which is a frame with beads that slide along parallel rods, used for counting. One Abacus shelf is equivalent to one parallel rod. Coins in the same shelf denote the same denomination. There is no fixed size for each shelf but the number of coins in the shelves is managed in a *pyramid* style with more "small coins" allocated at lower shelves of the coin-rack and less "large coins" at higher shelves. By using this approach, many coins are located at lower shelves (lower values). An Abacus shelf structure can be defined as below:

Abacus Shelf Structure Let the abacus has T shelves. The denomination with respect to **minimum divisible unit** (MDU) of shelves i ($1 \leq i \leq T$) is denoted as d_i. MDU is the smallest denomination that we can use in this money system (e.g. 1 cent). Shelf 1 (the lowest shelf) contains the MDU coins and higher shelves contain larger denomination coins. Shelf i has an all-one bit string, t_i, with c_i bits in length where c_{i+1} should never larger than c_i. Each bit of t_i denotes one coin with denomination d_i. When coins are going to be added onto the rack, a set of rules should be followed as below:

Abacus Coin Allocation Rule Coins are allocated to the highest shelf first and then packed down to the lowest MDU shelf, shelf 1. Let the divisibility precision is N where N = (the total coin value)/(minimum divisible unit value). The total amount is first divided into 2 halves and padded one '1' bit to the

Shelf No.	Denomination, d	Bit String, t	Shelf Amount
6	50	1	50
5	20	1	20
4	10	1	10
3	5	11	10
2	2	11	4
1	1	111111	6

Table 6.1: An Abacus stores \$100 worth coins.

highest shelf, t_T whose denomination d_T should be just less than $\lfloor N/2 \rfloor$. Then, d_T is deducted from N and the coin is further divided into halves and the half is compared with the denomination of the shelf immediately below the previous level, we denote the denomination of this level as d_{T-1}. If the denomination is less than the value of the half, one '1' bit is padded to this shelf and the half is then deducted by the corresponding d_{T-1}. '1' bit is padded continually into this level and the half is further deducted by d_{T-1} until it cannot be deducted. Table 6.1 shows an Abacus in which a total amount of \$100 coins are residing. For simplicity, I choose \$1 as the MDU in this example and thus $N = 100$.

It is noticed that there are a lot of small coins that are convenient for daily low value transactions. For example, if a payment requires \$13, the payer chooses one \$10 coin, one \$2 coin and one \$1 coin for this payment. These invoke transmissions of $t_4(1)$, $t_2(1)$ and $t_1(1)$ to the payee where $t_i(1)$ means that the content of bit one of that bit string, t in shelf i $(1 \leq i \leq 6)$. This procedure is carried out until no combinations of the coins are available for a payment. As previous example, if the coins on the Abacus have almost been used up and only one \$50 coin left and if the payer needs to pay \$1 to make a phone call at that time, subdivision of the \$50 coin is required. In the next section, a divisible

Abacus approach is proposed.

6.3 Divisible Abacus Electronic Coins

As mentioned in the last part of previous section, subdivision of large value coins is necessary when all small coins are used up. In this section, the tree approaches are presented, which can be imported to Abacus scheme to achieve a divisible Abacus electronic cash system. Basically, a tree approach is to divide a coin value into many small pieces and distribute the pieces as nodes on a tree structure. A tree consists of a node called the **root** together with multiple branches and ends up with many **leaves**. The root contains the value of the coin and the direct descendant nodes break down the value in some manner provided that the sum of these nodes is equal to the root value. In the following two subsections, two types of tree approaches are described and analyzed.

6.3.1 Binary Tree Abacus Approach

Binary Tree (BT) approach was adopted for divisible cash scheme [DC95, EO94, OO92, Oka95, Pai93] before. Each coin of worth $w = 2^l$ is associated with a tree of $(l + 1)$ levels and w leaves. Each node of the tree represents a certain denomination. Also, each node has two sons and a unique root node exists at the top of the tree. In general, w needs not to be equal to 2^l. This induces a situation that the value of a node is not divisible by 2. Then, the two corresponding sons would have different values whose difference is equal to 1. Figure 6.1 shows a BT of \$5, which is divided to 5 \$1 leaves.

This tree is constructed with 4 levels by computing $l = (\lceil \log_2 N \rceil + 1)$,

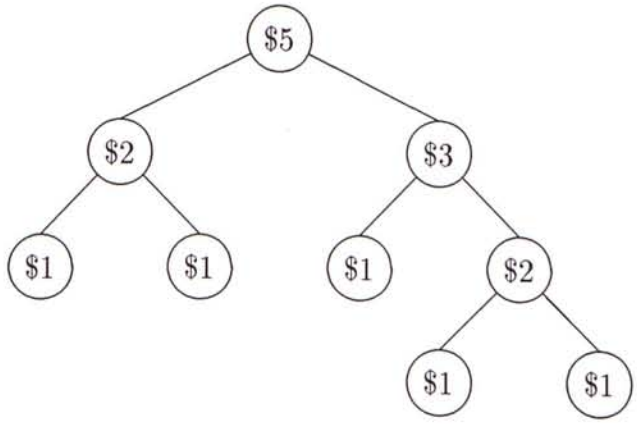

Figure 6.1: A Binary Tree denoting $5.

where N is the divisibility precision. In order to prevent over-spending and to guarantee the total value of spent nodes would not be greater than the coin worth value, the following restrictions and rules must be obeyed:

1) Value of a node, o, is the total of that of the direct sons of o.

2) **Route Node Rule**: When a node is used, all descendant nodes and all ancestor nodes cannot be used.

3) **Same node Rule**: No node can be used more than once.

Figure 6.2 shows a BT of $50 coin. It has 7 levels and 50 leaves. The total number of nodes of this BT is 99.

6.3.2 Multi-tree Approach

Binary Tree approach is a special case of Multi-Tree approach. Alternatively, we can say that **Multi-Tree** (MT) approach is a general prolongation from BT

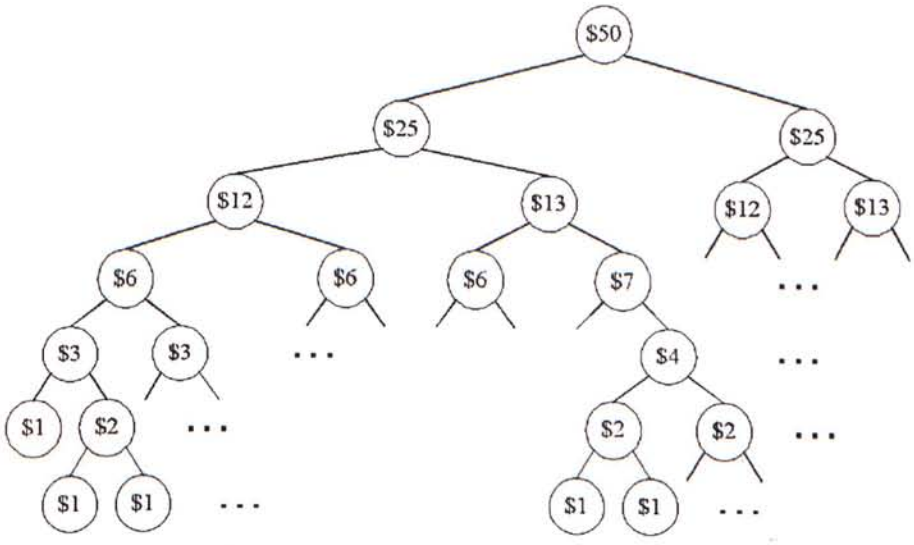

Figure 6.2: A Binary Tree approach of a $50 coin.

approach. Therefore, the restrictions and rules of BT also apply to MT. As BT, the total number of levels of a MT-n is $l_n = (\lceil \log_n N \rceil + 1)$, where n is the number of branches for each node and N is divisibility precision. Then, how many branches it should be required for best daily usage? This question induces the following argument:

> If more branches are available for each node, each parent node is subdivided to many sons. Thus, the node value on each level is decreased rapidly to the minimum divisible unit and the number of levels is decreased also. These cause the total number of nodes decreased. However, if there are too many branches, more tokens (nodes) may be needed to add up for one payment because the value of each token may be small. This degrades the payment protocol's efficiency. On the contrary, if fewer branches are available for each

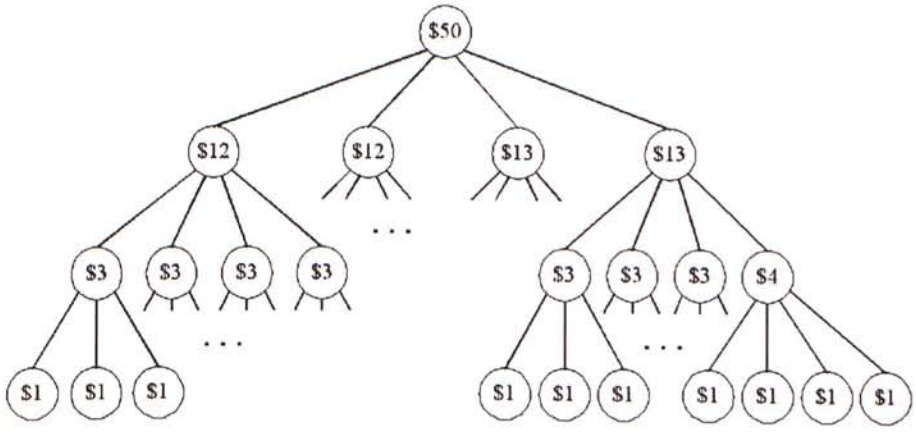

Figure 6.3: A MT-4 approach of a $50 coin.

node, BT is a MT-n with $n = 2$ as an extreme case. It shows later in this paper that it achieves the least average tokens required for one payment but the coin size is larger.

There are altogether n^{i-1} nodes at ith level ($i = 1, \cdots, l$). As the previous example, if the coin worth $50 and a MT-4 of this coin will have 4 levels, $l_4 = (\lceil \log_4 50 \rceil + 1) = 4$, and 50 leaves. The total number of nodes of this MT is 71. Comparing with that of BT, $l_2 = 7$ levels, BT needs 3 more levels to represent the same coin. Figure 6.3 shows this MT-4 approach of $50 coin.

6.3.3 Analysis

When considering the total number of nodes, BT has 99 nodes for $N = 50$ while MT-4 has 71 nodes. It can also be shown that when n increases, fewer nodes are required for each coin. The lower bound of the total number of nodes is N when $n = N$. The tree approaches such as BT and MT are both suitable to use with Abacus because all such tree structures can provide the maximum number of

minimum divisible unit (MDU) tokens (BT and MT-4 have the same number of leaves). For example, if a coin worth \$50 and the MDU is \$1, it is shown above that both BT and MT-n can provide 50 \$1 tokens. Larger MT-n achieves this with fewer nodes and fewer levels to yield fewer total numbers of nodes. This can save the storage size of the coin.

We noticed as above that the MT-n Abacus requires less resources in storing coins. It provides better resource efficiency. For protocol efficiency, the average number of tokens required to transmit per payment is considered. During a payment, the minimum number of nodes used is 1, for both BT and MT-4, that is when the whole coin being used once and for all. For computing the maximum possible number of nodes used for one payment, the maximum number of nodes per level used is $(n-1)$ where n is the number of branches for each node and the maximum number of levels used is $(l-1)$. Thus, the maximum possible number of nodes used in one payment is $(n-1)(l-1)$. The average number of nodes used is $[(n-1)(l-1)+1]/2$. For BT, the average number of nodes used is 4 if $N = 50$ while for MT-4, the number is 5. Thus, the data transfer of BT approach is more efficient than that of a MT-n approach for $n > 2$ during data transfer by transmitting the minimum number of tokens per payment.

Abacus can be used in conjunction with tree structure to achieve divisible electronic cash system. This approach is very efficient by following the following rule:

> Use all available coins on the abacus table for all payments whenever possible. If there are not enough available coins without subdividing for a payment, BT or MT-4 is then employed to divide large denomination coins.

In the next section, I'll show how the Abacus be implemented in an electronic cash system.

6.4 Abacus Electronic Cash System

In this section, the basic protocols in the abacus electronic cash system will be introduced. The definition of the system is essentially based on those of [CFN90, OO92, Oka95]. The electronic cash system consists of the following parts.

1. **Opening Account**: A customer, Alice, obtains an electronic license from the bank, B. This part is analogous to account opening in a bank. An efficient opening protocol is shown below which uses cut-and-choose methodology.

2. **Cash Withdraw**: Alice withdraws a certain amount of electronic cash from her bank account.

3. **Payment**: Alice uses electronic cash to pay for goods in Bob's shop.

4. **Deposit**: Bob stores electronic cash to his bank account.

6.4.1 Opening Protocol

As in [OO92] and [Oka95], the electronic cash consists of an **electronic license** and electronic coins. Alice opens a bank account and obtains the electronic license (N, L) where

$$L = (N + a)^{1/K} \bmod n \tag{6.1}$$

in which (n, K) is the RSA [RSA78] public key of the bank, B; a is another B's public key for blind signature [Cha83, Cha88] and N is a Williams integer (see section 2.7). In this protocol, B signed the electronic license blindly to ensure N and L are kept secretly from B and to maintain the identity of Alice untraceable. Also, cut-and-choose methodology is employed to let Alice prove B that N is honestly generated without revealing any of her identity information on N and L. On the other side, L is the signed version of N such that it cannot be created by others but only can the bank. All others can verify it by using B's public key, K.

This opening protocol is shown as below:

Step 1. Alice prepares the Williams integers N_i from two large prime factors P_i, Q_i for $N_i = P_i Q_i$, where $P_i \equiv 3 \pmod 8$ and $Q_i \equiv 7 \pmod 8$, for $i = 1, \cdots, k$ where k is a positive integer as B's secure parameter (e.g. $k = 100$).

Step 2. For each pair of P_i and Q_i, Alice gives B

$$x_i = g^{P_i} \bmod \Phi \tag{6.2}$$

$$\text{and} \quad y_i = g^{Q_i} \bmod \Phi \tag{6.3}$$

as Alice's identity. B cannot reveal P_i or Q_i from x_i and y_i. The security comes from discrete logarithm problem (see section 2.10). Then Alice computes and sends to B

$$s_i = (N_i + a) r_i^K \pmod n \tag{6.4}$$

where r_i is the blinding factor chosen randomly.

Step 3.

B selects a random subset of $(k-1)$ distinct blind candidates indices $U = \{i_j\}$, where $(1 \leq i_j \leq k)$ for $j = 1, \cdots, (k-1)$ and asks Alice to open the corresponding blind candidates.

Step 4. Alice compiles by giving (P_i, Q_i) and r_i for all i in U to B.

Step 5. B checks $g^{P_i} \bmod \Phi \equiv x_i$ and $g^{Q_i} \bmod \Phi \equiv y_i$. Then B computes and checks if $P_i Q_i = N_i$ and $s_i \equiv (N_i + a) r_i^K$.

Step 6. From this point, only s_{i_k} has not been revealed. If all the above checks are positive, B signs s_{i_k} as

$$s_{i_k}^{1/K} \equiv (N_{i_k} + a)^{1/K} r_{i_k} \pmod{n}. \tag{6.5}$$

This technique employs the multiplicative property of RSA where $n = pq$ and p, q are large prime integers.

Step 7. Alice extracts the signed license as

$$L = (N_{i_k} + a)^{1/K} \pmod{n} \tag{6.6}$$

by dividing the blinding factor, r_{i_k}. For simplicity, we ignore the foot notation of N hereafter.

In this protocol, k sets of (s_i, x_i, y_i) and $(k-1)$ sets of (P_i, Q_i, r_i) are submitted to B.

6.4.2 Withdrawal Protocol

After Alice opens an account at the bank, B, she can withdraw electronic cash from B and stores in her electronic wallet. The principle of the withdrawal protocol can be defined as the idea below:

Idea Alice withdraws a certain amount of money from bank, B. The money consists of a set of coins with different denominations that are allocated in accordance with **Abacus coin allocation rule**. The smallest denomination of a coin is the MDU (e.g. 1 cent). During payment, the best combination of the coins is to choose with reference to the guideline stated in section 6.3.3. When "small coins" are not enough, *refill* is required. Refill is conducted by subdividing larger value coin into many small pieces (tokens) such that each subdivided piece is worth any desired value less than the original coin and the total value of all tokens is equivalent to the original coin value. BT or MT-n approaches can be employed.

The withdrawal protocol is shown as below:

Step 1. Alice constructs an Abacus rack or table, Ξ according to **Abacus coin allocation rule** stated in section 6.2 for an amount of $\$w$. Ξ has T shelves and the denomination with respect to MDU of shelf i ($1 \leq i \leq T$) is d_i. Each shelf has an all-one bit string, t_i, with c_i bits in length where c_{i+1} should never be larger than c_i. For each bit, a corresponding random value b_{ij} is chosen, where $j = 1, \cdots, c_i$. A total of Ψ coins are requested to issue where

$$\Psi = \sum_{i=1}^{T} c_i \tag{6.7}$$

$$\text{and} \quad w = \sum_{i=1}^{T} c_i d_i. \tag{6.8}$$

Step 2. Alice sends Ξ and D_{ij} to B for

$$D_{ij} = r_{ij}^{e_i} H(N \| b_{ij}) \bmod n_i. \tag{6.9}$$

where $r_{ij} \in \mathbb{Z}_{n_i}$ is a random integer, H is a one-way hash function, (e_i, n_i) is B's RSA public key, which corresponds to the abacus shelf denomination, d_i. The total number of D_{ij} prepared by Alice is Ψ.

Step 3. B gives $D_{ij}^{1/e_i} \bmod n_i$ to Alice and charges Alice's account $\$w$.

Step 4. Alice extracts the coins

$$C_{ij} = (H(N \| b_{ij}))^{1/e_i} \bmod n_i. \tag{6.10}$$

Alice can withdraw any amount of money, $\$w$, she likes. The abacus rack splits this amount to some coins with well-defined denominations. Cut-and-choose protocol is not used here and B only need to send back Ψ blindly signed coins to Alice.

6.4.3 Payment and Deposit Protocol

There are two conditions during a payment:

1. **Direct Pickup**: The coins are available right on the abacus rack. No subdivision of the coins is required.

2. **Break Down**: When there is no available coin on the abacus rack, subdivision of the large coin is required.

These two conditions can be happened in a single payment as well. In condition 1, it is equivalent to the spending of root node of a coin in tree structure approach. While in condition 2, it is equivalent to the spending of nodes of a coin. Therefore, these conditions can be concluded as standard tree structure approach. For each coin no matter what condition it is, two stages are carried out, those are **coin authentication** and **denomination revelation**. The followings show the two procedures. Please note that the notations have been altered from [Oka95] slightly to cope with multiple coins.

Coin Authentication

1) Alice sends her identity (N, L) and coins (C_{ij}, b_{ij}) where $i \in \{1, \cdots, T\}$ and $j \in \{1, \cdots, c_i\}$ to the merchant, Bob.

2) Bob checks that $L^K \equiv N + a \pmod{n}$ and $C_{ij}^{e_i} \equiv H(N \| b_{ij}) \pmod{n_i}$.

Denomination Revelation Let f_Γ, f_Λ and f_Ω are three randomized hash functions. Alice memorizes all the nodes already spent for each coin. Since direct pickup of a coin is equivalent to spending the root node of the coin, it is adequate to describe the procedure of node payment below.

Let Alice selects a node $n_{j_1 \cdots j_t}$ ($j_k \in \{0, (q-1)\}$, $k = 1, \cdots, t$ and q is the number of branches of a multi-tree structured electronic coin, which has a total of l levels). When Alice spends this node, the following payment protocol will be carried out.

Step 1. Alice computes

$$[(\langle(\Omega_{j_1 \cdots j_{t-1}})^{2^{t-1} j_t} (\Omega_{j_1 \cdots j_{t-2}})^{2^{t-2} j_{t-1}} \cdots (\Omega_{j_1})^{2 j_2} \times f_\Gamma(C \| 0 \| N)\rangle_{QR})^{1/2^t} \bmod N]_{-1} \tag{6.11}$$

where

$$\Omega_{j_1\cdots j_i} = \langle f_\Omega(C\|j_1\|\cdots\|j_i\|N)\rangle_1 \qquad \text{for } i = 1,\cdots,t-1. \quad (6.12)$$

Step 2. Bob computes $\Omega_{j_1\cdots j_i}$ when $j_{i+1} = 1$ $(i = 1,\cdots,t-1)$ and verifies that $(\Gamma_{j_1\cdots j_i}/N) \equiv -1$ and

$$(\Gamma_{j_1\cdots j_t})^{2^t} \equiv d(\Omega_{j_1\cdots j_{t-1}})^{2^{t-1}j_t}(\Omega_{j_1\cdots j_{t-2}})^{2^{t-2}j_{t-1}}\cdots(\Omega_{j_1})^{2j_2} \times f_\Gamma(C\|0\|N) \pmod N \quad (6.13)$$

where $d \in \{\pm 1, \pm 2\}$.

Step 3. If all correct, Bob chooses a random value $e' \in \{0,1\}^u$, and sends his identity ID, time δ, and e' to Alice, where $u = O(m)$, $m = |P|$. Bob computes $e = h(ID\|\delta\|e')$ where $e \in \{0,1\}^u$ and h is a randomized hash function. This is to prevent the bank, B from crediting an invalid shop's account.

Step 4. Alice computes e accordingly and then computes $\Lambda_{j_1\cdots j_t}$ such that

$$(\Lambda_{j_1\cdots j_t})^{2^{u+1}} \equiv 2^{2e}\langle f_\Lambda(C\|j_1\|\cdots\|j_t\|N)\rangle_{QR} \pmod N \quad (6.14)$$

Step 5. Bob verifies that

$$(\Lambda_{j_1\cdots j_t})^{2^{u+1}} \equiv d'2^{2e}f_\Lambda(C\|j_1\|\cdots\|j_t\|N) \pmod N \quad (6.15)$$

where $d' \in \{\pm 1, \pm 2\}$.

When Bob deposits these coins and tokens to his bank account, he simply forwards all the history of the transaction that conducted with Alice to the bank.

6.5 Anonymity and System Efficiency

This protocol guarantees customer's anonymity. The customer is protected by blind signature during money withdrawal. The bank cannot trace the withdrawn money notes since the customer's identity P and Q are concealed. This security is equivalent to **discrete log problem** (see section 2.10). At the same time, the bank can be ensured that the customer generates his identity honestly by using **cut-and-choose methodology** (see section 2.13). The secure parameter, k, is large enough (e.g. $k = 100$) that the odds of having a ruse to pass the bank are 1 in k. During payment, the honest customer only uses the money note ONCE, the merchant and the bank even collusion between them cannot reveal the customer's identity. On the other side, if the customer double spending or over spending the electronic cash, there is method for the bank to reveal his identity efficiently. It is even much easier than checking the fingerprints on a real money note. However, linkability between the payments exists which let the bank or third party is able to trace the spending route of the electronic coins.

In the system efficiency issue, Abacus electronic cash scheme acquires the properties of Abacus model and tree structures. As an example, if Alice make a series of payments to various kinds of shops in a sequence like this: \$5, \$7, \$23, \$12, \$1, \$40, \$7 and \$5. She withdraws \$100 worth coins from the bank. Let the MDU is \$1 for simplicity. We can treat those payments as a sample sequence such as {5,7,23,12,1,40,7,5}. By using BT approach, 15 tokens are required to transmit for these 8 payments. If Abacus MT-4 is used, 12 coins (direct pickup) and 9 tokens (break down) or 7 tokens, if maximized, are required to transmit. If Abacus BT is used, 12 coins (direct pickup) and 7 tokens (break down) or

6 tokens, if maximized, are required to transmit. Since the number of coins breaking down is decreased for Abacus approach, less computation is required for the payments and this yields a higher efficiency.

Chapter 7

Conclusions

In this report, two new fundamental concepts are proposed which give new directions on the development of untraceable divisible off-line electronic cash. Unit-of-time spending limit enforced electronic cash is the first concept and the interest-bearing electronic cash is the second break-through concept. In addition, these two concepts are also realized with two compatible schemes. Direct construction technique and secret-sharing line approach are elaborated in details. Further work can also be done be employing these two concepts to tree structured electronic cash approach. It is also shown that these new concepts can bring out many practical applications and still more are going to be emerged.

On the other side, the Abacus type divisible cash structure has been introduced. Implementation of an Abacus type divisible untraceable brokerless electronic cash scheme and improvements on opening protocol have been demonstrated. It is shown that the Abacus type electronic cash system is more efficient

than the previous proposed systems. Abacus approach splits the money to various smaller coins, which is good for daily use especially for the small transactions. Furthermore, Abacus is very flexible to use with tree structures (both Binary Tree and Multi-tree structures) to achieve divisibility and to further compress the coin size. On the system efficiency aspect, the crux of the Abacus electronic cash system is the improvement on both protocol efficiency and resources efficiency. By considering the protocol efficiency, Abacus type electronic cash scheme gives an efficient opening protocol, which is practical for daily low transaction applications. For Abacus approach used with Binary Tree, fewer tokens in average are required to transfer for each payment than those used in Multi-Tree. It is good for application whose data transfer protocol is carried on narrow bandwidth link (e.g. smart card application). When considering the resource efficiency, Abacus table structure in conjunction with Multi-Tree approach provides the smallest coin size among the alternatives presented and also optimizes each payment activities by minimizing the computational requirement.

Finally, we turn our vision to look into the relationship between electronic cash and physical cash. It's already realized to be awkward to pay for the services or goods provided on computer networks by checks or paper bills. People are already giving up the use of physical cash on the Internet and also in the meantime, they are getting acquainted in using various kind of electronic means such as electronic purse cards, prepaid phone cards and even electronic cash. With the further improvements of technologies and boosting up of population on ordering goods and services through the global communications networks with the diversifying needs, it is not only fantastic but practical of this new *cyber* society being make the physical cash obsolete soon.

Appendix A

Internet Payment Systems

Figure A.1 shows the 4 kinds of Internet payment systems that are going to be covered in this annex.

A.1 Bare Web FORM

By using only **FORM** tag of Hypertext Markup Language (HTML)[1] and simple Common Gateway Interface (CGI)[2] scripts, we can create our first online cyber-shop cashing counter[3]. Figure A.2 shows the system's working principle:

Step 1. A buyer downloads the HTML file containing payment form from the shop's homepage (Form.html). This HTML file contains a pair of FORM tag which has a hypertext link referring to the address of a CGI program.

Step 2. The buyer then fills in his card number, expiration date and other

[1] www.w3.org/TR/REC-html32, and
www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html
[2] www.w3.org/CGI/Overview.html
[3] www.isoc.org/individual-join.html

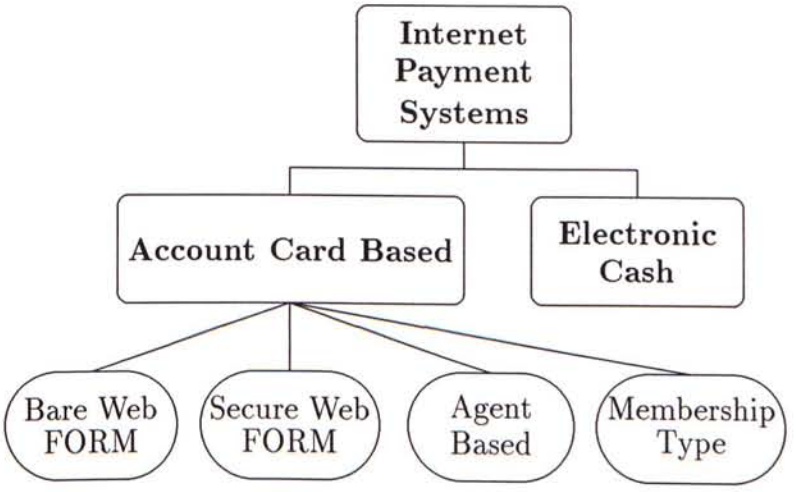

Figure A.1: Various Internet Payment Systems

required information and press a "send" button to send back the form.

Step 3. The buyer's information will then be routed to the address of the CGI program.

Step 4. The CGI program parses the buyer's card information, checks for validity and informs other components of the cyber-shop to dispatch something that the buyer requested to buy.

Usually the CGI program would not perform online card authentication for cost consideration. Card authentication is performed in batch mode. For example, it is carried out once per day. Thus, this payment system is usually used by merchants dealing with goods which require shipping to their customers.

On the other side, card information is transmitted through Internet *insecurely* by using HTML's FORM tag which in turn is riding over Hypertext

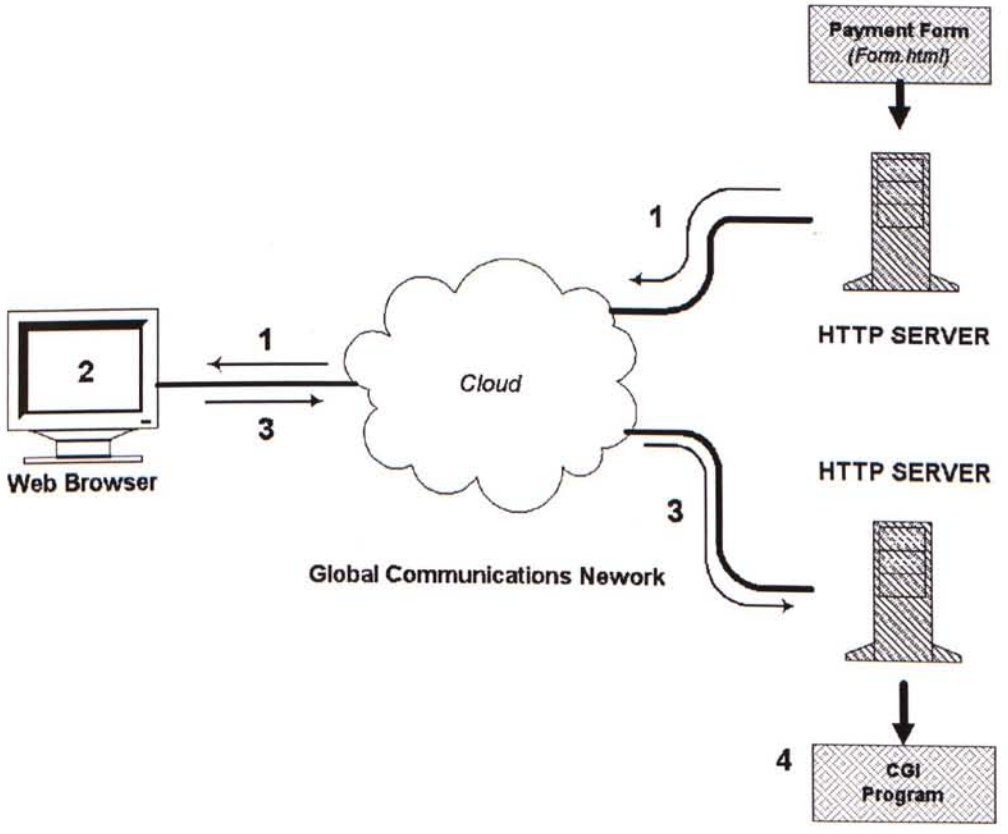

Figure A.2: Interactions between HTML FORM and CGI Program

Transfer Protocol (HTTP)[4]. Cyber eavesdroppers can get your card information easily from the bare HTTP messages. One criminal scenario can be described as an intruder who is either a router's administrator or be able to crack into the router such that he can access all the raw data flowing through this router. If this router is the only gateway for a merchant's web server to connect to the outside world, all buyer's card information has to travel through this router before reaching the merchant's server. The intruder can therefore easily obtain all buyers' card information easily by parsing the HTTP messages. This is a generic attack called *man in the middle.*

A.2 Secure Web FORM Payment System

This payment system is an extension of the previous one. It is similar to the bare Web FORM approach, but there is a *secure channel* being enforced for carrying data transmitted between the merchant and the buyer. Commonly, it is a set of protocols residing on top of transport layer. Data to be transmitted is first *encrypted* at this security reinforced layer before passing down to transport layer and then to other lower layers. This kind of payment system is now used widely on the Internet.

One of the most popular protocols to provide the secure channel is called **Secure Sockets Layer** (SSL)[5] which is first proposed by Netscape Communications Corporation. By now, SSL is supported by almost all Graphical User Interface (GUI) based World Wide Web (WWW) browsers including Netscape Navigator and Communicator, Microsoft Internet Explorer and America Online

[4]www.w3.org/Protocols

[5]www.netscape.com/eng/security

Web Browser.

Besides solely supporting HTTP, SSL is also extending itself to many other aspects. For example, it allows client/server (e.g. telnet and ftp)[6] applications to communicate in a way that prevents eavesdropping, tampering or message forgery.

A.3 Membership Type Payment System

The above two payment systems require buyer's credit card information to transmit over Internet directly. From this system onward, I describe two other types of payment systems which require no card information wandering on Internet.

Membership type payment system[7] requires each buyer to supply his card information via phone or postal services to merchant in advance. The merchant will then issue ***userid*** and ***password*** to the buyer. This initial step is called *Phase of Registration.* During purchasing, the buyer sends his userid and password to the merchant, which are encrypted for transmission by a proprietary system.

Although the card number is never transmitted over Internet, buyer's convenience is compromised with an additional phase of registration. Also, each pair of userid and password is only good for one merchant and is persistent. For an online shopping mall with a huge number of shops, this system will become too clumsy to be applied.

[6] www.cryptsoft.com
[7] www.netmarket.com

A.4 Agent Based Payment System

To tackle the last problem in previous system, the agent based payment system makes a little modifications from the previous one.

During the phase of registration, buyer gives his card information via phone to an *agent* instead of individual shop. The agent sends back a ***userid*** via Email. To buy things, buyer sends his userid to the shop who accepts the agent's userid. The userid is sent unencrypted through bare Web FORM. After the shop receives the userid, it passes the userid to the agent with the buyer's ordering details. The agent will then sends the buyer an email to confirm the purchase.

Userid is on longer stuck on a single merchant and every purchase is further confirmed by the buyer via email. That is also why the userid is not required to keep secret during transmission. Another extra feature is that each buyer has to have an email account. More information can be obtained online from First Virtual Holdings Incorporated's homepage[8].

A.5 Internet-based POS

This system is more or less similar to our conventional Point-of-Sale (POS) system in the shops. It provides online credit card authentication. Great advantage is noticed on instantly providing goods or services requested by the buyer such as downloading softwares or documents. This system requires each buyer to install proprietary software on his machine and each merchant has to install an

[8]www.fv.com

Internet-based POS system to communicate with an online credit card authentication server. The buyer's software is responsible for sending the buyer's card information to the online POS securely. Details of the system can be obtained from CyberCcash's homepage[9].

[9]www.cybercash.com

Appendix B

Papers derived from this thesis

[1] S. Wong, Victor K. W. Wei. Unit of time spending limit enforced single term off-line coins. Submitted to *Electronic Letters.*

[2] S. Wong, Victor K. W. Wei. A method for imposing spending limit on electronic coins. Submitted to *1998 IEEE International Symposium on Information Theory.*

[3] S. Wong. Abacus type electronic cash. Submitted to *the special issue on Electronic Commerce and the Internet of the Telecommunication Systems.*

Bibliography

[Bra94] Stefan Brands. Untraceable off-line cash in wallets with observers. *Advances in Cryptology - CRYPTO '93*, pages 302–318, 1994. FTP site: ftp.cwi.nl/ftp/brands/crypto93.ps.

[CFN90] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. *Advances in Cryptology - CRYPTO '88*, pages 319–327, 1990.

[Cha83] David Chaum. Blind signature for untraceable payments. *Advances in Cryptology - CRYPTO'82*, pages 199–203, 1983.

[Cha85] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, Oct 1985.

[Cha88] David Chaum. Blinding for unanticipated signatures. *Advances in Cryptology - EUROCRYPT '87*, pages 227–233, 1988.

[Cha90] David Chaum. Online cash checks. *Advances in Cryptology - CRYPTO '89*, pages 288–293, 1990.

[Cha92] David Chaum. Achieving electronic privacy. *Scientific American*, pages 96–101, Aug 1992.

[CLS94] William Caelli, Dennis Longley, and Michael Shain. *Information Security Handbook.* Macmillan Publishers, 1994.

[CP92] David Chaum and Torben Pryds Pedersen. Transferred cash grows in size. *Advances in Cryptology - EUROCRYPT '92*, pages 390–407, 1992.

[Dam90] Ivan Bjerre Damgard. Payment systems and credential mechanism with provable security against abuse by individuals. *Advances in Cryptology - CRYPTO '88*, pages 328–335, 1990.

[DC95] Stefano D'Amiano and Giovanni Di Crescenzo. Methodology for digital money based on general cryptographic tools. *Advances in Cryptology - EUROCRYPT '94*, pages 156–170, 1995.

[DH76] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, Nov 1976.

[ECo] Electronic commerce news. Periodical.

[ElG85] T. ElGamal. A public-key cryptosystem and a signature scheme based on discrete logarithms. *Advances in Cryptology - CRYPTO '84 Proceedings*, pages 10–18, 1985.

[EO94] Tony Eng and Tatsuaki Okamoto. Single term divisible electronic coins. *Advances in Cryptology - EUROCRYPT '94*, pages 306–319, 1994.

[Fer94] Niels Ferguson. Single term off-line coins. *Advances in Cryptology - EUROCRYPT '93*, pages 318–328, 1994.

[For97] Warwick Ford. *Secure electronic commerce: building the infrastructure for digital signatures and encryption.* Prentice Hall, 1997.

[Kim91] Paul Kimberley. *Electronic Data Interchange.* McGraw-Hill, Inc., 91.

[Kos97] David Kosiur. *Understanding Electronic Commerce.* Microsoft Press, 1997.

[Lyn96] Daniel C. Lynch. *Digital money: the news era of Internet commerce.* Wiley, 1996.

[MC93] Albert J. Marcella and Sally Chan. *EDI security, control, and audit.* Artech House, 1993.

[Men93] Alfred J. Menezes. *Elliptic curve public key cryptosystems.* Kluwer Academic Publishers, 1993.

[Mil86] Victor S. Miller. Use of elliptic curves in cryptography. *Advances in cryptology - CRYPTO '85 proceedings*, pages 417–426, 1986.

[Mil88] Roger LeRoy Miller. *Economics Today*, pages 327–345. Harper & Row, Publishers, Inc., 6 edition, 1988.

[MV90] Alfred Menezes and Scott Vanstone. The implementation of elliptic curve cryptosystems. *Advances in Cryptology - AUSCRYPT '90*, pages 2–13, 1990.

[MVOV97] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography.* Boca Raton, Fla.: CRC Press, 1997.

[Niv72] Ivan Morton Niven. *An Introduction to the Theory of Numbers.* New York: John Wiley & Sons, 3rd edition, 1972.

[Oka95] Tatsuaki Okamoto. An efficient divisible electronic cash scheme. *Advances in Cryptology - CRYPTO '95*, pages 438–451, 1995.

[OO90] Tatsuaki Okamoto and Kazuo Ohta. Disposable zero-knowledge authentications and their applications to untraceable electronic cash. *Advances in Cryptology - CRYPTO '89*, pages 481–496, 1990.

[OO92] Tatsuaki Okamoto and Kazuo Ohta. Universal electronic cash. *Advances in Cryptology - CRYPTO '91*, pages 324–337, 1992.

[Pai93] Jean Claude Pailles. New protocols for electronic money. *Advances in Cryptology - AUSCRYPT '92*, pages 263–274, 1993.

[PO95] Michael Peirce and Donal O'Mahony. Scaleable, secure cash payment for www resources with the payme protocol set. *Fourth International World Wide Web Conference*, Dec 1995.

[PW92] Birgit Pfitzmann and Michael Waidner. How to break and repair a "provably secure" untraceable payment system. *Advances in Cryptology - CRYPTO '91*, pages 338–350, 1992.

[Rie85] Hans Riesel. *Prime numbers and computer methods for factorization.* Birkhauser, 1985.

[RSA78] Ronald Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21:120–126, Feb 1978.

[Sch94] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C.* John Wiley & Sons, Inc., 1994.

[Sha78] Daniel Shanks. *Solved and Unsolved Problems in Number Theory.* New York: Chelsea Pub. Co., 2nd edition, 1978.

[Sha79] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, Nov 1979.

[Sim92] Gustavus J. Simmons. *Contemporary Cryptology: The Science of Information Integrity.* IEEE Press, 1992.

[Sti95] Douglas Robert Stinson. *Cryptography: theory and practice.* CRC Press, Inc., 1995.

[Tsi97] Yiannis S. Tsiounis. *Efficient Electronic Cash: New Notions And Techniques.* Phd thesis of the department of computer science, Northeastern University, Jun 1997.

[Wil80] H. C. Williams. A modification of the rsa public-key encryption procedure. *IEEE Transactions on Information Theory*, IT-26(6):726–729, 1980.

[ZO94] José Luis Zoreda and José Manuel Otón. *Smart Cards.* Artech House, 1994.