阴阳五行诀之一：根基济灵钥之钥

MEMORIZABLE PUBLIC-KEY CRYPTOGRAPHY (MePKC) & ITS APPLICATIONS

```
1111111
1166611
1611161
1166611
1116611
1116611
1116611
1116611
1666611
1116661
1666611
1111111
```

Mainly authored by Kok-Wah LEE (李国华) @ Xpree Li, plus

some ASCII arts of 2D key authored by Wei-Dong Chui (徐伟栋) & Wei-Jian Chui (徐伟坚)

Copyright Statement

The copyright of this book (aka "thesis" due to its role as a collection of statements and theories put forward and supported by arguments) belongs to the authors under the terms of the primary US Copyright Act in which it is firstly published online from a computer server located in the United States of America, secondly Malaysia Copyright Act 1987 due to author residency and citizenship, as well as tertiary Berne Convention for all its signatory countries for any international copyright entitlement.

Due to the relatively high research costs invested by the author, for refund, as well as for building up a fund for further maintenance, research and development, any original and novel idea conceptions from the author in this book is only free of usage for public interests, press report, private study, research, and teaching throughout the World, with the condition that proper originality citation for source references has been clearly shown. Yet for any commercial usage, prior consent has to be obtained from the author or his successor(s).

Email: E96LKW@hotmail.com
URL (h): www.geocities.com/xpree/
URL (o): www.xpreeli.com

DECLARATION

I hereby declare that the original and novel idea conceptions in this book of research essays towards a future possible doctorate thesis in information engineering, generally, and information security, particularly, have all been done by me, except Appendix B to have some new and creative child-made ASCII arts by Wei-Dong Chui and Wei-Jian Chui.

李国华 @ 李锦华
Kok-Wah Lee @ Xpree Jinhua Li
Find me Xpree or Xpreeli in the Internet!

Email (Home): E96LKW@hotmail.com
Email (Business): contact@xpreeli.com
URL (Home): www.geocities.com/xpree/
URL (Business): www.xpreeli.com

First unpublished draft on 25 October 2008 in Malaysia
First edition (version 1.0) of online publication on 14 March 2009 in the USA

ACKNOWLEDGEMENTS

I hereby would like to express my gratitude to the following persons together with their inputs that have been given me in completing the electronic book of this PhD research project, which has contributed mainly in the novel knowledge field of key/password security leading to the memorizable public-key cryptography (MePKC) and its applications. Here are the listees:

(i) My parents, relatives and friends together with those anonymous people

- For help and giving me the physical, emotional and spiritual supports.

(ii) Lake-Tee Khaw, University of Malaya, Kuala Lumpur, Malaysia

- For giving the advantages and disadvantages of SD (Statutory Declaration).

(iii) Gita Radhakrishna, Multimedia University, Melaka, Malaysia

- For supplying the legal contents about copyright.

(iv) Alan Wee-Chiat Tan, Multimedia University, Melaka, Malaysia

- For supplying C++ class of big number arithmetic after my given idea; and
- Being a nominal supervisor since 16 August 2008 till now.

(v) Voon-Chet Koo, Multimedia University, Melaka, Malaysia

- For prototyping the RJ45 switch using conventional DIP switch.

(vi) Hong-Tat Ewe, Multimedia University, Cyberjaya, Malaysia a

- For being a nominal supervisor since 27 May 2004 till 15 August 2008; and
- For having lots of unpleasant and trust-less interactions.

(vii) Matt Bishop, University of California at Davis, CA, USA

- For supplying a proceedings paper.

(viii) Alex X. Liu, Michigan State University, MI, USA

- For supplying a journal paper.

(ix) Chee-Onn Chow, University of Malaya, Kuala Lumpur, Malaysia

- For supplying a journal paper.

DEDICATION

特将这本研究论文书献给于我敬爱的父亲李厚芳和母亲徐亚妹。

This book of research essays is dedicated to my respected and beloved parents,
Hew-Fong Lee and Ah-Mooi Choi.

H-(^_^)-H

Find me Xpree or XpreeLi in the Internet!

CN: 语言与文字是了解一个文化的终极密码。

EN: Language and Writing are ultimate keys to understand a culture.

MY: Bahasa dan Tulisan adalah kunci terdasar to memahami satu ketamadunan.

🗝 🗝

ABSTRACT

The acquisition of a doctorate research degree is qualified by the sufficient and novel knowledge contribution from a PhD postgraduate student. Generally, the contribution of a scientific researcher is based on the publication counting, credit of journal and proceedings, citation counting, qualified peer evaluator, and contribution impact. Lately, D. L. Parnas (2007) called for a more accurate evaluation of researcher's contribution instead of the publication counting and citation counting. Parnas suggested that a researcher should provide some of one's main papers to be evaluated by some qualified peer evaluators. Here, proposal is delivered to improve Parnas' suggestion, where the researcher shall also list out the impacts of one's selected papers like the successful implementation and realization of a research result for public usage and application. This step of emphasizing researcher evaluation based on contribution impact is crucial to the welfare of human civilization to avoid the number game of discussing a war on some plain papers. Subsequently, this PhD project is carried out based on this evaluation principle of contribution impact, where the prototypes are expected to be realized and/or commercialized for public usages.

The main knowledge contributions of this research project are some key management techniques to create big and yet memorizable secret(s), especially 2D key, memorizable public-key cryptography (MePKC), and their applications in information engineering. Password the secret is the most popular, easily compatible, and cost-effective computer authentication method. In the current prior art, a memorizable secret reaches a maximum size at 128 bits (Schneier, 2006). This has limited the AES (Advanced Encryption Standard) key at 128 bits and storage technology of private key to be in the forms of encrypted private key, split private key (Ganesan, 1996, September 17), and roaming private key (Baltzley, 2000, November 28).

Here, the invented big secret creation methods and systems are (i) self-created signature-like Han character of CLPW (Chinese Language Password) and CLPP

(Chinese Language Passphrase), (ii) two-dimensional key (2D key), (iii) multilingual key, (iv) multi-tier geo-image key, (v) multi-factor multimedia key using software token, and their hybrid combinations. From every one of these inventions, the AES-256 key can be realized as well as the other applications in information engineering needing a big secret. Yet the most important contribution is the realization of fully memorizable private key towards the MePKC using finite field cryptography (FFC), elliptic curve cryptography (ECC), hyperelliptic curve cryptography (HECC), or torus-based cryptography (TBC). The explanation here adopts 2D key and multilingual key to create a fully memorizable private key for ECC. To resist the demand for longer key size due to the advancement of computing technologies, key strengthening is used.

Multimedia semantic noises are studied. Textual semantic noises like capitalization, punctuation marks, misspelling, mnemonic substitution, permutation, character stuffing, and ASCII mutual substitution table, are proposed to increase the randomness of CLPW, CLPP, and 2D key. The available styles of 2D key are multiline passphrase, crossword, ASCII art, Unicode art, etc.

Due to the technical and legal factors, there are different asymmetric key pairs for different cryptographic schemes. Hence, there are many private keys to be supported. From Forrester Research (Kanaley, 2001), an active Internet user manages an average of 15 keys on a daily basis. However, Adams and Sasse (1999) reported that users could only be expected to cope with a maximum of 4 or 5 keys that are unrelated and regularly used. If one key is used for two or more accounts, there is a risk of domino effect (Ives, Walsh & Schneider, 2004). Hence, the key management is a balance issue of usability and security. There exist some methods to generate multiple slave keys (aka site key) from one master key for multiple online accounts by using the domain name, random number, single sign-on server, and key strengthening. Here, some new key management methods are proposed to extend the generation of slave keys from one master key to both the online and offline accounts, which is called multihash key, by using domain name, key strengthening, and hash truncation. A brother of multihash key called multihash signature, using multiple hash values of a message from different hash iteration, is proposed to provide object-designated signature function. The object may be recipient, action, feature, function, meaning, etc., as a representation.

Annually, the World and USA lose USD$600 billion and USD$250 billion, respectively, due to economic espionage (Burgess & Power, 2008). The MePKC secures the computer communications of an electronic file over an insecure channel and an encrypted local file in the computer. However, only the ciphertext is protected and not the decrypted plaintext. Sometimes, the acquisition of many ciphertexts by an attacker through the pervasive hacking can be in favour of the attacker for ciphertext-only attack, which is especially effective for the cracking of digital signature. The current technique to resist hacking by using firewall is effective at the server side due to experienced network administrator, but not at the client side due to the technical difficulties and financial cost. Here, a simple method and device using DIP (Dual In-Line Package) switch is proposed to secure a hacking-free data storage for common end users.

MePKC can be applied to digital timestamping scheme. Humans are needed to hold the private key in the form of fully memorizable secret and not the computing devices. The legal case of laser inventor Gordon Gould (Taylor, 2002), which lasts for 30 years, has informed us the weaknesses of IP proving method using SD (Statutory Declaration). Here, MePKC timestamping scheme can be used as the evidence for general IP (Intellectual Property) like patent and copyright. This is especially effective for the first-to-invent patent filing system. Today, all the countries in the world are using first-to-file patent filing system, except the US is still using the first-to-invent patent filing system. To resist the hacking threats, a limited first-to-invent patent filing system is proposed. It has a window filing period of one year from the claim of a timestamp, and uses the timestamp to prove the IP originator(s). Ruth Taplin (2004) reported that IP assets in 2004 accounted for up to 70% and 40% of market values of all corporate assets in the US and Japan, respectively. This reflects the truth that the developed countries are now relies heavily on the knowledge economy, where IP is the main pillar. It also tells the crucial importance to apply MePKC timestamping scheme.

As a whole, the current and other applications of big and yet memorizable secret are as follows:

(i) methods and systems to realize memorizable symmetric key the secret till resistance to quantum computer attack;

(ii) methods and systems to realize memorizable public-key cryptography (MePKC);

(iii) methods and systems to improve security strength of other cryptographic, information-hiding, and non-cryptographic applications of secret beyond 128 bits;

(iv) method and system to harden the identification of embedded data in steganography although stego-data has been detected;

(v) method and system to transfer fund electronically over a remote network using MePKC;

(vi) method and system to license software electronically over a remote network using MePKC;

(vii) methods and systems to authenticate human-computer and human-human communications at a local station or over a remote network using MePKC;

(viii) method and system to use digital certificate with more than one asymmetric key pair for different protection periods and password throttling;

(ix) method and system to use three-tier MePKC digital certificates for ladder authentication;

(x) method and system to store, manage, and download voice and video calls of mobile phone and wired phone at online distributed servers;

(xi) method and system of multipartite electronic commerce transactions; as well as

(xii) method and system to boost up the trust level of MePKC digital certificate by using more than one certification authority (CA) and/or introducer of trust of web.

In a nutshell, some new key management techniques are invented. These techniques are used to create human-memorizable big secret(s) and further to realize the MePKC and its applications. The secure hacking-free DIP switch and the proposed limited first-to-invent patent filing system are hoped to be adopted prevailingly and pervasively to resist the economic espionage through virtual hacking. In the long run, the MePKC is expected to be fully utilized to create and promote an environment-friendly and electronically networked info-computer era as for the e-commerce communication to be upgraded from bipartite into multipartite cryptosystem.

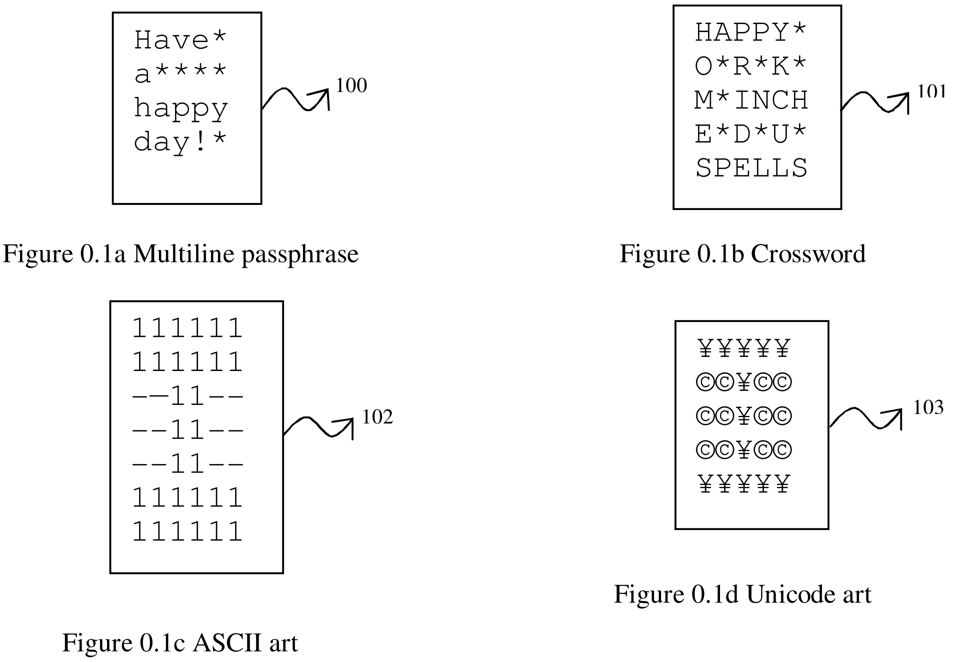


Figure 0.1a Multiline passphrase

Figure 0.1b Crossword

Figure 0.1c ASCII art

Figure 0.1d Unicode art

Figure 0.1 Two-dimensional (2D) key

Keywords: Key/password security, big secret(s) creation methods, 2D key, memorizable public-key cryptography (MePKC), multihash key, multihash signature.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

This book of research essay in the form of thesis towards a future possible doctorate thesis is the author's works prepared outside the working time of undergraduate teaching. It is published online here mainly for advertisement purposes to save development costs, public peer reviews due to the lacking of evaluation experts in the author's social networks, and to speed up the research process of human civilization for the betterment of multicultural societies.

In addition, in parallel with the purpose to gain public peer review, this book will be an output documentation of a research project registered on 27 May 2004 to achieve three aims at one stroke. These three aims are to solve an imperative research problem, to develop intellectual properties (IPs) to support an entrepreneurship, and to qualify a person for a doctoral degree. The proposal defence seminar, first work completion seminar, second work completion seminar, notice of thesis submission request, and MMU (Multimedia University) approval of this thesis title to enable its experts' evaluation are on 14 March 2005, 18 February 2008, 2 July 2008, 23 July 2008, and 1 December 2008, respectively. Nevertheless, this thesis-like book is mainly prepared in October 2008.

Before reading further the contents of this book (aka thesis), please be reminded that the process to produce the research results of this thesis is full of human-made obstacles like graft, office politics, supervisor appointment, placement of doctoral candidature under closer monitoring on the brink of termination, oral requests to give up the teaching assistantship in the form of sound snatching to stop the doctoral candidature, listings of authorship and inventorship, procrastination of work completion seminar, extraordinary request of another work completion seminar even though sufficient novel research results have been presented, etc.

Hence, to those whom have significantly and purposefully caused the damages and delays in the production of any author's research outputs, like a cash value of MYR$200-00 and procrastination of 1 hour, please confess to the God(s) and surrender yourself to the judge(s) for deserved punishment before proceeding with the thesis contents and any applications for any purposes. Otherwise, for the breach, please damn yourself as well by God(s) for the deserved punishment together with your offspring supporting your deeds. And yet for those who are helpful for their kind-hearted deeds towards the success of this thesis book, may the God(s) blesses them and their offspring supporting their deeds.

Due to the relatively high research costs invested by the author, for refund, as well as for building up a fund for further maintenance, research and development, any original and novel idea conceptions from the author in this book is only free of usage for public interests, press report, private study, research, and teaching throughout the World, with the condition that proper originality citation for source references has been clearly shown. Yet for any commercial usage, prior consent has to be obtained from the author or his successor(s).

After the author's decision for not furthering his doctoral research studies under the Lee Foundation Scholarship as communicated by Professor Michael T.-C. Fang, this research project began with its idea conception in the end of 2003 by having the official PhD project application date on 12 November 2003. It began with the studies of multimedia communications security in general and autosophy communications in particular. Then, in October 2005, some novel ideas were conceptualized on how to protect the data crystal of autosophy communications in particular, which is then generalized for any common computer data protection, to networked information security, and any applications of big secret beyond 128 bits in information engineering.

Let's create and maintain a networked info-computer age for a more paperless, petroleum-less, and environment-friendly human society by having safer multipartite electronic computer communications as from the original and novel knowledge contribution of this research project.

CHAPTER 1 OVERVIEW

1.1 Introduction

The world human population in year 2008 has achieved beyond 6.7 billion. At the same time, the world climate, resources, and environment are having red alarms on. Information communications technologies (ICT), especially the electronic communications of Internet, are believed to be tools to reduce the paper usages and transportation demands, as well as to cultivate a global economy with smoother demands and supplies. Security, health, food and beverages, accommodation, family, career, education, finance, sex, entertainment, sport, etc. are human major concerned topics. Their importance is in descending order for a majority of people.

Here, when ICT is applied to preserve more Earth resources and to conserve friendly environment, information security is always a major people concern for important computer communications. As for the Internet, identity theft is a serious crime in the electronic commerce and electronic government. Yet another serious offence is copyright piracy of literary works, software, music, image, and video. Due to the hacked computer databases of human records, the rights of privacy and publicity are also hard to be controlled and guarded.

1.2 Motivation

In term of information security, it mainly consists of cryptology, information hiding, and random number generator (RNG). Cryptology further consists of cryptography and cryptanalysis. Information hiding further consists of steganography and digital watermarking. RNG further consists of hardware RNG and software pseudo-random number generator (PRNG).

To access and control a user identity of an information security system, there are four types of authentication factors: What you know like secret, what you have like token, what you are like biometrics (Menezes, Oorschot, & Vanstone, 1996; Boatwright & Luo, 2007), and whom you know like introducer (Brainard, Juels, Rivest, Szydlo, & Yung, 2006), in the ascending order of implementation costs.

These factors can be used individually or mixed. Among them, password the secret is the most prevailing one for applying the symmetric key cryptography in the Internet due to the low implementation costs, as well as good hardware and software compatibilities. However, a secret, especially a long one, is subject to the forgetfulness or the exposure of a secret written down. The situation becomes worse when there are lots of accounts to be handled. If a secret is used for multiple accounts, there exists domino effect of password reuse problem (Ives, Walsh, & Schneider, 2004). Moreover, the memorizability size of a secret using the current prior art is limited to 128 bits for a protection period of 30 years. To solve these problems, token or biometrics with optionally bi-factor using a secret is used.

Nevertheless, token has the weaknesses of poor hardware and software compatibilities, low portability when number of tokens per user is many, high implementation costs (installation and maintenance), easy loss, possible dropping damages, and token cracking (de Koning Gans, Hoepman & Garcia 2008; de Winter, 2008; Garcia, de Koning Gans, Muijrers, van Rossum, Verdult, Schreur & Jacobs, 2008; Wikipedia Contributors, 2008az).

Meanwhile biometrics has the disadvantages of poor hardware and software compatibilities, domino effect due to limited biometrics to support multiple accounts, no perfect accurate system due to FAR (False Acceptance Rate) and FRR (False Rejection Rate), low usability and efficiency due to no universal accessibility and no permanent availability from physiological and medical factors (Maghiros, Punie, Delaitre, Lignos, Rodríguez, Ulbrich, Cabrera, Clements, Beslay, & van Bavel, 2005), high implementation costs (installation and maintenance), as well as irreplaceability and irreusability problems of biometrics upon hacking and stealth.

For examples of no universal accessibility of biometrics authentication systems, there is no support for homozygotic twins, 5% of human are not fingerprint recognition (Haylock, No date; Maltoni, Maio, Jain & Prabhakar, 2003; Vacca, 2007, p. 280) supported due to diseases like eczema ("Singaporean Female," 2008) and arthritis, human undergone surgery changing the facial structure needs re-enrolment for face recognition, 1.8 aniridia patients out of 100,000 births and patients after laser iridotomy to correct angle-closure caused by glaucoma have no iris and are not iris recognition supported, eyes alignment problem with camera of blind people and patients of pronounced nystagmus (tremor of the eyes) are poorly iris recognition supported, wheelchair users have usability problems of camera location and insufficient height variation, cataract patients after operation may need re-enrolment, and today DNA methods fail to differentiate monozygotic twins. For example of no permanent availability, the high biometrics deformation rates of very young and very old require frequent re-enrolment.

For the fourth authentication factor of “whom you know” like introducer and referee, even though the authentication burden of the introducee can be relieved, the burden has in fact been transferred to the introducer and it is up to the introducer to use the authentication factor of what you know, what you have, and/or what you are. Furthermore, there exist trust, responsibility, and obligation problems between the introducer and introducee. The human interaction models (Kurokawa, 1988, 1990, 1991, 1997) are then required to analyze the security probability of this factor.

In view of the limitations of token and biometrics, how good if the weaknesses of secret like memorizability and entropy size can be improved until the token and biometrics are not needed for majority applications.

1.3 Research Aims

Here, the first main focus of this research project is for this direction: Methods and systems to create big and yet memorizable secret(s). From a sufficiently large and yet memorizable master key, it shall be possible to derive multiple unique slave keys for multiple offline and online accounts. These slave keys shall be impossible to be used to derive other slave keys.

For public-key cryptography (PKC), the smallest practically secure private key size is 160 bits by using the FFC (Finite Field Cryptography) or ECC (Elliptic Curve Cryptography) (Gehrmann & Näslund, 2005, 2006, 2007; E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b). Using the current prior arts like encrypted private key, split private key (Ganesan, 1996b), and roaming private key (Baltzley, 2000), there has been no fully memorizable private key yet. Here, the second main focus of this research project is to develop fully memorizable private key towards MePKC (Memorizable Public-Key Cryptography), aka MoPKC (Mobile Public-Key Cryptography). The third research focus is to apply the outputs from the first and second foci of this research project for various applications in the field of information engineering.

1.4 Research Methodology

This research project originally contributes novel methods and systems to create big and yet memorizable secret(s) and then MePKC for various applications in information engineering. As from Spafford (1993), there are three types of techniques to prove a model in a computing dissertation: Analytic method using formal manipulations, stochastic method using statistical measurements, and building a prototype for experimental testing.

For the research methodology of this research project, the third method of building a prototype is used to show that it is possible to create big and yet memorizable secret by using 2-dimensional (2D) key (Lee, 2006b, 2008i), and further for the practical realization of MePKC (Lee & Tan, 2006b; Lee, 2008j). Some prototypes of the other big secret creation methods have also been built (Lee & Tan 2006a; Lee, 2008k).

To show the security strength and protection period of various security schemes in this research project, the first method of analytically formal manipulations has been used. Since the author cum researcher is an electrical engineer and not educated as a mathematician, the reduction-based security (aka provable security) approach is only tried on his best effort as time allows. Hence, those cryptographers from the mathematics field are expected to carry out some provable security studies on the security schemes proposed here whenever the big secret creation method(s) is applied, especially for the MePKC and its applications.

1.5 Organisation of the Thesis

Generally, the present invention of this thesis generally relates to computer communications security. More particularly, the present invention relates to key management of cryptography and information security. Most particularly, the present invention relates to methods and systems to create big and yet memorizable secrets that are large enough for the higher levels of security strength of security systems like AES-256, 256-bit ECC, 256-bit PRNG, and so on, (where AES stands for Advanced Encryption Standard; ECC stands for Elliptic Curve Cryptography; and PRNG stands for Pseudo-Random Number Generator), together with their derived applications in the general field of information engineering and specific field of information security like memorizable public-key cryptography (MePKC).

Particularly, the present invention broadly provides novel generation methods and systems of big memorizable secrets to practically realize stronger security levels of cryptographic, information-hiding, and non-cryptographic applications in the information engineering, especially MePKC (Memorizable Public-Key Cryptography). The first independent embodiment of the present invention is the methods and systems to create big and yet memorizable secrets. The second independent invention embodiment is mutlihash key using hash iteration and hash truncation to create multiple slave keys from a single master key. And yet the third independent embodiment of the invention is multihash signature that allows object-designated message with specific meaning, function, or recipient. From these three independent inventions, there are then various types of dependent inventions for various practical applications mainly due to the existence of big memorizable secrets.

The organisation of this thesis has three components: Preliminary section, chapter section, and postscript section. The preliminary section consists of front page, copyright page, declaration, acknowledgements, dedication, abstract, table of contents, list of tables, list of figures, and prefaces. For the postscript section, it has an appendix to show the writing systems of the world, references, and acronyms.

There are 15 chapters in the chapter section. Chapter 1 is an overview of this research project. Chapter 2 discusses the life of research postgraduate students in brief, general conditions to get a doctoral degree, and the practical evaluation on knowledge contribution to qualify a person as a quality researcher.

Chapters 3, 4, and 5 form closely related contexts to explain methods and systems to create big and yet memorizable secret. These methods include Chinese language passphrase (CLPP), 2-dimensional (2D) key, multilingual key, multi-tier geo-image key, multi-factor multimedia key using software token, and their hybrid combinations. Chapter 6 enhances the secret randomness by using multimedia noises.

Chapter 7 presents a method and system called multihash key to generate multiple unique slave keys (aka site keys) from a master key for both offline and online accounts. This multihash key uses hash iteration and hash truncation. Every slave key is computationally infeasible to be used to derive another slave key. Chapter 8 on multihash signature uses the concept of hash iteration again to generate multiple unique signature of a message file from only a single pair of asymmetric key. Both multihash key and multihash signature are integrated into various MePKC applications as explained in the latter chapters.

Chapters 9 to 13 are all on the applications of big secret and MePKC. Chapter 9 includes the applications for symmetric key cryptography, MePKC, as well as other cryptographic, information-hiding, and non-cryptographic applications. Chapter 10 includes the identification hardening of embedded data in steganography, electronic fund transfer, and electronic software licensing. Chapter 11 includes the local/remote human-computer and human-human authentication without shared secret, MePKC digital certificate having more than one asymmetric key pair for different protection periods and password throttling, as well as three-tier MePKC digital certificate for ladder authentication. Chapter 12 includes archiving the voice/video calls of wired/wireless phones, multipartite electronic commerce transactions, as well as trust boosting of MePKC digital certificate by using more than one certification authority and/or introducer of trust of web. Chapter 13 talks on MePKC timestamping scheme for evidence of intellectual property (IP) originality.

Chapter 14 is a hardware contribution to further secure the computer communications of MePKC from virtual hacking over a connected computer communications network like Internet. It is about a hack-proof data storage using innovated DIP (Dual In-Line Package, aka DIL) switch. Chapter 15 concludes this thesis by giving a concise summary on the originally contributed novel knowledge and some suggestions for future research.

CHAPTER 2 EVALUATION ON KNOWLEDGE CONTRIBUTION

2.1 Life of Research Postgraduate Students

Due to the abundance of knowledge to be explored since the availability of computer assistance for the limits outside of the human capabilities, the research postgraduate students at the universities studying for master's and doctoral degrees blossom not only in the developed countries but the developing countries like Malaysia as well. However, many postgraduate students in the developing countries encounter the suitability problems of closely related research supervision and fresh supervisors who are lacking of the supervision experience. Hence, it is good for a postgraduate student to browse some informative materials discussing about the life of research postgraduate students to smoothen the postgraduate studies.

Callahan (2001) has a brief article telling the story of Ph.D. study in the USA. Pratt (1997) has a graduate school survival guide for us. desJardins (1994, 1995) published articles in ACM Crossroads on how to be a good graduate students. Lauer (No date) talked on how to prepare a Ph.D. thesis proposal in computing science.

Some supervisors require postgraduate students to publish copyrighted journals and conference proceedings papers as a prerequisite for thesis submission. Hence, it is good to know how to prepare a first draft of paper (San Francisco Edit, No date). Smith (1990) told the task of the referee who would review the manuscript. An IEEE fellow, who is B. K. Bose (2006), disclosed the tips to publish a transactions article. Spafford (1993) briefed on Ph.D. dissertation (Wikipedia Contributors, 2008t). Comer (1993) edited an article on how to write a Ph.D. dissertation.

2.2 General Conditions to Get a Doctoral Degree

For lengthy and more informative materials on the life of research postgraduate students, there are some books on this topic (Noble, 1994; Elphinstone & Schweitzer, 1998; Cryer, 2000; Rugg & Petre, 2004; Phillips & Pugh, 2005).

The first doctoral degree was conferred in Paris circa 1150 (Noble, 1994). Since then, the most basic condition to get a doctoral degree is to have contributed sufficiently original and novel knowledge to the society. For honorary doctorate, the analogous condition is sufficiently significant contribution to the society.

In some universities, the thesis publication is an optional component. Yet in some universities now, a postgraduate student has to follow some preliminary courses like entrepreneurship and research methodology. Other possible conditions are proposal defence of research proposal and seminar defence of thesis. Depending on research supervisors and universities, publishing journals and conference proceedings papers are optional conditions to qualify for thesis submission.

2.3 Prior Approaches of Research Evaluation

Just like the quality control and evaluation of products and services, there are various approaches to evaluate the research output. For doctoral dissertation awards and novel idea competitions of the universities and professional organizations like ACM (Association for Computing Machinery), Google, and IEEE (Institute of Electrical and Electronics Engineers, Inc.), good evaluation techniques of research results and novel ideas are required to determine who the award winners shall be.

The evaluation of a researcher's contribution is traditionally majority-wise based on the publication counting, credit of journals and proceedings, citation counting, H-index (Hirsch, 2005; Batista, Campiteli & Kinouchi, 2006; Wikipedia Contributors, 2008bg) qualified peer evaluator, and contribution impact. The last element, which is contribution impact, is in fact the main key measure to evaluate the contribution of a researcher to the welfare of human civilization. However, this element is not easy to be carried out by the management without the advice of the expert. Hence, in deciding the recruitment, promotion, compensation, funding allocation, reviewer list, and consultancy partnership, the publication counting method is always used due to its easiness.

Lately, Parnas (2007) published his view on the publication counting method and called for the halt of number game. He referred a paper by Ren and Taylor (2007), which calls for automatic and versatile publication ranking, and pointed out the weaknesses of publication counting method. These weaknesses are encouraging superficial research, overly large groups, repetition, small and insignificant studies, as well as rewarding publication of half-baked ideas. Parnas also acknowledged that publication counting method could corrupt the researchers because malicious researchers would have the behaviour of publishing pacts, clique building, anything goes, bespoke research, and minimum publishable increment (MPI). As a solution, he called for an accurate researcher evaluation, where the researcher's papers were studied by qualified peer evaluators. The factors of slower time and evaluators' compensation are the disadvantages of this method.

2.4 Contribution Impact

Here, a refinement of researcher evaluation from the Parnas' proposal is proposed. Heavier weight shall be given to the contribution impact, which is to be studied by the qualified peer evaluators. The evaluation methods based on publication counting, credit of journals and proceedings, and citation counting are inaccurate. For example, the same research result can be published two times as both the patent and journal (Rivest, Shamir & Adleman, 1978, 1983). A second case is that a same idea is published three times as proceedings paper, patent and journal (MacKenzie & Reiter, 2001c, 2002, 2004). The example of the second case has another closely related conference proceedings paper (MacKenzie, Oprea & Reiter, 2003). The third case is that an idea is published four times as technical report, proceedings paper, patent, and journal (MacKenzie & Reiter, 2001a, 2001b, 2003, 2006). One more possible case is that an idea may be published as many as eight times in the forms of technical report, conference proceedings paper, patent, journal, thesis, book chapter, and book.

The proposal here is mainly aimed at evaluating the research results of the scientists and engineers. The research types are fundamental, applied, and design-end engineering. Meanwhile, the research results can be discovery, innovation or invention documented as one or more types of IPs (Intellectual Properties). It can be copyright like technical report, proceedings paper, magazine, letter, journal, book, thesis, computer program, architectural plan, etc. Conference proceedings papers include articles in the conference, symposium, workshop, and colloquium. Other forms of IP are patent, utility model, industrial design (aka design patent), layout-design of integrated circuit, trade mark, and confidential information. Two more intangible assets are publicity right and privacy right.

Proceedings paper, magazine, letter, journal, and thesis are normally reviewed lightly within a short time by one to three peer evaluators. On the other hand, patent, utility model, and industrial design are evaluated intensively by patent examiner within a long period of time ranging from 1 to 6 years or more. To fulfil the patentability, there are four conditions: Novelty, utility (aka industrial applicability), non-obviousness, and specification (i.e. within the patent classification of a country). Some research results like evaluation, survey, analysis, case study, and commentaries are not patentable. Commentaries include on-site experience, viewpoint, technical opinion, and forum.

Besides, the guidelines for inventorship and authorship (Devenport, 2005; Albert Einstein College of Medicine of Yeshiva University, 2008; Wikipedia Contributors, 2008ao) shall be enforced strictly. The inventorship is enacted where only people with idea conception can be listed and not the people with reduction to practice, which when listed can acquit the accused infringers from infringement charges (Setty & Gentry, 2002). For authorship, the law is not yet enacted and there exist only ethical rules. Hence, there are authors due to idea conception, analytic evaluation, reduction to practice, novel supervisory advice, nominal supervision, fund raising, etc., where some of them are honorary authorship not accepted by some editorial offices (IEEE Publications, 2002).

For contribution impact, the successful implementation and realization of a research result for public usage and application are very important. Many research results are only documented without a further development. This is in fact no positive impact at all but negative impact due to a waste of resources.

Hence, besides Parnas' proposal to request a few peer reviewers to evaluate a researcher's selected papers, the evaluated researcher has to list out the current impact of one's research results. For paper with multiple authors or inventors, the specific researcher has to point out one's contribution in the listed paper for impact evaluation. For example, article like this (Re, Borean, Bozzi, et al., 2002) requires detailed elaboration on the particular contribution of every single listed inventor or author. This is important to avoid the pseudo-amplification phenomenon where a co-written article is considered as whole unit in the invention counting of an inventor or publication counting of an author.

The contribution impact can be a core, supporting element, and influence to a past, present, and future product and/or service useful to the human society. The obsolete technology, replacing technology, and alternative competing technology have also to be accompanied with the researcher's selected papers.

Modern society has evolved from agricultural and industrial economy into knowledge economy. In 2004, Taplin (2004) reported in a paper that IP assets accounted for up to 70% and 40% of market values of all corporate assets in the USA and Japan, respectively. Chandran (2007) stated that for any company, and especially for pharmaceutical companies, IP is more valuable than any of its tangible physical assets, where IP constituted more than 80% of the total revenues of any company while tangible assets accounted for only 20%. Again, Ocean Tomo LLC (Ocean Tomo, No date; Wikipedia Contributors, 2008d) figured out that the components of S&P 500 market value in 2005 also had 79.7% to be intangible assets. Among the IP, patent has the largest economic value.

From Abril and Plant (2007), only 5% of the US patents were licensed and only 3% generated royalties. Research results with commercial values are normally filed as patent today for refunding research costs, usage control, enabling mass production, affordable cost and price control, reasonable income, technology control, official government records, archival for future generations to further study, etc.

Under the new rules of USPTO (Dudas, 2007), a patent can only carry a maximum of 5 independent claims and maximum 20 dependent claims in total. For every independent claim, if fully developed, explained, and evaluated, it can amount to a proceedings paper and/or journal. Furthermore, inventorship is stricter than authorship, only the researcher with novel conception can be listed and not the researcher carrying out the reduction to practice. Hence, the inventorship can better determine the main original knowledge contributor.

Therefore, the Parnas' suggestion has to include the request for the evaluated researcher to list out the impacts of one's research papers to the human civilization. Listed impacts by the researcher can speed up the evaluation process. This additional step can also encourage more research results to be fully developed for public usage, instead of just discussing a war on some plain papers.

2.5 Research Quality

When the researcher has listed out the implemented products and services as the impacts of one's research papers, the research quality of these outputs started as an idea is then to be evaluated. The idea evaluation criteria based on Google Project 10^{100} (2008) are reach, depth, attainability, efficiency, and longevity. These five criteria correspond with five questions "How many people will this idea affect?", "How deeply are people impacted? How urgent is the need?", "Can this idea be implemented within a year or two?", "How simple and cost-effective is your idea?", and "How long will the idea's impact last?", respectively.

Yet there are two persons telling how important a good idea is. Pierer studied law and economics, and he is the president and CEO of Siemens AG. Oetinger studied political science in Berlin and business administration at Stanford Graduate School of Business, and he works for Boston Consulting Group, where he is a senior vice president and director of The Strategy Institute. In their book entitled "A Passion for Ideas: How Innovators Create the New and Shape Our World" (Pierer & Oetinger, 2002), they wrote in the book cover that "The creation, implementation, and sustainability of new ideas is the lifeblood ensuring the growth and viability of any organization. Without continuing innovation, competitive advantage and global market share are endangered. Once-thriving organizations can find themselves unprepared for the future." This tells how important new ideas are to any organization including company, university and government for their thriving sustainability and continuous success. In a smaller scale, the successful completion of a research degree like master's and doctoral degrees also depends on the sufficient novel ideas contributed to the pool of human knowledge for further betterment of our civilization.

In the more developed countries, a good idea resulted from the research is normally filed for patent before any public disclosure or within one year from the first date of the public disclosure. Good examples can be observed from the fields of fibre optic communications (Hayes, 2001, pp. 1-12), laser (Taylor, 2002), and cryptography (Rivest, Shamir & Adleman, 1983). For less developed countries, the IPs like patent and copyright are not so strongly focused and emphasized as in more developed countries. The reasons are explained by Scalise (1999) that less developed countries respond to their comparative advantage in imitation by lowering IPR protection to reduce the cost of imitation's inputs, and more developed countries respond to their comparative advantage in innovation by raising IPR protection to increase the value of innovation's output.

Hence, to catch up with the more developed countries, the less developed countries have to culminate in their research to produce the most dominant form of IP, i.e. patent. Turk (2005) expressed that the most important IP legal issues of the twenty-first century is to standardize global IP rights to facilitate a better mass production, lower operating costs, and smoother supply-demand chains.

The patent grant rate in the US is about 75% (Ebert, 2004, 2005, 2007). The high patent grant rate does not mean there are lots of useful and yet competitive inventions. In fact, Professor Adam Jaffe in his testimony before the Judiciary Subcommittee on the Courts, the Internet, and Intellectual Property claimed that lots of US patents were worthless and unimportant ("Prepared Testimony," 2007). This fact is also true for copyrighted publications of journals and conference proceedings.

Then, this comes to the question where the great inventors have gone (Bessen, 2004). However, fame is just one of the main factors people filing patents. Another more important factor is the economic values of a patent to perform like a property (Meurer & Bessen, 2008). This has caused an abundance of patent applications in the US and examination becomes a problem due to the lacking of patent examiners ("Prepared Testimony," 2007). The abuse of US patent system is called to a halt (Lemley & Moore, 2004; Lemley & Sampat, 2007). To solve the examination problem, Beth Noveck created a voluntary system called "Peer to Patent Project" for community patent review to assist the USPTO patent examiners (Duane, 2008; Oram, 2008; “Peer to Patent,” No date). Later, the JPO (Japan Patent Office) has also launched its version of community patent review system called Komyunitipatentorebyu [コミュニティパテントレビュー]. Lately, the US Chamber of Commerce (USCOC) has recommended the Peer to Patent Project for further implementation (“Recommendations for consideration,” No date). In analogy, the journals and conference proceedings also have a similar review system called Computing Review under the ACM (“Computing Reviews,” No date).

The inventions in this research project in the IP form are mainly software patents or CIIs (Computer-Implemented Inventions). In the US, there has been a time where people questioning on whether software patents shall exist (Samuelson, 1990). Now, the question has changed to how to improve software patent quality (Thatcher & Pingry, 2007). Here, the inventions of this research project have been filed for patents (Lee, 2008h) and now are pending for patent examinations. The implementation for public usages has been partially done. However, the research quality is yet to be investigated.

To evaluate the research quality, a currently common method is citation counting. This is true for journals, conference proceedings as well as patents. There are also citation analyses of journals, conference proceedings, and standards based on the citing patents. Example is IEEE papers based on US patents (Thomas & Breitzman, 2006a; Platt, 2006a, 2006b; Breitzman, 2007, 2008) and EU patents (Breitzman, 2006). From these IEEE papers, Canning (2006) suggested the libraries to optimize the financial budgets for scholarly collections. IEEE also has occasional published articles to analyze the US patents (Goldstein, 2006; Sweet, 2007).

Besides, there are some organizations specialized in the research evaluation (“1790 Analytics,” No date; “Research Evaluation,” No date). The 1790 Analytics discovered that hot patents were normally linked with government-funded scientific research (Thomas & Breitzman, 2006b). Hot patents are patents whose impact on recent technology developments is particularly strong. These hot patents shall be identified to know the most competing patents for commercial profits. The damage award (Aharonian, No date) of a patent infringement has a record of USD$1.52 billion for Alcatel-Lucent v. Microsoft in the US (Wikipedia Contributors, 2008an).

Spours (2006) discussed on how to exploit patents for profit. Since the academicians' research funding at the universities is normally financed by the taxpayers' money, it shall not be wasted without any refund or profit. To encourage the commercialization of university research, the US Bayh-Dole Act (Bayh, 2006) was enacted in 1980 to allow the researchers and universities to be the IP assignees. Barham and Foltz (No date) investigated the patent activities and commercialization efforts of university life science researchers.

Bray and Lee (2000) analyzed the university revenues from technology transfer and concluded that the average equity from a university spin-off company was more than 10 times the average annual income from a traditional license. Another available data of university licensing revenue is from School of Medicine, Johns Hopkins University, Baltimore, Maryland, USA ("Licensing Revenue," No date). From this encouraging university patent revenue, it also indirectly tells that the research output is practical, useful, and cost-effective for the industries to apply. Branstetter and Ogura (2005) had an article to show the example of knowledge spillover from the bio-science academia to the industry.

For the specific patent quality, Malackowski and Barney (2008) had a general discussion on this topic. On the other hand, Ocean Tomo (No, date; Wikipedia Contributors, 2008d) has an Ocean Tomo 300 Patent Index to show that patent quality can be measured.

There are also some patents on how to analyze and select good patents due to their practical usability and competitiveness (Breitzman & Narin, 2001; J. A. Barney & J. R. Barney, 2003; Williams, 2004). Patent exists on how to track and audit IP of open source software (Donner, 1999, 2001; Bonnet, Baroniunas & Webbink, 2008).

There is another patent telling a method to obtain investment income based on the capitalization of intellectual property (Elliott, 2007). Yet, there are patents on how to license intellectual property assets (Shelton, 2007), how to do intangible property transaction and leaseback business method (Walker, 2007), and protection against the changes of intellectual property value (Risen & Covello, 2000).

All of these hint that optimum supply-demand chains of intangible assets are forming gradually from problem, research, solution, investment, patenting, capitalization, licensing, and commercialization to the customers. It also tells that the method to evaluate and apply the novel idea, especially patent, is getting better in terms of relevancy, accuracy, systematization, and valuation. The inventions in this research project are expected to undergo these processes in the coming decades.

As of today, the best research evaluation experts are from 1790 Analytics LLC (No date), USA, doing the patent analysis and intellectual property evaluation. Its services are recruited by the IEEE, USA, which is the top organization in the field of electrical and electronic engineering, as well as financial experts for equity investment and other top ranking organizations.

Lastly, a reminder here is to note and emphasize that under the IP laws, only those persons who have created sufficiently novel knowledge contribution are qualified to be coined as the originating IP contributors like inventor, designer, author, etc. This point is important to get rid of the malicious supervisors whom have no significantly novel knowledge contribution but wanted to be listed as co-contributors to seek for better career profile towards more bonus and salary increment, faster and easier promotion, as well as higher social recognition for awards and reputation.

CHAPTER 3 CREATING BIG MEMORIZABLE SECRET (PART 1)

3.1 Required Protection Periods and Their Key Sizes

According to Kerckhoffs' law (Schneier, 1996), a cryptosystem shall depend 100% on the secrecy of password or key only. In the words of Shannon's maxim, it means "enemy knows the system". This law makes the civilian cryptosystem to have publicly known algorithm except the classified governmental and military information. This is needed to gain the public confidence for general daily applications from the fear of possible backdoor. There are various applications of secret in information engineering. Here, the required protection periods and their key sizes are briefly discussed to know how big a memorizable secret shall be.

If a cryptographic algorithm is securely tested, the required key length in character (L_C) of a password will depend on the factors of number of characters (C), key space (S), secure period (T), guesses per unit of time (G), and probability of guessing (P) (US Department of Defense, 1985). The minimum key length has to be able to resist the brute force attack. The relationships of L_C, C, S, T, G and P are given in Equations (3.1) and (3.2).

$$S = \frac{GT}{P} \tag{3.1}$$

$$L_C \geq \left\lceil \frac{\log_2 S}{\log_2 C} \right\rceil \tag{3.2}$$

Nowadays, character encoding of ASCII is the most popular computing code. ASCII has some key sets of 26 lowercase characters, 26 uppercase characters, 10 digits, 62 alphanumeric characters, 33 non-alphanumeric characters, 95 printable characters, etc. If a key only has symbols of digits, its specialized name is *passcode*. If a key is long or consists of printable characters, it is named as *passphrase*.

There were once three Data Encryption Standard (DES) challenges as in year 1997, 1998, and 1999. Using the distributed network computing, maximum guesses of 2.45x10^9 keys per second was once recorded. For the latest guesses per computer as at end of year 2005, it is about 1.5x10^7 keys per second. The increment rate follows the Moore's Law (Wikipedia Contributors, 2008v) where computer performance is doubled for every 18 months. This indicates that strong password has to be longer as time passing by if there is no special key processing added.

Key length in bit (L) means that there are 2^n possible keys for n-bit key. By year 2010, the required key is 80 bits for symmetric key algorithm as announced by U.S. National Institute for Standards and Technology (NIST). Meanwhile, asymmetric key algorithm, like RSA, needs 1024 bits to be equivalently strong with 80-bit symmetric key algorithm as claimed by RSA Security. The key space varies and depends on the security requirements.

For the AES suggested by NIST to replace the DES, it has three types of symmetric key sizes. These key sizes are 128, 192, and 256 bits. Therefore, we have AES-128, AES-192, and AES-256 to fulfill the demands of security levels at 128, 192, and 256 bits (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b). For other security levels at 80 and 112 bits, NIST suggested two-key Triple Data Encryption Algorithm (2TDEA) and three-key Triple Data Encryption Algorithm (3TDEA), respectively (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b).

Table 3.1 Minimum symmetric key sizes for different security levels of protection

Key Size, bit		Protection
# 1	32	Individual attacks in "real-time". Only acceptable for authentication tag size.
# 2	64	Very short term protection. Obsolete for confidentiality in new systems.
# 3	72	Short to medium term of protection depending on organization size.
# 4	80	Smallest general purpose level, < 5 years protection.
# 5	112	Medium term protection. About 20 years.
# 6	128	Long term protection. Good, generic application independent recommendation, about 30 years.
# 7	256	Foreseeable future. Good protection against quantum computers.

Password choice depends on the strength and memorizability. Strength depends on key size in bit. Memorizability depends on number of memorized secrets in a human brain. For minimum key sizes at different security levels, it is shown in Table 3.1 (Gehrmann & Näslund, 2005, 2006, 2007).

For short term memory of English-based digit, Miller (1956) showed an average of 7 items plus or minus 2 (7 ± 2) (Jones, 2002). The good option is longer key size in bit and still memorizable. Some articles on memory can be referred (Baddeley, Thomson & Buchanan, 1975; Ellis & Hennelly, 1980; Hoosain & Salili, 1988; Cowan, 2001; Wikipedia Contributors, 2008l, 2008as, 2008aw) and we can see that a user has 6.5 unique passwords in average (Florencio & Herley, 2007), or 4 to 5 unrelated keys (Adams & Sasse, 1999). These are textual secret; whereas graphical secret has higher memorizability (Standing, Conezio & Haber, 1970; Standing, 1973).

On the other hand, there are 3 conventional mathematical hard problems used in asymmetric key cryptosystem, which is also called public-key cryptosystem. These problems are integer factorization problem, discrete logarithm problem, and elliptic curve discrete logarithm problem. NIST categorizes the applications of these problems for public-key cryptography as integer factorization cryptography (IFC), finite field cryptography (FFC), and elliptic curve cryptography (ECC), respectively.

IFC has a long key size for public and private keys. FFC has a long public key and a short private key. ECC has a short key size for public key and private key. The minimum asymmetric key sizes for IFC, FFC, and ECC in equivalent with the security levels of symmetric key sizes are shown in Table 3.2 (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b).

Table 3.2 Minimum asymmetric key sizes in equivalent with the security levels of symmetric key sizes

Security (bits)	IFC		FFC		ECC	
	Public	Private	Public	Private	Public	Private
80	1024	1024	1024	160	160	160
112	2048	2048	2048	224	224	224
128	3072	3072	3072	256	256	256
192	7680	7680	7680	384	384	384
256	15360	15360	15360	512	512	512

A good password has to be strong and memorizable (Gehringer, 2002). The random password with printable ASCII characters is the strongest password but it is poor in memorizability (Yan, Blackwell, Anderson & Grant, 2004). However, password with good memorizability tends to be weak password and under the password cracking threats of guessing and dictionary attack (Klein, 1990). As time lapses, longer key length is needed due to the advancement of computer technology. Hence, the trend is the strong and memorizable passphrase or special key processing technique like key strengthening is adopted to get rid of the quest of longer key size.

The most popular email encryption software called Pretty Good Privacy (PGP) 9.0 (PGP Corporation, 2006) allows a maximum of 255 characters to be the passphrase. Microsoft Windows operating systems also have this feature. Methods exist on how to create secure keys (Adams, Sasse & Lunt, 1997; Brown, Bracken, Zoccoli & Douglas, 2004). Therefore, it is a research problem here questioning on how to have big enough and yet memorizable secret(s) for various applications in information engineering, generally, and security engineering, particularly.

3.2 Review of the Secret for Symmetric Key Cryptosystem

In civilian information security, according to Kerckhoff's Law, a security system shall depend fully on the secrecy of a key, and not the algorithmic software nor its hardware. The main reason for this law is that public confidence has to be earned to show that there is no backdoor in the security system relying solely on the secrecy of key, and disclosing its algorithm and hardware to the public, especially academic and corporate researchers, for comments.

For authentication to a security system, it basically has four methods: Secret for what you know, token for what you have, biometrics for what you own, and person for whom you know. Due to the factors of cost, hardware and software compatibilities, password/key the secret is the most popular. Short key is called password and long key is called passphrase. The key selection is always the balance of the factors of memorizability and security. Long and random key is securer but harder to remember. The current prior art of single-line key input field limits the practical memorizable key size to a maximum of 128 bits for majority normal users.

To create longer password called passphrase, there are now four existing methods: Sentence-type passphrase, acronym-type passphrase, diceware, and coinware. Sentence-type passphrase is memorizable and has long key size, but vulnerable to dictionary attack; whereas acronym-type passphrase taking the first, last, other locations, or hybrid location is memorizable and resists to dictionary attack, but has a small key size. Diceware and coinware use several dices and coins, respectively, to randomly select a word from monolingual, bilingual, or multilingual wordlists, where they can resist dictionary attack, but memorizablity reduces as the key size becomes longer. Hence, these passphrase generation methods are still insufficient to create random, memorizable, and yet big secret, that can resist guessing attack and dictionary attack, to fulfil the need for secret bigger than 128 bits.

Table 3.3 Various key sizes corresponding to the numbers of ASCII characters, Unicode (version 5.0) characters, and password units of various secret creation methods, as well as the settings sufficiency of some key input methods and systems

Key size (bit)	80	96	112	128	160	192	256	384	512
Number of ASCII character (6.57 bits)	13	15	18	20	25	30	39	59	78
Number of Unicode character (16.59 bits)	5	6	7	8	10	12	16	24	31
Number of CLPW unit (85.41 bits)	1	2	2	2	2	3	3	5	6
ASCII-based (4 * 5) 2D key (131.4 bits)	Yes	Yes	Yes	Yes	No	No	No	No	No
ASCII-based (5 * 6) 2D key (197.1 bits)	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
ASCII-based (7 * 6) 2D key (275.9 bits)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Unicode-based (5 * 5) 2D key (414.8 bits)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
BW multilingual key (16.59 bits)	5	6	7	8	10	12	16	24	31
BW multilingual key + grid (19.79 bits)	5	5	6	7	9	10	13	20	26
Color multilingual key (24.59 bits)	4	4	5	6	7	8	11	16	21
Color multilingual key + grid (27.79 bits)	3	4	5	5	6	7	10	14	19
Multi-tier geo-image key (64.82 bits)	2	2	2	2	3	3	4	6	8
Multi-factor key using software token	Halving the memorizable key sizes of MePKC and other applications at equivalent security levels by using AES.								

N.B.: BW = Black-and-white

According to Bruce Schneier (2006, 2007), for a survey of 34,000 MySpace users' passwords, about 99% of the passwords have 12 ASCII characters or less. An

ASCII character carries about 6.57 bits, which means 99% of the 34,000 MySpace passwords have 78.84 bits or less. This reflects the facts that almost all the symmetric keys of the current symmetric key cryptosystems in practice reach at a key size less than 128 bits. In other words, memorizable key the secret is only practically applicable to the current popular symmetric key cryptosystems like 112-bit 3TDES (3-Key Triple Data Encryption Standard) and 128-bit AES (Advanced Encryption Standard). However, in a large-scale password habit survey (Florencio & Herley, 2007), the average password entropy is about 40.54 bits and a user has 6.5 passwords for 25 accounts where 8 accounts in average are used daily.

Table 3.3 shows the numbers of ASCII and Unicode (version 5.0) characters for various key sizes. In Unicode 5.0, there are 98884 graphic symbols or 16.59 bits per graphic symbol. The repertoire of Unicode graphic symbols can be upgraded from time to time in future versions to enlarge the number of graphic symbols. Memorizable keys for 192-bit and 256-bit AES are out of the reach of the current key management method and system. Hence, need exists to have better key management method and system to create larger key/password the secret larger than 128 bits.

3.2.1 Related Work: Single-Line Key/Password Field

Conventionally, when secret is used for authentication, single-line key field will be the area for a user to enter a key. For the current longest possible key, it is a single-line passphrase. Passphrase can be formed from acronym, sentence, diceware, and coinware (Lee & Ewe, 2006). Nevertheless, limit exists due to the problems of memorizability and ASCII character input from keyboard. The first problem is due to the human factor; whereas the second is due to the user interface. These problems prohibit the applications of symmetric key sizes at higher security levels, whenever a user cannot remember and/or conveniently enter a long single-line passphrase.

3.3 Review of the Secret for Asymmetric Key Cryptosystem

Besides the symmetric key cryptography, asymmetric key cryptography or public-key cryptography (PKC) is one of the two main components in the field of cryptography. PKC emerges in the 1970s (Diffie & Hellman, 1976; Goldwasser, 1997). Symmetric key cryptosystem has a shared secret key between a pair of users, but each PKC user has an asymmetric key pair consisting of a private key known only to the user and a public key shared with the other users. Amazingly, PKC can solve the key sharing and distribution problems of symmetric key cryptosystem. Moreover, PKC can resist the guessing attack, dictionary attack, and pre-computation attack that symmetric key cryptosystem is susceptible to. Nevertheless, PKC processing speed is about 1000 times slower than the symmetric key cryptography. Consequently, PKC and symmetric key cryptosystem have to be used in hybrid mode for maximum performance of effectiveness.

Now, there are three main conventional asymmetric cryptosystems: IFC (Integer Factorization Cryptography), FFC (Finite Field Cryptography), and ECC (Elliptic Curve Cryptography). IFC is based on the mathematical hard problem of integer factorization. FFC is based on discrete logarithm problem. And ECC is based on elliptic curve discrete logarithm problem.

RSA (Rivest-Shamir-Adleman) cryptosystem is a type of IFC being the very first practical realization of PKC since 1977. FFC like ElGamal encryption and DSA (Digital Signature Algorithm), as well as ECC are firstly introduced in the 1980s. Then, there are other PKC based on different mathematical hard problems but not yet well-standardized. Nevertheless, so far all the key sizes of asymmetric private key for IFC, FFC and ECC are too big to be human-memorizable. The large key sizes of RSA cryptosystem for its both private and public keys, as well as FFC cryptosystem for its public key, have even caused the USA government to shift to ECC having significantly smaller public and private key sizes. For more details on their practically secure key sizes, please refer to two NIST articles (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b).

Due to the reason that private key is not fully human-memorizable using the current prior art, a private key is either fully or partially in the form of a token. In the mean time among the prior art, there are three basic methods for private key storage: (i) Encrypted private key stored in the local computing system or device; (ii) split private key firstly proposed by Ravi Ganesan (1996b) on 18 July 1994 in the US

Patent US5557678; and (iii) roaming private key firstly proposed by Cliff A. Baltzley (2000) on 25 November 1998 in the US Patent US6154543. All the three methods are bi-factor or multi-factor authentication, where at least one factor is a secret and another factor is a software token or hardware token.

The first method of private key storage encrypts the private key using a symmetric key and stores the ciphertext of private key in the local computing system like hard disk drive or a device like smartcard, floppy disk, or USB flash drive. Encrypted private key method suffers from the problems of loss, damage, side-channel attacks, mobility, hardware and software compatibility, and password domino cracking effect of its digital certificate carrying only one asymmetric public key.

The second method splits a private key into two or more portions, where the first portion is a memorizable password or derivable from the memorizable password kept by the owner of that private key. The second and possible other portions of the private key are kept by one or more servers in the encrypted form like the first method. The first, second and possible other split portions of the private key may also be derived from various authentication factors like token and biometrics. Split private key method suffers from the problems of malicious central authority attack on the user's short password, dictionary attack on the stolen encrypted partial private key, and password domino cracking effect of its digital certificate carrying only one asymmetric public key.

For the third method, roaming private key also has encrypted private key but its ciphertext is stored in a network system like server, and owner of the private key can download it from anywhere and anytime as long as the user has network access. The roaming private key method suffers from the problems of side-channel attacks, hardware and software compatibility, malicious central authority, dictionary attack on the stolen encrypted private key, and password domino cracking effect of its digital certificate carrying only one asymmetric public key.

In the US Patent US7113594, Boneh and Franklin (2006) described a new type of PKC called identity-based cryptography (IDC). In this method, a user's unique public identity like email or phone number is the public key and hence memorizable. However, its private key is not memorizable and has to be generated by a trusted third party (TTP).

Notwithstanding, as compared with symmetric key cryptosystem using password or key the secret, the popularity of token-based PKC using fully or partially encrypted private key, is low due to the problems of mobility convenience, implementation costs, hardware and software compatibilities, and management difficulty of certificate revocation list. Hence, there exists a need to get rid of fully or partially encrypted private key, and to invent key input method to let the private key fully human-memorizable as like the symmetric key.

3.4 Potential Methods to Create Big and Yet Memorizable Secret

One of the many invented methods here to create big and yet memorizable secret is to innovate the graphical password or picture password. From psychological studies, it claims that human graphical memory is stronger than human textual memory. The graphical password is categorized into recognition-based and recall-based methods by Xiaoyuan Suo, Ying Zhu, and G. Scott Owen (2005). For recognition-based method, it can be the types of cognometrics and locimetrics. Meanwhile for recalled-based method, it can be the type of drawmetrics.

Passfaces invented by J. H. E. Davies (1997), as in the US Patent US5608387, is a type of cognometircs, where a user is requested to recognize some pre-selected image sequence of human faces as password. Davies's method has the weakness of low entropy per image. For G. Blonder's method (1996), as in the US Patent US5559961, it is a type of locimetrics, where a user has to select a few areas of an image in sequence as password. Blonder's method is vulnerable to hot-spot attack and shoulder-surfing attack. For Draw-a-Secret scheme by I. Jermyn, A. Mayer, F. Monrose, M. Reiter, and A. Rubin (1999), it is a type of drawmetrics, where a user draw lines and points on a grid in the form as like a hidden hand signature. For this Draw-a-Secret scheme, its weakness is its authentication process for either acceptance or rejection is not exact as in the previous two graphical password methods, but estimation having FAR (False Acceptance Rate) and FRR (False Rejection Rate).

Besides these three main groups of graphical password, there are icon-like graphical password scheme by P. V. Haperen (1997), as in the UK Patent Application GB2313460, and event-based graphical password scheme by J. Schneider (2004), as in the US Patent Application US2004/0250138. The both of these latter methods are cognometric. Their common weakness is that the key space or password space is limited by the fine differentiation capability of human visual memory over images that may have only minor differences. This causes the entropy per image selection to be still unsatisfactory not big enough for the demands of information engineering for the stronger security levels to carry more bits of strength. Hence, there exists a need to boost the key space of graphical password for higher entropy per image selection and yet still human-memorizable and visually differentiable.

Another potential method to have big memorizable secret is to create Chinese language password (CLPW) through Chinese character encodings and their Romanization. T. D. Huang, as in the US Patent US4500872, proposed on 19 February 1985 to use phonetic encoding and symbolic encoding to represent a Chinese character. The character space of Chinese language is huge by more than 16 bits per character and yet human-memorizable and differentiable. This CLPW method can also be extended to other CJKV languages due to the common sharing for the usages of Han characters (漢字 or 汉字) like Chinese Hanzi, Japanese Kanji, Korean Hanja, and Vietnamese Hán Tự. However, the current CLPW has a weakness that it is subject to dictionary attack. Hence, there exists a need to create CLPW resisting the dictionary attack.

There are some inventions to create password that can resist the dictionary attacks. Among them are (i) “System and Method for Generating Unique Passwords” by Martin Abadi, Krishna Bharat, and Johannes Marais (2000) in the US Patent US6141760; (ii) “Password Generation Method and System” by M. R. McCulligh (2003) in the US Patent US6643784; (iii) “Method and System for Automated Password Generation” by P. M. Goal and S. J. Kriese (2004) in the US Patent Application US2004/0168068; (iv) “Method and Apparatus for Password Generation” by M. R. Dharmarajan (2005) in the US Patent Application US2005/0132203; and (v) “Method and System for Generating Passwords” by B. E.

Moseley (2006) in the US Patent Application US2006/0026439. Nevertheless, even though these five methods can resist dictionary attacks, they have lower memorizability. Hence, there exists a need not only to have a password generation method that can resist dictionary attack, but can have high memorizability as well even for a big secret at least and beyond 128 bits.

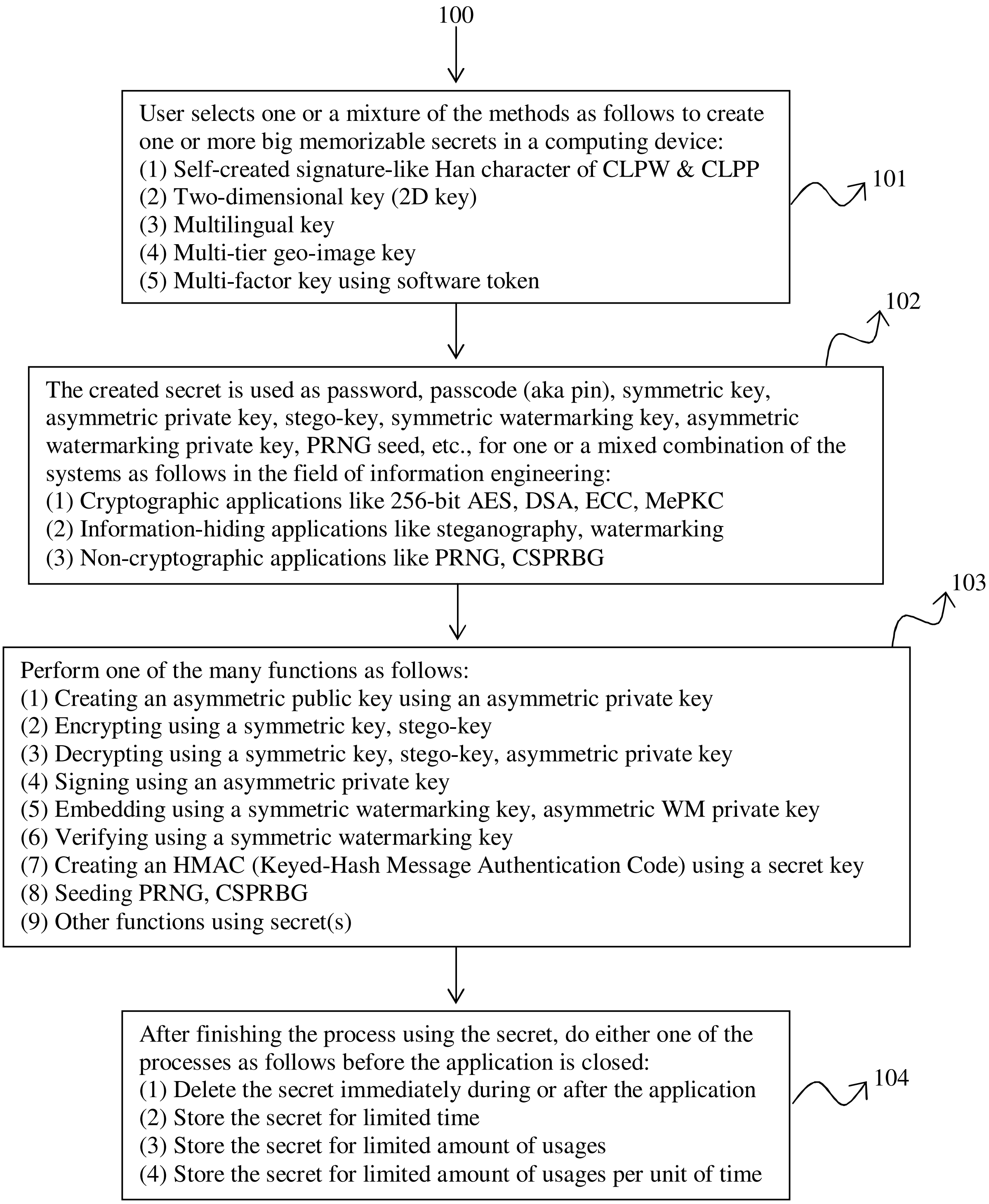
Yet another method to create a memorizable secret bigger than the current prior art was proposed by Whitfield Diffie and William A. Woods (2006) in their patent application filed on 22 June 2006 entitled "Method for Generating Mnemonic Random Passcodes", US Patent Application US2007/0300076. However, the password created by this method is not yet big enough for many applications in the information engineering.

3.5 Methods and Systems to Create Big Memorizable Secret

Accordingly, the present invention mainly provides some methods and systems to create big memorizable secrets. These methods and systems include (i) self-created signature-like Han character; (ii) two-dimensional key (2D key); (iii) multilingual key; (iv) multi-tier geo-image key; and (v) multi-factor key using software token. Every method and system can be used individually or mixed as a hybrid combination. The size of big memorizable secret is at least 128 bits. Figure 3.1 illustrates the main and basic operations for the generations and applications of one or more big memorizable secret(s).

3.6 Potential Applications of Available Big Memorizable Secret

With the realization of big memorizable secret, not only the big secret keys of symmetric key cryptosystems of higher security strength like AES-192 and AES-256 can be realized firstly, but memorizable public-key cryptosystem (MePKC) secondly, and other cryptographic, information-hiding, and non-cryptographic applications thirdly, in the field of information engineering that need big and yet memorizable secret.



User selects one or a mixture of the methods as follows to create one or more big memorizable secrets in a computing device:
(1) Self-created signature-like Han character of CLPW & CLPP
(2) Two-dimensional key (2D key)
(3) Multilingual key
(4) Multi-tier geo-image key
(5) Multi-factor key using software token

The created secret is used as password, passcode (aka pin), symmetric key, asymmetric private key, stego-key, symmetric watermarking key, asymmetric watermarking private key, PRNG seed, etc., for one or a mixed combination of the systems as follows in the field of information engineering:
(1) Cryptographic applications like 256-bit AES, DSA, ECC, MePKC
(2) Information-hiding applications like steganography, watermarking
(3) Non-cryptographic applications like PRNG, CSPRBG

Perform one of the many functions as follows:
(1) Creating an asymmetric public key using an asymmetric private key
(2) Encrypting using a symmetric key, stego-key
(3) Decrypting using a symmetric key, stego-key, asymmetric private key
(4) Signing using an asymmetric private key
(5) Embedding using a symmetric watermarking key, asymmetric WM private key
(6) Verifying using a symmetric watermarking key
(7) Creating an HMAC (Keyed-Hash Message Authentication Code) using a secret key
(8) Seeding PRNG, CSPRBG
(9) Other functions using secret(s)

After finishing the process using the secret, do either one of the processes as follows before the application is closed:
(1) Delete the secret immediately during or after the application
(2) Store the secret for limited time
(3) Store the secret for limited amount of usages
(4) Store the secret for limited amount of usages per unit of time

Figure 3.1 Generations and applications of one/more big memorizable secrets

These cryptographic applications include cryptographic schemes like encryption, signature, key exchange, authentication, blind signature, multisignature, group-oriented signature, undeniable signature, threshold signature, fail-stop signature, group signature, proxy signature, signcryption, forward-secure signature, designated-verifier signature, public-key certificate (aka digital certificate), digital timestamping, copy protection, software licensing, digital cheque (aka electronic cheque), electronic cash, electronic voting, BAP (Byzantine Agreement Protocol), electronic commerce, MAC (Message Authentication Code), key escrow, online verification of credit card, multihash signature, etc.

Those information-hiding applications include steganographic and watermarking schemes like stego-key in steganography, secret key in symmetric watermarking, private key in asymmetric watermarking, etc. Meanwhile, the non-cryptographic applications are PRNG (Pseudo-Random Number Generator) and CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator). Hence, there exist lots of needs to have big memorizable secret for lots of cryptographic, information-hiding, and non-cryptographic applications in the field of information engineering, generally, and security engineering, particularly.

3.7 Future Development of Keys the Secret

These keys the secret need good generation methods (Scalet, 2005) and key management (Fumy & Landrock, 1993; Beach, 2001; Witty, 2001). Wailgum (2008) questioned on whether there were too many passwords or humans were lacking of memory power. In term of memory, there are two forms: Recognition-based and recall-based. Weinshall and Kirkpatrick (2004) presented those recall-based passwords. Bill Gates with Microsoft has once claimed the ending of the passwords (Allan, 2004; Kotadia, 2004; Fried & Evers, 2006).

Subsequently, there are introductions of some password alternatives like Information Card (Wikipedia Contributors, 2008ap), Windows CardSpace (Wilson, 2008), Higgins Project, OpenID (Wikipedia Contributors, 2008am), Identity Metasystem (Jones, 2005; Cameron & Jones, 2006), Identity Selector, digital identity

(Cameron, 2005; Cavoukian, 2006; Wikipedia, Contributors, 2008al), Authorization Certificate, Extended Validation Certificate, etc.

Furnell (2005) analyzed whether human could get rid of passwords and concluded that passwords could not be replaced. Here, if the inventions and innovations on big secret(s) creation methods and their applications are adopted, especially MePKC (Memorizable Public-Key Cryptography), the complicated mentioned password alternatives may be made simpler or at best be avoided. More literatures on password are available at PasswordResearch.com website [URL: www.passwordresearch.com].

For security of asymmetric key cryptosystems, the mathematical hard problems depend on the researchers' creativity and innovation as well as the computing technologies to crack them. For example, the cryptanalytic attacks like Wiener (1990) and so on, that can be discovered in the future, may request for longer asymmetric key sizes and/or other mathematical hard problems. Challenges with awards offered by the PKC services providers to crack certain PKC with certain key sizes are always there for the public to attempt.

Anyway, the practically secure key sizes for symmetric and asymmetric key cryptosystems at different protection periods are always under the regular evaluations by a lot of researchers (Williams, 2002). Website of KeyLength.com [URL: www.keylength.com] ("Cryptographic Key Length Recommendation," No date) is a collection database for lots of documentations on these practically secure key sizes for various applications in security engineering, particularly, and information engineering, generally.

CHAPTER 4 CREATING BIG MEMORIZABLE SECRET (PART 2)

4.1 Passphrase Generation Methods

Civilian cryptosystem applies Kerckhoff's law to have security dependency 100% on the password secrecy. This reflects the fact that key length and key space are very important to ensure enough entropy or randomness to secure a cryptosystem. For stronger password, passphrase is suggested. Currently, there are three methods to generate passphrase: Acronym, full sentence and diceware. Here, an alternate method to diceware is proposed: Coinware (Lee & Ewe, 2006), by using the coin. Coinware uses four coins to generate one hexadecimal digit. The created word lists are in hexadecimal order and can be applied for multilingual passphrase generation. Its exemplary application for Chinese language password is then shown. Readily-made Chinese character word list in the Unicode CJK unified ideographs enables fast hexadecimal reading for random passphrase generation. Hanyu Pinyin and Sijiao Haoma are used to Romanize and uniquely represent each Han character. Jyutping and Rōmaji are then used for Cantonese and Japanese languages, respectively.

Table 4.1 Passphrase generation from acronym

Sentence	Passphrase
Passwords should be impossible to remember and never written down	psbitranwd
Passwords should be impossible to remember and never written down	PsBiTrAnWd
Good or bad, you have to do it.	Goby,htdi.
Good or bad, you have to do it.	Drd,ueoot.
It may be a few sentences. One, two or more.	Imbafs.O,tom.

4.1.1 Acronym

For the passphrase created using the acronym (Schneier, 1996; PGP Corporation, 2006; Yan, Blackwell, Anderson & Grant, 2004), a user has to remember one or a few sentences. Then, the first, second, or last, etc. characters of each word in the sentence(s) are joined to form an acronym. Both alphanumeric and non-alphanumeric ASCII characters may become the character of the acronym. The techniques of *capitalization* and *permutation* may be used to increase the randomness. This acronym will then act as the key. It has the features of high randomness and short key length. The examples of this method are in Table 4.1.

4.1.2 Full Sentence

The passphrase generation using the acronym is sufficient if the key length requirement is short. When the minimum key size demand is long, normally one full sentence or a few short sentences are entered directly as the key (Schneier, 1996; PGP Corporation, 2006; Yan, Blackwell, Anderson & Grant, 2004). So far, it is an open problem to type the entire phrase into a computer with the echo turned off (Schneier, 1996). If the masked password is shown during the password entering process, then it will subject to shoulder surfing attack.

Besides, since the passphrase of full sentence has each word to be selected associatively, its randomness is magnitude-wise high but relatively low if password ciphertext is available. For example, superuser of any computing system can easily obtain ciphertext of the password. By gaining access to the encrypted password, the threats of ciphertext-only attack and frequency analysis of short cryptogram (Hart, 1994; Lee, Teh & Tan, 2006) are then possible. For instance, the unicity distance of English language is about 30 characters. Once the encrypted password is equal to or more than the unicity distance, unique decipherability of the encrypted password will be feasible.

4.1.3 Diceware

Using full sentence for passphrase generation, the word frequency distribution can be under computational analysis (Kučera & Francis, 1967). To get rid of the association of words, diceware (PGP Corporation, 1996) introduced by A. G. Reinhold is an improved passphrase generation method.

There are many software pseudo-random number generators (PRNGs). Unfortunately, they have lots of pitfalls (Eastlake, Crocker & Schiller, 1994) to ease any possible attack. Hence, some hardware random number generators (RNGs) such as coin and dice are very much better than the software PRNGs.

Diceware uses dice to select a word from an ordered word list. The word list can be in any language and based on senary or base-6 numeral system. For the most popular diceware, it is an English word list with 7776 (= 6^5) words. Five dice values are needed to locate one word randomly. Every selected word carries entropy of 12.92 bits. Table 4.2 shows the minimum diceware words for different security levels.

Table 4.2 Minimum diceware words (7776 word list) for different security levels

Key Size (bit)		32	64	72	80	112	128	256
Diceware	word	3	5	6	7	9	10	20
	bit	38	64	77	90	116	129	258

4.1.4 Coinware

In addition to diceware using dice, *coinware* using coin is proposed here. Coin tossing is conducted to generate random passphrase. Each face of the coin is labeled as binary bits “0” and “1”, respectively. Four coin values are used to derive a hexadecimal digit. Therefore, the word list is in hexadecimal order. Table 4.3 shows the conversions between the binary (BIN) and hexadecimal (HEX) numeral systems.

Table 4.3 Conversions between binary and hexadecimal numeral systems

BIN	HEX	BIN	HEX	BIN	HEX	BIN	HEX
0000	0	0100	4	1000	8	1100	C
0001	1	0101	5	1001	9	1101	D
0010	2	0110	6	1010	A	1110	E
0011	3	0111	7	1011	B	1111	F

4.1.5 Monolingual, Bilingual, and Multilingual Word Lists

Having word list and random number generator, computational analysis on word frequency distribution is avoided and random passphrase generation is ensured. For the word list, one may use readily made word list or prepare a new word list.

For readily made word list, it is normally monolingual unless one combines two or more monolingual word lists with different languages. To prepare a new word list, one may go for monolingual, bilingual, or multilingual to suit one's linguistic ability. The purpose to have word list consisting of more than one language is to increase the key space and consequently the key entropy per word.

For the word list, each word has to be unique, short, and memorizable. Start with the shortest word. Then slowly increase the character length of the word until the key space setting of the word list is met. To be easy, one may set the key space of monolingual word list to 4096 (= 2^{12}) or 8192 (= 2^{13}) words. Two or more monolingual word lists can be joined to form bilingual or multilingual word lists.

Nowadays, when language proficiency is excluded, bilingual or multilingual people ("Mother Tongue," No date; Paradowski, No date; Wikipedia Contributors, 2008p, 2008aa, 2008ab, 2008ae, 2008av) exceed the monolingual people (Wikipedia Contributors, 2008a), where English language is a second language due to its soft power of computer language in the recent decades and hard power of British Empire and American influence since the 19th century (Wikipedia Contributors, 2008bb, 2008bc). The language studies (Matthews, 1997; Crystal, 1999; Finegan, 2004; Gordon, 2005; Wikipedia Contributors, 2008ai) are based on text corpus (Wikipedia Contributors, 2008z), branching into a few fields like theoretical linguistics, applied linguistics, and corpus linguistics (Wikipedia Contributors, 2008h, 2008u, 2008y).

The language policy (Wikipedia Contributors, 2008j) of a country has determined the language(s) in education and the speakers of a language. There are various languages spoken and written by various people from different countries (Wikipedia Contributors, 2008c, 2008m, 2008p, 2008ac, 2008ad, 2008aq, 2008ar, 2008au). For example, there is a statistical analysis of Chinese language (Zhang, Xu & Huang, 2000). Till here, we can see that the bilingual and multilingual word lists have their demands and can increase the key space of coinware for higher complexity.

4.1.6 Key Length Requirements of Coinware

It is important to know the key size equivalence for symmetric and asymmetric (RSA, discrete logarithm and elliptic curve) cryptosystems (Schneier, 1996; Williams, 2002; Gehrmann & Näslund, 2005, 2006, 2007) for different security levels. This step enables a user to prepare suitable and sufficiently strong password or passphrase before opening an account and conducting an encryption. Table 4.4 shows this important information (Williams, 2002).

Table 4.4 Key size equivalence for symmetric and asymmetric cryptosystems (bit)

Symmetric Key Size (bit)		48	64	80	112	128	192	256
RSA		480	816	1248	2432	3248	7936	15424
Discrete Logarithm	Field Size	480	816	1248	2432	3248	7936	15424
	Subfield	96	128	160	224	256	384	512
Elliptic Curve		96	128	160	224	256	384	512

The minimum coinware word relies on the key space of the word list. The monolingual, bilingual, or multilingual word lists of 4096, 8192, 12288, 16384, and 24576 words have entropies per word of 12, 13, 13.58, 14, and 14.58 bits, respectively. Table 4.5 shows the minimum coinware words for various word list sizes (WLS).

Table 4.5 Minimum coinware words for various word list sizes (WLS)

Symmetric Key Size (bit)			48	64	80	112	128	192	256
Coinware	WLS 4096	word	4	6	7	10	11	16	22
		bit	48	72	84	120	132	192	264
	WLS 8192	word	4	5	7	9	10	15	20
		bit	52	65	91	117	130	195	260
	WLS 12288	word	4	5	6	9	10	15	19
		bit	54	67	81	122	135	203	258
	WLS 16384	word	4	5	6	8	10	14	19
		bit	56	70	84	112	140	196	266
	WLS 24576	word	4	5	6	8	9	14	18
		bit	58	72	87	116	131	204	262

The current common demands of security levels are 80- and 128-bit for the symmetric cryptosystem. These security levels ensure protection of 5 and 30 years, respectively. From Table 4.5, word list size of 8192 is suitable for monolingual and bilingual users. Monolingual users can use a monolingual word list of 8192 words. Meanwhile, bilingual users can use two unique monolingual word lists of 4096 words each. For multilingual users, word list size of 24576 is suggested where three unique monolingual word lists of 8192 words each can be used.

4.2 Chinese-Character-Encoded Passphrase

For coinware applications, there are readily-made word lists in Unicode (The Unicode Consortium, 2006) for various languages. This is because both coinware and Unicode are in the hexadecimal order for their word lists. This is especially true for the CJK languages of Chinese language, Japanese language, and Korean language that use the Han characters. Word list is also a character list for CJK languages. Here, we discuss on the Chinese language password generation with optional coinware.

In computing system, password is dominated by Roman alphabet or Latin alphabet due to character encoding of ASCII. Here, a pronounceable and memorizable password policy in Chinese language is proposed. Phonetic encoding of Hanyu Pinyin (Popular Book, 2003) and symbolic encoding of Sijiao Haoma (or four corner method) (Wikipedia Contributors, 2007a) are used to create the uniqueness of each Chinese character or Han character for alphanumeric representation.

Based on about 70229 Han characters in the Unihan database of CJK ideographs for the version of Unicode 4.1, each Han character has entropy of 16.1 bits. Five Han characters will satisfy the 80-bit minimum randomness requirement of symmetric key cryptosystem for strong password. Self-created signature-like Han character and passphrase represented by printable ASCII generates shorter, stronger and more memorizable passwords with 27.3 and 85.4 bits per Han character, respectively. Other CJK languages using the Han characters like Cantonese language and Japanese language are also applicable via the pronunciation Romanization systems of jyutping and rōmaji, respectively.

Han character is also called Chinese character (Wikipedia Contributors, 2007h). For the Chinese character input methods, they are either based on pronunciation, character structure, or a combination of pronunciation and character structure (Wikipedia Contributors, 2007c). These methods are closely linked to

Chinese character encodings (Wikipedia Contributors, 2007d) to allow a user to enter a Chinese character. A user normally remembers the pronunciation and/or character structure of a Chinese character to facilitate its input.

Huang (1985) proposed a type of Chinese character encoding to ease Chinese input by using the combination of pronunciation and character structure. This kind of Chinese character encoding can create a key per Chinese character. The maximum size of this encoding is six characters, where there are three characters for phonetic sound, one character for tone, and two characters for character structure. The memorizability of this Chinese-character-encoded key is better than the Environ password, but its security is subject to dictionary attack.

4.2.1 Environ Password

An analogue to the Romanization of Chinese language to have alphabets and digits is the Environ password (Anderson, 2001, p. 49). Good memorizability exists when it is linked to a learnt language. For English language, U.K. government introduced the case insensitive Environ password in October 2005 for short-term protection (Wikipedia Contributors, 2007b). It has an 8-character key pattern as in Table 4.6. This pronounceable password has 34.9 bits per unit.

Table 4.6 Environ password

Form	[consonant - vowel - consonant - consonant - vowel - consonant - digit - digit] [consonant - vowel - consonant – digit - consonant - vowel - consonant - digit]
Example	pinray34, yankan77, supjey56, kinkin99; pin3ray4, yan7kan7, sup5jey6, kin9kin9

4.2.2 Unicode

Unicode unifies the Han characters of CJK languages into CJK unified ideographs or Unihan under ISO 10646. There are three major blocks of Han characters or Chinese characters in the Unicode character encoding: CJK unified ideographs, CJK unified ideographs extension A, and CJK unified ideographs extensions B. For the mean time, Unicode Consortium is preparing the CJK unified ideographs extension C and CJK unified ideographs extension D (Wikipedia

Contributors, 2008at). The CJK unified ideographs extension C with 4251 Han characters will be included into the next version after Unicode 5.1.

For Unicode 4.1, the first block lists the Han characters from [4E00] to [9FBB] in hexadecimal value. The second block lists from [3400] to [4DB5]. The third block lists from [20000] to [2A6D6]. Hence, there are three readily made word lists or character lists for Chinese language. These word lists have 20924, 6582 and 42711 words or characters, respectively. In addition, there are CJK compatibility ideographs having 12 characters. For a combined word list, it is a key space of 70229 characters. After radical exclusion, the key space has about 70000 characters. This forms a Chinese language word list with high entropy of 16.10 bits per Han character.

To start coinware, first flip or toss a coin to randomly select a binary bit "0" or "1". If bit "0", the first and second blocks of CJK unified ideographs and CJK unified ideographs extension A are chosen. If bit "1", the third CJK block of CJK unified ideographs extension B is chosen. Then continue with coin tossing to obtain four coin values representing four binary bits. These four binary bits are converted into one hexadecimal digit. Repeat coin tossing to get four coin values for another three rounds. Four randomly obtained hexadecimal digits will locate the unique Han characters in the previously selected CJK block(s). These three blocks are available at [URL: http://www.unicode.org/charts/]. If the hexadecimal digits do not hit any Han character, get another set of hexadecimal digits. Coming to here, the selected Han character will need Chinese character Romanization to enable computer input.

4.2.3 Chinese Language Password (CLPW)

Zhonghua Zihai in 1994 has 85,568 Chinese characters (Wikipedia Contributors, 2007d). It means a Chinese character may have entropy of 16.38 bits when the Unicode Unihan database is further enlarged. For key security, this is an advantage over the 6.57-bit ASCII characters, which are used for the Latin languages. For computers with support of Chinese character encoding, Chinese language password (CLPW) is shorter for the similar key size of ASCII-based password. This indicates better memorizability. For computers without support of Chinese character encoding, which are general for majority of the computers, Romanization of Chinese language is needed to create the same advantage in term of memorizability.

Chinese input methods and Chinese character encodings can be used to Romanized CLPW. The Romanization of Chinese language is either based on pronunciation, character structure, or a combination of the both. To uniquely represent a Chinese character, Huang (1985) has a good reference for CLPW Romanization, where both Huang pronunciation and character structure are used to create a Chinese-character-encoded word with a maximum of 6 characters.

However, this approach requires modernization. The pronunciation system of Hanyu Pinyin （汉语拼音） (Popular Book, 2003) and character structure system of Sijiao Haoma or four-corner method （四角号码） (United Publishing House, 2001, 2002; Wikipedia Contributors, 2007a) are proposed to create a Chinese language password. In Hanyu Pinyin, there are 415 unique syllables with 22 initials (or onsets) and 39 finals. This pronunciation system is illustrated in Table 4.7.

Table 4.7 Phonetic encoding of Hanyu Pinyin (Mandarin-based)

Initial (22)	nil	b	p	m	f	d	t	n
	l	g	k	h	j	q	x	z
	c	s	zh	ch	sh	r		
Final (39)	a	o	e	ê	ai	ei	ao	ou
	an	en	ang	eng	ong	i	ia	io
	ie	iao	iu	ian	in	iang	ing	iong
	u	ua	uo	uai	ui	uan	un	uang
	ueng	ü	üe	üan	ün	-i	er	

N.B.: For Romanization, ê and ü can be represented by [oe] and [v], respectively.

In addition to initials and finals, there are 5 tone marks. These tone marks are numbered as 1, 2, 3, 4 and 5 in corresponding with Yinping （阴平）, Yangping （阳平）, Shangsheng （上声）, Qusheng （去声） and Qingsheng （轻声）.

Then, the 4+1-digit Sijiao Haoma is added to describe the character structure of a Chinese character. The upper left number is the first digit. The upper right number is the second digit. The lower left number is the third digit. The lower right number is the fourth digit. The fifth digit is Fuhao or attached number （附号）, which represents the middle character structure on the right hand side. Figure 4.1 shows a Chinese poem to easily memorize the Sijiao Haoma. Table 4.8 shows the strokes represented by Sijiao Haoma.

横一垂二三点捺
叉四插五方框六
七角八八九是小
点下有横变零头

Figure 4.1 Chinese poem for easy memorization of Sijiao Haoma

Table 4.8 Character structure encoding of Sijiao Haoma

Stroke name （笔名）	Digit （号码）	Stroke （笔形）
Tou （头）	0	亠
Heng （横）	1	一
Chui （垂）	2	丨 丿 亅
Dian （点）	3	丶
Cha （叉）	4	十 メ
Chuan （串）	5	扌 丰
Fang （方）	6	口 囗
Jiao （角）	7	乛 厂
Ba （八）	8	八 人 入
Xiao （小）	9	小 忄

Table 4.9 Forms of Romanized Chinese-character-encoded words

Form	[Hanyu Pinyin] (Tone Mark) [Sijiao Haoma] (Fuhao)				
Example	han3714	han43714	han37140	3714han	3H7A1N4

Finally, the Hanyu Pinyin, tone mark, Sijiao Haoma, and Fuhao are joined to form a Romanized Chinese-character-encoded word as in Table 4.9 for the Chinese character of Han （汉）. Tone mark and Fuhao are optionally included. For the computer input of CLPW for Han character, the Hanyu Pinyin and Sijiao Haoma can be typed side by side. If the coin tossing gives a 5-digit hexadecimal string of [06C49], then the Han character of （汉） is selected from the Unicode. Table 4.9 shows the possible forms for the Chinese character Romanization of （汉） [Hanyu Pinyin = han4] [Sijiao Haoma = 37140].

This creates a Chinese-character-encoded word ranging from 5 to 12 ASCII characters. Several Chinese-character-encoded words can be used as a Chinese language password. The capitalization and permutation can slightly increase the entropy. However, it is subject to dictionary attack.

4.2.4 Key Length Requirements for Coinware in Chinese Language

Referring to the created Han character combined list in Section 4.1, it is 16.10 bits per Han character. Using this key entropy, minimum coinware words at different security levels for Chinese language password can be derived as in Table 4.10.

Due to high key entropy of Chinese language word list, it is obvious to observe the significant drop of minimum coinware words as from Tables 4.5 and 4.10. As Japanese language and Korean language are using the Han characters as well, similar word lists with large key space can be created. For Chinese language family, it is applicable to Mandarin, Wu, Cantonese, Min, Jin, Xiang, Hakka, Gan, Hui, and Ping languages/dialects with speaking population of 800, 90, 80, 50, 45, 35, 35, 20, 3, and 0.2 millions, respectively.

Table 4.10 Minimum coinware words for Han character combined list (70229 words)

Symmetric Key Size (bit)		48	64	80	112	128	192	256
Coinware (Chinese Language)	word	3	4	5	7	8	12	16
	bit	48	64	80	112	128	193	257

4.2.5 Self-Created Signature-like Han Character

The combination of 415 Hanyu Pinyin syllables, 5 tone marks, and 10,000 Sijiao Haoma numbers are more than enough to encode all the Han characters available at present. In order to increase the randomness or entropy of Han character, the creation of new Han character is a must.

This situation happens in real life for the individual name in gaining uniqueness. The created Han character is also signature-like. For Han character creation, it may follow the six methods of Liushu （六书） (Huang, 2002). The Liushu includes Xiangxing （象形） (pictograms), Zhishi （指事） (ideograph), Huiyi （会意） (logical aggregates), Xingsheng （形声） (pictophonetic compounds), Jiajie （假借） (borrowing), and Zhuanzhu （转注） (associate transformation) (Wikipedia Contributors, 2007h; Xu, 2001; Luo, 1990, 2003).

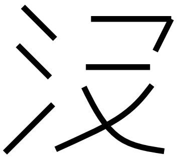


Figure 4.2 Example of self-created signature-like Han character by modifying the Han character of （汉） from [Hanyu Pinyin = han4] and [Sijiao Haoma = 37140] to [Hanyu Pinyin = han4] and [Sijiao Haoma = 37141]

An example of created Han character is shown in Figure 4.2. The Han character of (汉) is modified from [han437140] to [han437141] by adding a horizontal stroke between the upper right corner and lower right corner. Self-created signature-like Han characters enlarge the key space of Chinese language password to 4,150,000. When tone mark and Fuhao are included, it becomes 207,500,000 or entropy of 27.63 bits per Han character. The efficiency of Chinese language password is greatly increased.

4.2.6 Self-Created Chinese Language Passphrase (CLPP)

To further increase the entropy of Chinese language password, we can have self-created Chinese language passphrase (CLPP). At least one non-alphanumeric character has to be included together with *capitalization*, *permutation*, *character stuffing*, and text-based semantic noises. Character stuffing is like bit stuffing in data communication to enable the syllable length at a fixed value of 6. It is 6 because the maximum syllable length is 6, excluding the tone mark.

Adding fixed syllable length, tone mark, Sijiao Haoma with Fuhao, and one non-alphanumeric character together, a string of 13 ASCII characters is obtained as a basic unit of a self-created CLPP. The non-alphanumeric character is used as a separator and text-based semantic noise.

Table 4.10 shows examples of self-created CLPP. This Chinese-character-encoded passphrase has 85.41 bits per Han character. It has good memorizability, resistance to dictionary attack, and suitability for general password usages.

Table 4.11 Forms of self-created Chinese language passphrase for （汉）

Form	No character stuffing	With character stuffing & noise	Capitalization & permutation
Example	han4&37140	h@n4***&37140	37140&HaN4***

4.2.7 Cantonese Language Password Using Jyutping

Han unification of Unicode builds Han characters database for CJK languages (Chinese, Japanese, and Korean). The proposed password and passphrase generation method can be applied to any CJK languages using the Han characters by changing the pronunciation Romanization system. The character structure encoding of Sijiao Haoma remains the same for all the Han characters in any CJK languages.

Cantonese language is used by a global population of about 80 millions. Being the official language in Hong Kong SAR (Special Administrative Region) and Macau SAR of PRC (People's Republic of China), the regulation works of Cantonese language are done here. It shares majority of the Han characters with Chinese language in Mandarin except those Han characters in the HKSCS (Hong Kong Supplementary Character Set). For HKSCS-2004, it has 4941 Han characters as in year 2004 under ISO 10646 standard. Hence, it is compatible with Unicode, which implements the ISO 10646 standard.

There are many Cantonese pronunciation systems. Among them, two systems are Romanized and computer friendly. One of them is the standard Cantonese pinyin or HKED （《常用字廣州話讀音表》拼音方案） （「教院式」拼音方案）. This is the only pronunciation Romanization system accepted by Education and Manpower Bureau of Hong Kong and Hong Kong Examinations and Assessment Authority. Another is jyutping proposed by LSHK (The Linguistic Society of Hong Kong) in year 1993.

Nowadays, regulation works of Cantonese pronunciation for Unicode adopt jyutping system. Han characters in Unicode are matched with jyutping, where the lists are downloadable from the URLs of [http://www.iso10646hk.net/jp/index.jsp] and [http://www.info.gov.hk/digital21/eng/structure/jyutping.html].

Jyutping can also be applied into the coin tossing of coinware. As for the hexadecimal strings of [03400], [04E00], and [0E000], the jyutping for these Cantonese characters are [jau1], [jat1], and [mou5], respectively.

Table 4.12 Phonetic encoding of jyutping in Cantonese language

Initial (20)	nil	b	p	m	f	d	t	n	l
	g	k	ng	h	gw	kw	w	z	c
	s	j							
Final (59)	i	ip	it	ik	im	in	ing		iu
	yu		yut			yun			
	u	up	ut	uk	um	un	ung	ui	
	e	ep	et	ek	em	en	eng	ei	eu
			eot			eon		eoi	
	oe		oet	oek			oeng		
	o		ot	ok		on	ong	oi	ou
		ap	at	ak	am	an	ang	ai	au
	aa	aap	aat	aak	aam	aan	aang	aai	aau

In jyutping system, there are 20 initials and 59 finals as in Table 4.12. These initials and finals construct about 629 syllables for Cantonese language as compared to 415 syllables for Chinese language in Mandarin. For tone mark, 6 distinct tone contours are used for 9 tones. For completeness, the jyutping has syllables that have no matching Han character. Nevertheless, in the application for Cantonese language password, all jyutping syllables may be useful for self-created password.

Table 4.13 shows the examples of Cantonese language password. It is similar to Chinese language password in Mandarin. Sijiao Haoma is exactly encoded. For jyutping, the maximum syllable length is 6. Capitalization, permutation, and character stuffing can be used to generate self-created signature-like Cantonese language password and passphrase. The key space of self-created Han characters in Cantonese can reach 377,400,000 keys or 28.49 bits per Han characters.

Table 4.13 Forms of self-created Cantonese language passphrase for （汉）

Form	Traditional Chinese （漢）	Simplified Chinese （汉）	With character stuffing
Example	hon3&34185	hon3&37140	hon3***&34185

4.2.8 Japanese Language Password Using Rōmaji

In Japanese language, there are four writing systems: Two syllabaries of Hiragana (平仮名) and katakana (片仮名), one logogram of kanji (漢字), and one Romanization of rōmaji (ローマ字). The most widely used Hepburn Romanization is adopted for rōmaji. The password generation method for Chinese language password can be used for Japanese kanji via the combination of rōmaji and Sijiao Haoma.

Firstly, obtain the Sijiao Haoma with Fuhao for the Japanese word in kanji. Then, the kanji is converted to rōmaji for pronunciation Romanization. Character stuffing is longer for Japanese language password as the kanji is having variable number of syllables from a minimum of one syllable. For Hepburn Romanization, there are about 132 syllables.

Table 4.14 shows examples of kanji passwords. To avoid dictionary attack, self-created kanji with character stuffing, capitalization, and permutation, as in

Section 4.2.5 can be used. Coinware allows random selection of Han characters (Lee & Ewe, 2006).

Table 4.14 Forms of Japanese language password for （大）, （漢）, and （山）

Form	dai (大) (だい)	kan (漢) (かん)	yama (山) (やま)
Example	dai&40800	kan&34185	yama&22770

4.2.9 Key Length Requirements for Various Chinese Key Spaces

For unbreakable encryption, the key size has to be at least the same with message size as in one-time password (Shannon, 1949). So far, the full Unihan database of 70229 Han ideographs in Unicode 4.1 is used to build the Chinese language password and passphrase. In the Han unification of Unicode, Han ideographs are called as Hanzi in Chinese language, Kanji in Japanese language, and Hanja in Korean language.

If only the basic Unihan database of 20948 Han ideographs in the CJK unified ideographs are used, by excluding the CJK unified ideographs extension A and CJK unified ideographs extension B, the entropy will drop from 16.10 to 14.35 bits per Han character. Then more Han characters are required to fulfil the key length requirements. Hence for short, strong, and memorizable password, self-created signature-like Chinese language password and passphrase are in favourite. The situation of various Chinese key spaces is shown in Table 4.15.

Table 4.15 Minimum key lengths for various Chinese key spaces (in Han character)

Database	Key Space	Entropy (bit / Han char.)	Minimum Key Length (in Han character)			
			80-bit	128-bit	192-bit	256-bit
Basic Unihan	20948	14.35	6	9	14	18
Full Unihan	70229	16.10	5	8	12	16
Self-created Han Character	166,000,000	27.31	3	5	8	10
Self-created Cantonese	377,400,000	28.49	3	5	7	9
Self-created Passphrase	95^{13}	85.41	1	2	3	3

4.2.10 Example to Create CLPW and CLPP

This section shows an example on how self-created signature-like Han character is encoded to create big and yet memorizable secret. For the word etymology of “Chin” and “Han”, they are originated from the names of two early dynasties called Qin （秦） and Han （漢） in China. Even though there are many rounds of renaming in Chinese language for the country of China, its English name remains unchanged till today in carrying the phoneme of “Qin” for “Chin”. Therefore, Chinese character is also called Han character （漢字 or 汉字）. The repertoire size of Han characters is 85,568 in the dictionary of Zhonghua Zihai (Word Dictionary of Chinese Language) published in 1994. Han characters are used in CJKV languages, in which it is called Hanzi in the Chinese language, Kanji in the Japanese language, Hanja in the Korean language, and Hán Tự in the Vietnamese language.

It is to note that the entropy of Han characters is higher than the ASCII characters. Due to the logographic type of language, Han characters carry visual meaning and hence are easily memorizable. In other words, Han characters have the intrinsic features of high entropy and good memorizability, which mean their suitability for the creation of big and yet memorizable secret. Nevertheless, Han characters have input problem. The number of Han characters is too many to be represented by a single keyboard. Another problem is that direct application of Han characters as password the secret is vulnerable to guessing attack, dictionary attack, and pre-computation attack.

To solve the first problem, a Han character can be encoded using its character structure (or symbolic shape) and/or phonetic pronunciation based on ASCII characters. This process is called Romanization. For example, when pronunciation system of Hanyu Pinyin （汉语拼音） and character structure system of Sijiao Haoma (or four-corner method) （四角号码） are used to encode and Romanize the Han character of {han} （汉） in simplified form, the code is {han4} from Hanyu Pinyin and {37140} from Sijiao Haoma, forming one of many possible codes like {han437140} called CLPW (Chinese Language Password). However, the second problem of vulnerability to guessing attack, dictionary attack, and pre-computation attack, has not yet been solved.

To solve the second problem, the randomness of the CLPW using Han character has to be increased. A Han character from any encoding like Unicode encoding can be modified to become a self-created signature-like Han character new to the current available repertoire of Han characters. Phonetic pronunciation system and character structure system using ASCII characters can be used to encode and Romanize the self-created signature-like Han character into a CLPW that can resist the guessing attack and dictionary attack.

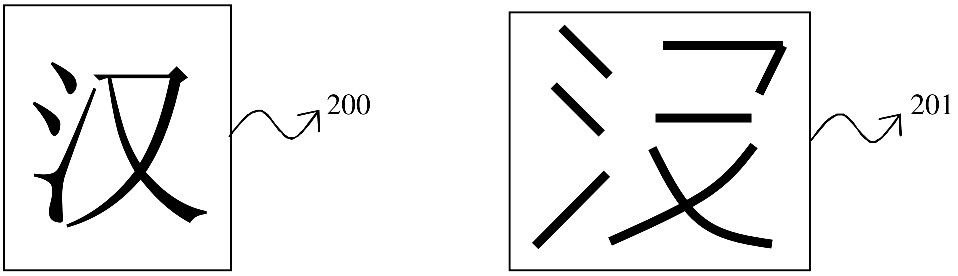


Figure 4.3 Example of self-created signature-like Han character from {han}（汉）

Figure 4.3 illustrates an example of self-created signature-like Han character by modifying the Han character of {han} (汉) in Box 200 of Figure 4.3 (left) from {hanyu pinyin = han4} and {sijiao haoma = 37140} to Box 201 of Figure 4.3 (right) {hanyu pinyin = han4} and {sijiao haoma = 37141}. In other words, the CLPW has been modified from {han437140} to {han437141}. The adoption of self-created signature-like Han character shares the similar habit with Chinese people to use a general name aliasing with another rare name. A name using frequently used Chinese characters allows easier memorizability and pronunciation, but harder differentiation due to name clashing. A second alias name using rarely used Chinese characters helps to make a person’s name unique and differentiable from the others, but carries a problem of harder pronunciation. Hence, pronounceable name is for easy calling and unique name is for easy differentiation.

Self-created signature-like Han characters enlarge the key space of CLPW to 4,150,000. When tone mark and Fuhao （附号） are included, it becomes 207,500,000 or entropy of 27.63 bits per Han character. The efficiency of CLPW is hence greatly increased. To further increase the randomness, a Chinese language password (CLPW) can be upgraded to a Chinese language passphrase (CLPP) by adding textual semantic noises like character stuffing, capitalization, permutation, punctuation marks, misspelling, mnemonic substitution, and/or alternative symbols from ASCII mutual substitution table. One unit of CLPW can be set to a fixed length like 13 ASCII characters or other size, and a few units of CLPW form a unit of CLPP. For a unit of CLPW, its 13 ASCII characters are formed from phonetic syllable of length 6, tone mark of length 1, Sijiao Haoma with Fuhao of length 5, and non-alphanumeric character as a separator of length 1.

Character stuffing is like bit stuffing in data communication to enable the syllable length at a fixed value of 6. It is 6 because the maximum syllable length is 6 in Hanyu Pinyin, by excluding the tone mark. Of course, other phonetic pronunciation systems, especially Chinese dialects and CJKV languages, like jyutping for Cantonese language and rōmaji for Japanese language, can be used as well. Similarly, other encodings of Han characters could be used. For the example of 13-character CLPW with textual semantic noises using the Han character of {han}（汉）, it can be in the forms of {h@n4***&37140}, {37140&HaN4***}, and so on. When the textual semantic noises are good enough from prediction, the ideal entropy of fully random absolute rate at entropy of 85.41 bits per unit of CLPW (or unit of Han character with modification and added noises) can be approached.

A few serial units of CLPW form a CLPP that has good memorizability, resistance to guessing attack and dictionary attack, as well as suitability for general usages. CLPP of size beyond 128 bits can realize the AES-128, AES-192, AES-256, DSA-256, ECC-256, and so on. When CLPP is used for MePKC operating on the platforms of FFC and ECC, even the pre-computation attack can be avoided. Table 3.3 shows the numbers of CLPP units for various key sizes. People knowing Han characters can memorize a CLPP with 2 to 4 units of CLPW as easy as remembering a person's name using rarely used Han characters.

Nevertheless, the current prior art of single-line key/password input field is not that friendly when there are two or more CLPW. There exists a user interface problem to input password with long key size in a single line. This problem happens also to other passphrases having a lot of characters. Whenever there is a pause or interrupt during the input process of a passphrase, it is hard to determine the starting points of every word or unit of a passphrase. In other words, a long passphrase like three to four units of CLPP has to be entered instantly without an interrupt or error. Any uncertainty in keying in a passphrase to a single-line key field requires the whole re-keying process of that passphrase.

4.3 Two-Dimensional (2D) Key

Conventionally, single-line key field is used to input a key. The selection of a key depends on the factors of memorizability and security. The minimum key sizes for symmetric and asymmetric key cryptosystems are 80 and 160 bits, respectively.

For symmetric key cryptosystem, National Institute of Standards and Technology (NIST) of USA proposed security level of 80-bit key to be phased out by year 2015 and used until year 2010 (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b). US government has an export policy to control the power of cryptographic algorithm by setting the maximum key size. The current export limit of symmetric key size has been raised from 40 bits to 128 bits.

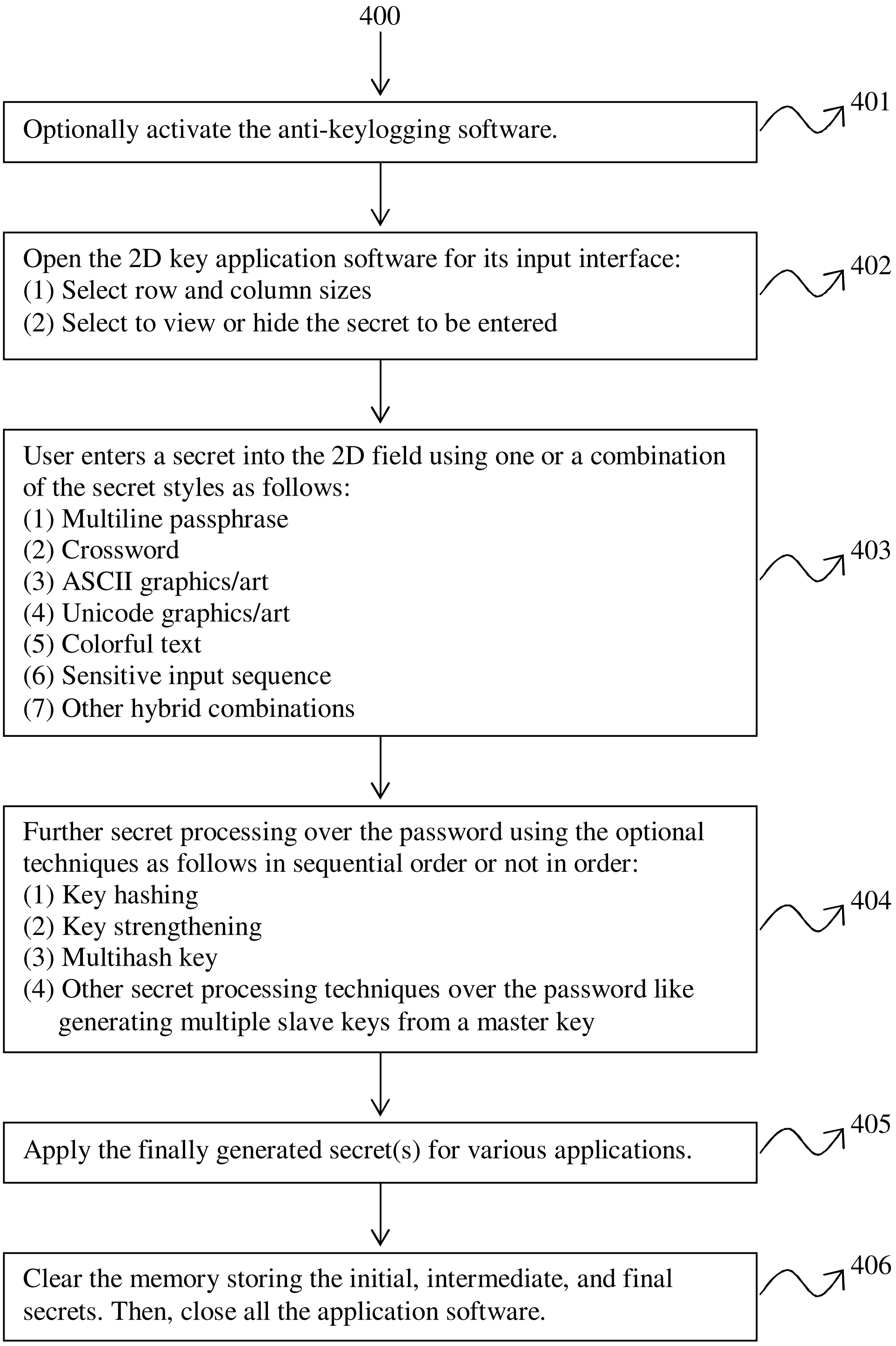
For the symmetric key cryptosystem of Advanced Encryption Standard (AES), there are three key sizes: 128, 192, and 256 bits. The asymmetric key cryptosystems, which demand for the minimum private key size at 160 bits by year 2010, are finite field cryptography (FFC) and elliptic curve cryptography (ECC). FFC and ECC are based on the mathematical hard problems of discrete logarithm problem and elliptic curve discrete logarithm problem, respectively. The corresponding sizes of private keys to the AES are 256, 384, and 512 bits, respectively (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b; Gehrmann & Näslund, 2005, 2006, 2007). The symmetric key is normally remembered by brain; whereas the asymmetric private key is encrypted using another symmetric key.

ASCII characters have entropy of 6.57 bits per character. Therefore, the nominal bit of an ASCII character is 8 bits, but its effective bit is 6.57 bits. To cater for the different symmetric key sizes at 80, 96, 112, 128, 192, and 256 bits as in Table 3.3, 13, 15, 18, 20, 30, and 39 ASCII characters are needed, respectively. An amount of 15 ASCII characters is perhaps still affordable and convenient for the human users. However, higher amounts may introduce two problems. Memorizability is the main problem. The difficulty to type a long passphrase into a computer will be another open problem (Schneier, 1996).

Here, a high-entropy key input method called 2-dimensional (2D) key as in Figure 4.4 is proposed to solve these problems. 2D key facilitates particularly the recognition of reference points of each sub-unit of a passphrase like CLPW of CLPP, and generally the creation of various secret styles of 2D key like multiline passphrase, crossword, ASCII graphics/art, Unicode graphics/art, colorful text, sensitive input sequence, and two or more of their hybrid combinations as partially illustrated in Figures 4.6-4.9, for Latin language users.

It uses a 2D display as user interface to improve the human factors of memorizability and input of ASCII characters from keyboard. The 2D key has the styles of multiline passphrase, crossword, ASCII art, colorful text, or sensitive input sequence. It can resist dictionary attack and fulfil the demands of human-memorizable key sizes even until 256 bits, which is impossible by using the single-line passphrase. CLPW and CLPP may also be used in the 2D key.

In addition to fulfilling the various key sizes of symmetric key cryptosystem, 2D key has novel revolution to the private key storage of asymmetric key cryptosystem. For the prior arts, we have encrypted private key, split private key, and roaming private key. With the introduction of 2D key, there shall be no more need to store the private key in a computing system, but inside the brain as like the symmetric key. This allows the creation of memorizable public-key cryptosystem (MePKC) as discussed in Chapter 9. MePKC has the special features of mobility, lower cost, and higher efficiency.



Optionally activate the anti-keylogging software.

Open the 2D key application software for its input interface:
(1) Select row and column sizes
(2) Select to view or hide the secret to be entered

User enters a secret into the 2D field using one or a combination of the secret styles as follows:
(1) Multiline passphrase
(2) Crossword
(3) ASCII graphics/art
(4) Unicode graphics/art
(5) Colorful text
(6) Sensitive input sequence
(7) Other hybrid combinations

Further secret processing over the password using the optional techniques as follows in sequential order or not in order:
(1) Key hashing
(2) Key strengthening
(3) Multihash key
(4) Other secret processing techniques over the password like generating multiple slave keys from a master key

Apply the finally generated secret(s) for various applications.

Clear the memory storing the initial, intermediate, and final secrets. Then, close all the application software.

Figure 4.4 Operation of 2D key input method and system

4.3.1 Related Work: Single-Line Key/Password Field

Conventionally, whenever secret is used as the authentication method, single-line key field will be the area for a user to enter a key. For the current longest possible key, it is a single-line passphrase. For passphrase, it can be formed from acronym, sentence, diceware, and coinware. Nevertheless, there is a limit due to the problems of memorizability and ASCII character input from keyboard. The first problem is due to the human factor; whereas the second is due to the user interface. These problems prohibit the applications of symmetric key sizes at higher security levels whenever a user cannot remember and/or conveniently enter a long single-line passphrase.

4.3.2 Related Work: Key Strengthening

Key strengthening is also called key stretching. It is used to make a weak key stronger. There are two forms of key strengthening. One uses password supplement (Manber 1996; Abadi, Lomas & Needham, 1997; Abadi, Needham & Lomas, 2000), and another uses many rounds of hash iterations (Kelsey, Schneier, Hall & Wagner, 1997). In this thesis, key strengthening is applied to achieve larger protection periods for symmetric and asymmetric cryptosystems like AES and MePKC.

$$S = n * L * R / P \tag{4.1}$$

S = Key space

n = Number of networked computers

L = Maximum lifetime of a key in years

R = Number of guesses per unit of time per unit of computer

P = Probability that a key can be guessed in its lifetime

Typical values:

$n = 10^9$ units = 29.9 bits

L = 4, 10, 20, 30, 300 years = 2, 28.2, 29.2, 29.8, 33.1 bits

R = 1.5 x 10^{7} s^{-1} = 23.8 bits (best performance in year 2005)

R = 1 s^{-1} = 0 bit (using key strengthening)

P = 10^{-6} = -19.9 bits

Equation (4.1) is a password length equation. When key strengthening is used, R becomes 1 guess per second and the variety of computer is a main factor to set the number of hash iterations. The computer performance of a variety of computers varies from 1 time for the slowest computer to 20 times for the fastest computer. This contributes a factor of $\log_2 20$ = 4.3 bits to Equation (4.1). Moore's Law states that the number of transistors on an integrated circuit for minimum component cost doubles every 24 months (Wikipedia Contributors, 2008v).

$$S = (n * L * R / P) * 2^{4.3} * 2^{L/2} \qquad (4.2)$$

When the variety of computers and Moore's Law are considered, it becomes Equation (4.2). From Equation (2), key strengthening can make a weak key to become 19.5 bits stronger.

4.3.3 2D Key Input Method

For single-line passphrase, the numbers of ASCII characters for different symmetric key sizes are shown in Table 3.3. An amount of 15 ASCII characters or 96 bits is a memorizabilty limit for many human users. This fact is statistically proven by Florencio and Herley (2007) in their large-scale study of web password habits for half a million users over a 3-month period, where the average key size is 40.54 bits ranging from exclusive 0 to inclusive 100 bits or]0, 100]. The difficulty of user interface to enter a key using keyboard into the single-line key field is another big problem.

The problems of human factor and user interface limit the practical application of symmetric key cryptosystem to be at the key size of 96 bits with 10 years of protection. Using key strengthening, the 96-bit key can be made 19.5 bits stronger, and 20-year protection is the maximum theoretical limit.

The 2-dimensional (2D) key input method is created to allow high-entropy keys. Figure 4.5 displays the pseudocode of 2D key input method. It tries to solve the human factor of memorizability and user interface of key input. 2D key has a 2-dimensional display alike a 2D matrix, where each character of a key is an element of the matrix.

```
1.0 User selects row size.
2.0 User selects column size.
3.0 User enters ASCII characters or Unicode symbols one by one.
4.0 User ends the key input by pressing the “Enter” key.
5.0 Computer hashes the input key.
6.0 Computer compares the hashed key with the stored hash.
        6.1 If the hashes match, authentication is verified.
        6.2 If the hashes mismatch, authentication is rejected.
```

Figure 4.5 Pseudocode of 2D key input method and system

The font used for 2D key has to be fixed-width font (Wikipedia Contributors, 2007f). Fixed-width font is also called non-proportional font and monospaced font. It is a typeface using fixed width for every glyph. Examples of fixed-width fonts are *Courier* for ASCII and *MS Mincho* for Unicode. When ASCII encoding is used, the 2D key has 6.57 bits per character. Meanwhile, when Unicode is used, it has 16 bits per character. Even though Unicode-based 2D key has higher entropy, it is inconvenient to enter a Unicode symbol for the mean time, and the fixed-width font for all the Unicode symbols has not yet been created. Hence, ASCII-based fixed-width font is used currently for the discussions as well as prototype demonstration.

To use 2D key input method and system, firstly a user needs to select the row size and column size of the 2D matrix for 2D key. The currently built prototype has a maximum row size or height of 10 characters, and a maximum column size or width of 13 characters. The column size is set at 13 due to the Chinese-character-encoded passphrase proposed in Section 4.2 has a maximum size of 13 per Chinese character. Alternatively, it can be a word in English language or other languages that has a size of 13 characters per word with character stuffing.

The input styles of 2D key are multiline passphrase, crossword, ASCII art, Unicode art, colorful text, and sensitive input sequence. Multiline passphrase, crossword, and ASCII art are currently implemented in the prototype; whereas Unicode art, colorful text, and sensitive input sequence require additional supports.

After selecting the row size and column size, the user can input ASCII characters using keyboard as the elements of the 2D matrix. The input characters can have any style or a mixed style of 2D key. These styles have good memorizabilty, and the 2D nature of 2D key generates more references at the user interface for key input. Single-line key field has only one reference at the first location of the only line. 2D key has a number of horizontal lines and each first location of the horizontal lines acts as references for key input. In addition, the first locations of the vertical lines can be secondary set of references for key input. This solves the location recognition problem of user interface in facilitating a user to enter a high-entropy key by having more indexed references.

Good memorizability allows a user to repeat a high-entropy key. The elements of 2D matrix can be either partially, fully, or extraordinary filled. To fill extraordinarily means adding some extra trailing characters as noise after the last element of the 2D matrix. The characters entered into the 2D key field are read by a computer line by line horizontally from top to bottom, hashed, and processed as usual alike the single-line key field. The hashing process is one round if key strengthening is not used. If key strengthening is used, the hashing iteration is set according to the computer response time per access ranging from 0.05 to 1 second, or any other tolerable ranges.

The advantages of 2D key are good memorizability, high-entropy key, more references at the user interface to facilitate key input, and resistance to dictionary attack. Even pre-computation attack can be avoided if the 2D secret is used on the platform of MePKC. Its disadvantages are more time for key input and possible shoulder-surfing attack. Nevertheless, for a long passphrase having many individual units like word, the key input time of 2D key is faster than the single-line key field whenever there is some interrupt and the user has forgotten the input sequence. This is because only that particular sub-unit has to be re-keyed in and not the whole secret, such like the secret style of multiline passphrase.

4.3.4 Styles of 2D Key: Multiline Passphrase

For single-line key field, it is hard to input a high-entropy single-line passphrase due to the problem of user interface. A user may lose the reference of starting character of a word in a passphrase. Using 2D key, multiline passphrase can be input, where each line consists of one word of a passphrase. Each word is padded to the longest word in the passphrase. The padding character can be any ASCII character and acts as a text-based semantic noise. Figure 4.6 shows a 2D key example using multiline passphrase. Its dimensions are 4 x 5, and uses character '*' as the padding character. This 2D key has entropy of 131 bits.

```
Have*
a****
happy
day!*
```

Figure 4.6 Styles of 2D key: Multiline passphrase

```
HAPPY*
O*R*K*
M*INCH
E*D*U*
SPELLS
```

```
WELCOME
:TO::::
::USE::
::::2D:
::::KEY
```

Figure 4.7 Styles of 2D key: Crossword

4.3.5 Styles of 2D Key: Crossword

The second style of 2D key is crossword. Instead of horizontal and vertical multiline passphrase, a user can enter a mixture of horizontal, vertical, and slanted passphrases. Figure 4.7 shows two 2D key examples using crossword. Their dimensions are 5 x 6 (left) and 5 x 7 (right), and use characters '*' and ':', respectively, as the background character. These 2D key have entropy of 197 and 229 bits, respectively.

4.3.6 Styles of 2D Key: ASCII Art / Unicode Art

The third style of 2D key is ASCII art or Unicode art. ASCII art is a graphical presentation of computer using the 95 printable ASCII characters (Wikipedia Contributors, 2007g). Unicode is a variant of ASCII art, where instead of using ASCII characters, Unicode symbols are used to create artistic graphics.

```
111111
111111
--11--
--11--
--11--
111111
111111
```

```
222222
2D2DDD
2D2D2D
2D2D2D
2D2DDD
```

```
**||**
------
|****|
------
**||**
```

Figure 4.8 Styles of 2D key: ASCII art

Figure 4.8 displays three 2D key examples using ASCII art. For the left example, ASCII characters '1' and '-' are used to display a Chinese character meaning "engineering". Its dimensions are 7 x 6 with entropy of 275 bits. For the middle example, ASCII characters '2' and 'D' are used to display a digit '10' with background character '2'. Its dimensions are 5 x 6 with entropy of 197 bits. For the right example, ASCII characters '|' and '-' are used to display a Chinese character meaning "center" with background character '*'. Its dimensions are 5 x 6 with entropy of 197 bits.

Figure 4.9 shows a 2D key example using Unicode art. Unicode symbols '¥' and '©' are used to display a Chinese character meaning "engineering" again. Unicode '¥' is entered using the keyboard by pressing the keys "0165" while holding the key of 'Alt'. Unicode '©' is entered using the keyboard by pressing the keys

“0169” while holding the key of ‘Alt’. Once the ‘Alt’ key is released, the Unicode symbol is entered. Its dimensions are 4 x 5. This 2D key has entropy of 320 bits.

```
¥¥¥¥¥
©©¥©©
©©¥©©
¥¥¥¥¥
```

Figure 4.9 Styles of 2D key: Unicode art

4.3.7 Styles of 2D Key: Colourful Text

The style of this 2D key needs some additional supports. Color encoding, special graphical user interface, and special computer processing are required. Although these supports make the user interface complicated for the computer, they can be implemented and have better memorizability for the human users. Color is definitely a main element of good memorizability. For example, by having 16 types of colors, every character in the 2D key will have an additional 4 bits. ASCII-based 2D key will become 10.57 bits per character; whereas Unicode-based 2D key is 20.59 bits per character. The entropies per character of ASCII-based and Unicode-based 2D keys will be increased by 60.9% and 24.1%, respectively. The additional color secret also carries more randomness to resist dictionary attack.

4.3.8 Styles of 2D Key: Sensitive Input Sequence

For the secret style of sensitive input sequence, it is an additional feature over the current 2D secret style where there is added entropy from the input sequence of a character to a specific element location of the 2D matrix. If a 2D key has the dimensions of *m* x *n*, the key space is increased by [(*m* * *n*)!]. If a 2D key of 4 x 5 as in Figure 4.4 is used, the key space is increased by [20!] or 61.1 bits from 131.40 bits to 192.47 bits, which is close to the left example in Figure 4.5 for the 2D key of dimensions 5 x 6 with 197.10 bits.

This style requires the space encoding for the element location of 2D matrix, table-like graphical user interface of *m* x *n* matrix, and human memory for the sequence of characters. In term of memorizability, there is not much improvement. However, the time to enter a 2D key of similar size is greatly reduced for the same amount of entropy.

4.3.9 Applications for Symmetric and Asymmetric Key Cryptosystems

With the emergence of 2D key having the styles of mutliline passphrase, crossword, ASCII art / Unicode art, colorful text, and sensitive input sequence, high-entropy key as high as 256 bits is possible. Chinese-character-encoded passphrase can be efficiently used for the 2D key style of multiline passphrase. We can now overcome the human factor of memorizability and user interface problem of single-line key field, which have limited the key size to 96 bits.

Table 4.16 shows the possible dimensions of ASCII-based 2D key for various key sizes of symmetric key cryptosystem. Key strengthening can boost up another 19.5 bits. If Unicode-based 2D key is used, the dimensions of 2D key can be greatly reduced. From Table 3.3, the settings sufficiency of some key input methods and systems for various key sizes is shown. It can be observed that larger key sizes than 128 bits for cryptographic, information-hiding, and non-cryptographic applications like AES-128, AES-192, AES-256, ECC-256, etc., can be realized by using the 2D key, especially the MePKC using fully memorizable private key.

Table 4.16 Dimensions of 2D key for various symmetric key sizes

Symmetric key size (bits)	80	96	112	128	192	256
Number of ASCII characters	13	15	18	20	30	39
Dimensions of 2D key	3 x 5	3 x 5	3 x 6	4 x 5	5 x 6	5 x 8

For asymmetric key cryptosystem, memorizable public-key cryptosystem (MePKC) can be created. This is possible by using the FFC and ECC with minimum size of private key at 160 bits. The private key of MePKC is stored in the human brain, and not stored as encrypted, split, and roaming private keys as in the prior arts. This provides mobility, lower cost, higher efficiency, and resistance to dictionary and pre-computation attacks.

Assuming that the maximum memorizable key size is 256 bits, 256-bit MePKC using FFC and ECC with 128-bit security strength can be realized. It has a protection period of 30 years. If key strengthening is used, 19.5 bits is added, or an increase of 10-bit security, which extends the protection to 50 years. This is very much enough for many practical applications. For more information, please refer to Chapter 9.

A software prototype of this 2D key (Lee, 2006b, 2008i) with the function of multihash key (Lee, 2007a) has been built up by using the Microsoft Visual Studio (Marshall, 2003). The 2D key can have optional anti-keylogging application software (Log This, No date; McNamara, 2003, pp. 197-202) to achieve higher security during the input. To get a copy of this software, please visit [URL: www.xpreeli.com].

There are other potential applications of 2D key methods and systems. Firstly, 2D key can be specialized to include only numeric digits or other sets of limited encoded characters for devices with limited space like the display and key pad of a bank ATM machine and computerized safety box. Secondly, the display of 2D key can be an LCD display or other display technologies integrated with a computer keyboard having a first partial 2D key optionally visible and a second partial 1D key in hidden mode only to better resist the shoulder-surfing attacks.

4.3.10 Conclusion

Here, the high-entropy 2D key input method has been proposed. It solves the memorizability problem due to human factor and user interface problem of single-line key/password field. Chinese-character-encoded passphrase is efficient for the 2D key style of multiline passphrase. Besides, 2D key has the styles of crossword, ASCII art, Unicode art, colourful text, and sensitive input sequence. The memorizable limit of 96-bit key is increased to 256-bit key, where even the private key is memorizable. This creates 160-bit to 256-bit MePKC with protection period up to 50 years.

CHAPTER 5 CREATING BIG MEMORIZABLE SECRET (PART 3)

5.1 Multilingual Key

2D key is a method suitable for Latin language users. For users of CJK languages using the Han character and other non-Latin languages, multilingual key is an alternative to 2D key. For the trends, statistics, and geo-political coverages of monolingual, bilingual, and multilingual language users, please refer to Section 4.1.5.

5.1.1 Related Works

For the related works of multilingual key, please refer to Section 3.4 on potential methods to create big and yet memorizable secret. These related works are graphical password, Passfaces, Draw-a-Secret scheme, icon-like graphical password scheme, and event-based graphical password scheme.

5.1.2 Black-and-White Multilingual Key

For multilingual key, graphical password/key method and system is somehow innovated to have both the features of cognometrics and locimetrics by using graphic symbols of multilingual languages from any symbol encoding code, such as Unicode, specifically. This invention is especially effective for logographic, bilingual, and multilingual language users. In this new secret creation method, there is a huge key space comprising black-and-white and/or colorful Unicode graphic symbols grouped into tabular pages as in Figure 5.1 illustrating one of the exemplary tabular pages {4E00-4EFF}.

For this black-and-white multilingual key, a user knowing a particular language has the property of cognometrics to recognize a graphic symbol. Furthermore, there exists also the property of locimetrics for a user to locate a tabular page, subsequently a graphic symbol, and finally a partitioned area of a Unicode graphic symbol.

500

4E	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	一	丐	丠	丰	乀	乐	习	买	亀	亐	亠	亰	什	仐	仠	仰
1	丁	丑	両	丱	乁	乑	乡	乱	亁	云	亡	亱	仁	仑	仡	仱
2	丂	丒	丢	串	乂	乒	乢	乲	亂	互	亢	亲	仂	仒	仢	仲
3	七	专	丣	丳	乃	乓	乣	乳	亃	亓	亣	亳	仃	仓	代	仳
4	丄	且	两	临	乄	乔	乤	乴	亄	五	交	亴	仄	仔	令	仴
5	丅	丕	严	丵	久	乕	乥	乵	亅	井	亥	亵	仅	仕	以	仵
6	丆	世	並	丶	乆	乖	书	乶	了	亖	亦	亶	仆	他	仦	件
7	万	丗	丧	丷	乇	乗	乧	乷	亇	亗	产	亷	仇	仗	仧	价
8	丈	丘	丨	丸	么	乘	乨	乸	予	亘	亨	亸	仈	付	仨	仸
9	三	丙	丩	丹	义	乙	乩	乹	争	亙	亩	亹	仉	仙	仩	仹
A	上	业	个	为	乊	乚	乪	乺	亊	亚	亪	人	今	仚	仪	仺
B	下	丛	丫	主	之	乛	乫	乻	事	些	享	亻	介	仛	仫	任
C	丌	东	丬	丼	乌	乜	乬	乼	二	亜	京	亼	仌	仜	们	仼
D	不	丝	中	丽	乍	九	乭	乽	亍	亝	亭	亽	仍	仝	仭	份
E	与	丞	丮	举	乎	乞	乮	乾	于	亞	亮	亾	从	仞	仮	仾
F	丏	丟	丯	丿	乏	也	乯	乿	亏	亟	亯	亿	仏	仟	仯	仿

Figure 5.1 One of the exemplary tabular pages of multilingual key consisting of the first 256 Han characters in the Unicode and starting from Unicode value {4E00}

The input method of multilingual key is normally a computer mouse, where it can also be other input devices like touch screen, tablet, stylus, keyboard, sound recognition, eye-tracking technology, Microsoft Surface, etc. The monitor tend towards wide-screen LCD at lower cost shall popularize the multilingual key.

5.1.3 Grid Partitioning for Higher Entropy and Randomness

To increase the entropy per image selection and its randomness to resist guessing attack and dictionary attack, invisible grid partitioning is applied to every graphic symbol based on the setting of 3 * 3, particularly, or any other settings such as 2 * 2, 4 * 4, and so on, as in Figure 5.2.

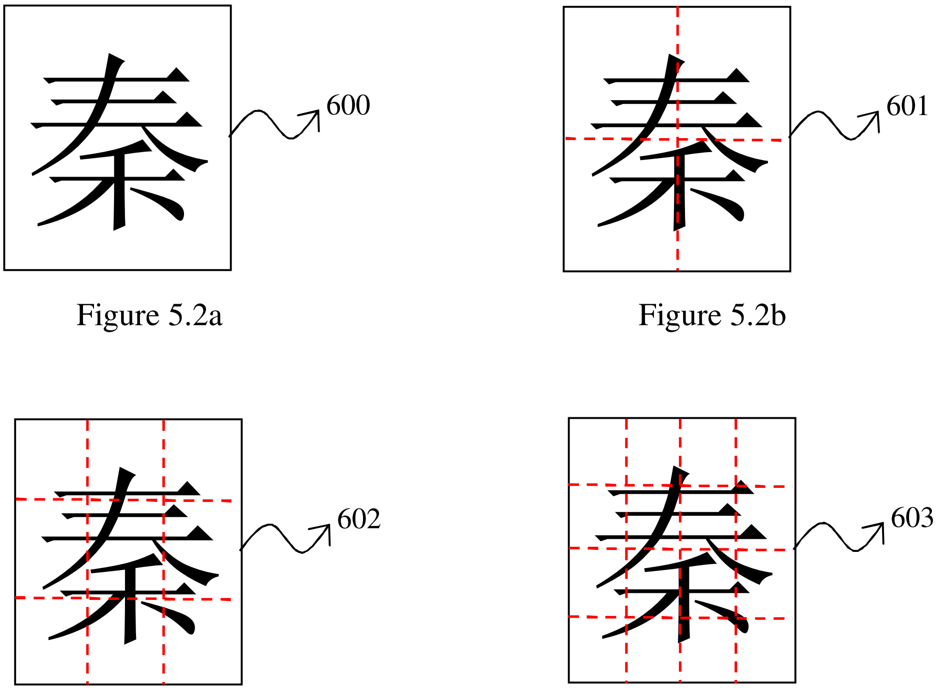


Figure 5.2a

Figure 5.2b

Figure 5.2c

Figure 5.2d

Figure 5.2 A Han character from Unicode before and after the grid partitioning for various settings: (Figure 5.2a) Without grid partitioning; (Figure 5.2b) with grid partitioning of 2 * 2; (Figure 5.2c) with grid partitioning of 3 * 3; and (Figure 5.2d) with grid partitioning of 4 * 4

These partitioned areas increases the entropy of multilingual key by 2, 3, and 4 bits, respectively, for 2 * 2, 3 * 3, and 4 * 4 settings. Every partitioned area represents the concatenation of a few bits to the bitstream encoding a graphic symbol using Unicode in a tabular page consisting of 256 symbols or flexibly any other amount. Among the settings of grid partitioning, 3 * 3 is selected as the optimum settings and used for further explanation.

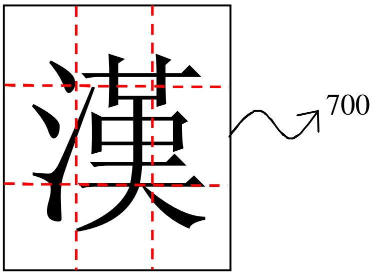
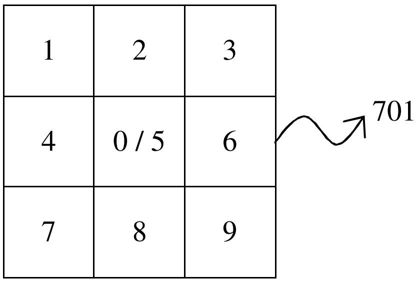
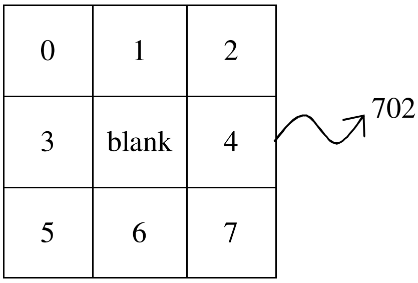


Figure 5.3a



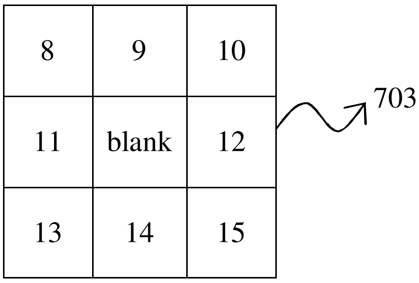
1	2	3
4	0 / 5	6
7	8	9

Figure 5.3b



0	1	2
3	blank	4
5	6	7

Figure 5.3c



8	9	10
11	blank	12
13	14	15

Figure 5.3d

Figure 5.3 Grid partitioning encoding of a graphic symbol, wherein (Figure 5.3a) illustrates the 3 * 3 settings where red lines are invisible; (Figure 5.3b) illustrates the encoding for human memorization and reference in the human context; (Figure 5.3c) illustrates the concatenated bit values to the Unicode value of a graphic symbol in the BMP (Basic Multilingual Plane) when a partitioned area is selected in the computer context; and (Figure 5.3d) illustrates the concatenated bit values to the Unicode value of a graphic symbol in the SIP (Supplementary Ideographic Plane) when a partitioned area is selected in the computer context

There are nine partitioned areas in the setting of 3 * 3 as in Figure 5.3a. The outer 8 partitioned areas are encoded by 3 bits. Meanwhile, the central partitioned area adds no bit. For Han characters and other multilingual languages, two Unicode planes are used in the multilingual key, where more Unicode planes can also be added. These are BMP (Basic Multilingual Plane) and SIP (Supplementary Ideographic Plane), where both can support 65536 (= 216) graphic symbols.

For computer context, graphic symbols from different Unicode planes are encoded by bit 0 for BMP and bit 1 for SIP; whereas the 9 partitioned areas have the central area to carry blank value, and the outer areas to represent bit values of 0, 1, 2, to 7 for BMP and 8, 9, 10, to 15 for SIP, as in Figures 5.3c and 5.3d, respectively. For human context, to ease memorization and references, the 3 * 3 partitioned areas are again encoded by digits from 0, 1, 2, to 9 as in Figure 5.3b. The central area represents digits 0 and 5; whereas the outer areas represent 1, 2, 3, 4, 6, 7, 8, and 9 for both graphic symbols from BMP and SIP. Hence, the 3 x 3 grid partitioning adds either 0 bit with one-fifth (1/5) probability, or 4 bits with four-fifth (4/5) probability, to the Unicode value of a selected graphic symbol.

For instance, for a Chinese language secret of [秦漢] (Qin Han), the code of multilingual key without grid partitioning is $\{79E66F22\}_{16}$ based on Unicode, where $\{79E6\}_{16}$ represents [秦] (Qin) and $\{6F22\}_{16}$ represents [漢] (Han). When 3 * 3 grid partitioning is used, two more digits of secret are added. Let the first digit to be $\{4\}_{10}$ to represent the western piece of partitioned areas of [秦] (Qin), and the second digit to be $\{5\}_{10}$ to represent the central piece of partitioned areas of [漢] (Han). Consequently, the constructed secret is [秦 4 漢 5] (Qin 4, Han 5).

Since both the Han characters of [秦漢] (Qin Han) are in the BMP, then the encoded secret for a computing device is $\{79E636F22\}_{16}$. The concatenated hexadecimal digit of $\{3\}_{16}$ to the end of the Unicode value of $\{79E6\}_{16}$ is constructed from $\{0011\}_2$ where the first bit represents the BMP and the last three bits represent the western piece of partitioned areas. For the numeric secret of $\{5\}_{10}$, no hexadecimal digit is added because digits $\{0\}_{10}$ and $\{5\}_{10}$ represent no concatenated value to the Unicode value of selected graphic symbol. The concatenation of these numeric secrets representing different partitioned areas can be at any location of the Unicode values of the selected graphic symbols.

Therefore, for black-and-white multilingual key with 3 * 3 grid partitioning, a selected image by clicking a partitioned area carries 16.59 or 20.59 bits, with probabilities of 1/5 and 4/5, respectively. For a sequence of many selected partitioned image areas, the average entropy per image selection for this type of multilingual key is 19.79 bits.

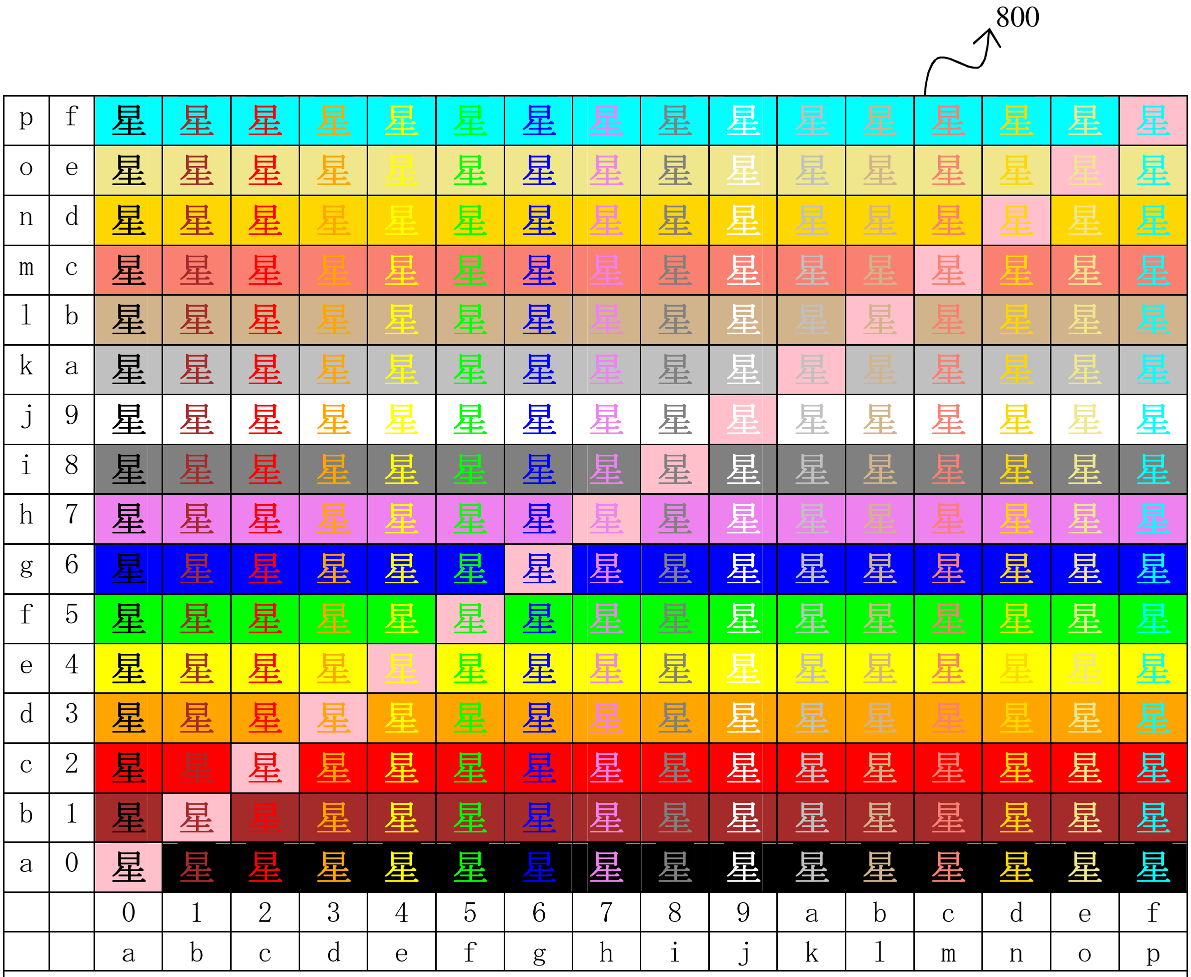
5.1.4 Colourful Multilingual Key

To further increase the key space for higher entropy, colourful multilingual key is an added option. The (16+1)-colour scheme of colourful multilingual key as in Figure 5.4 is selected for explanation, where it can also be other settings. The (2+1)-, (4+1)-, (8+1)-, and (16+1)-colour schemes of colourful multilingual key additionally add 2, 4, 6, and 8 bits, respectively, to the black-and-white multilingual key with 3 * 3 grid partitioning. This means that a selected partitioned image area of (16+1)-colour multilingual key has 24.59 or 28.59 bits and average entropy of 27.79 bits. Also, besides Unicode character and partitioning digit, a user needs to remember a third secret for the combination of foreground and background colours.

Yet to further increase the key space, some special text processing techniques can be used, wherein examples include special effects like directional shadow, 3D styles, and lighting; enclosed character using shapes like circle, square, triangular, or diamond; typeface variation like font type, font size, as well as font format of single strike through, double strike through, and underscore/underline; mirror images on the left, right, up/down; 45^o-, 90^o-, and 135^o-degree clockwise and anti-clockwise rotated images; solid and hollow images; and background watermark.

5.1.5 Font Technologies to Solve Data Size Problem

Nevertheless, the potential huge key space of colourful multilingual key with and without special text processing techniques has memory storage problem due to its huge image size if tabular pages of graphic symbols are stored in normal image file format like BMP, GIF, JPG, and PNG. For black-and-white multilingual key, its problem is not the image storage, but the image loading to the limited RAM, which is also a second problem to the colourful multilingual key.


800
p f
o e
n d
m c
l b
k a
j 9
i 8
h 7
g 6
f 5
e 4
d 3
c 2
b 1
a 0
星
0 1 2 3 4 5 6 7 8 9 a b c d e f
a b c d e f g h i j k l m n o p

Using X11 Color names or any other color encoding:

Black #000000	Brown #A52A2A	Red #FF0000	Orange #FFA500
Yellow #FFFF00	Green #008000	Blue #0000FF	Violet #EE82EE
Gray #808080	White #FFFFFF	Silver #C0C0C0	Tan #D2B48C
Salmon #FA8072	Gold #FFD700	Khaki #F0E68C	Cyan #00FFFF
Pink #FFC0CB			

N.B. (Nota Bene): The first 10 encoded colours from 0 to 9 are according to the resistor colour code. Other extended digits from 10 to 15 are the lighter colours from black to green, respectively modulus 10, and the last colour pink is used as the front-slash-wise diagonal background colour.

Figure 5.4 (16+1)-colour scheme for colourful multilingual key

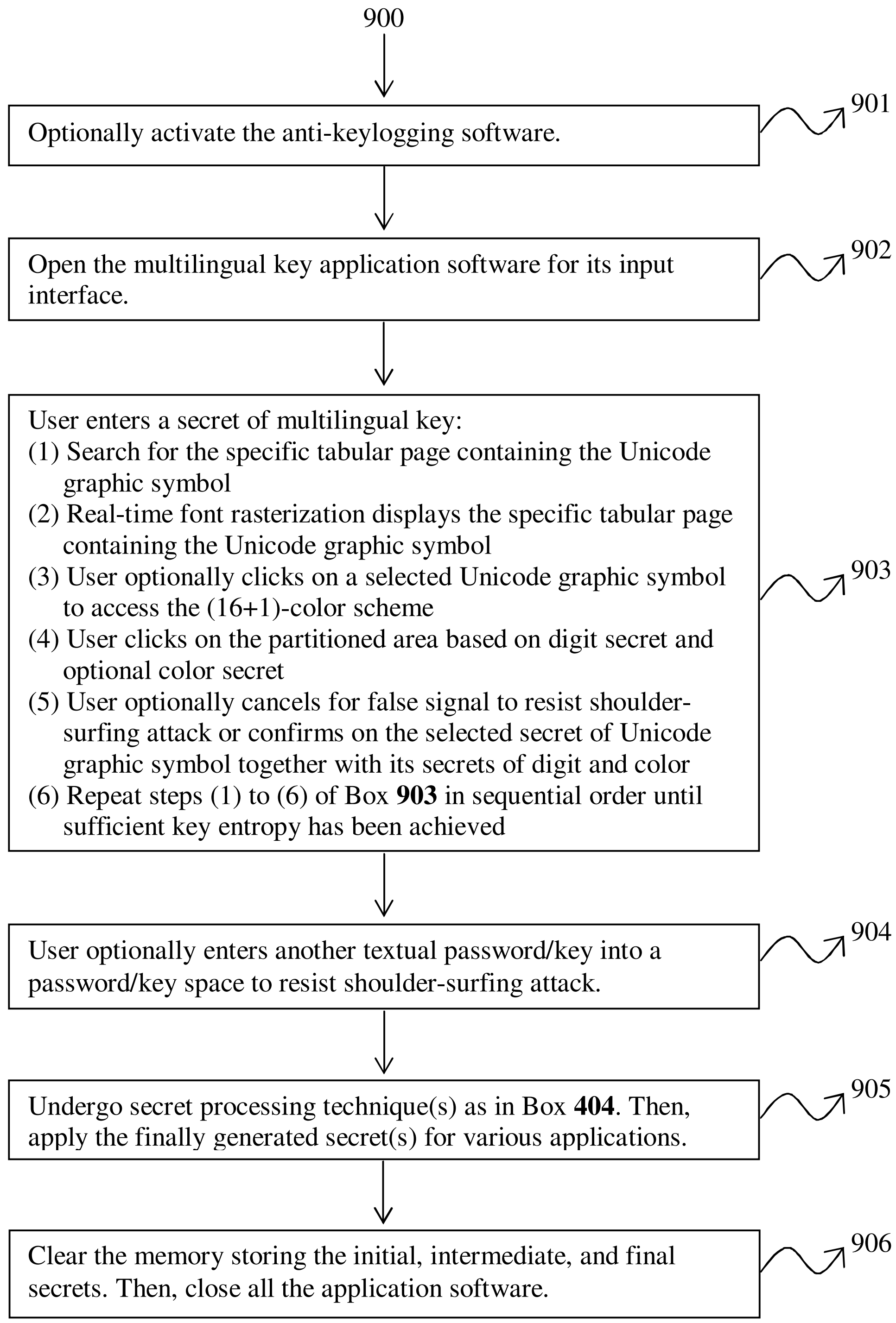
To solve the limited RAM problem of black-and-white and colourful multilingual keys, the image file format of PNG (Portable Network Graphics), which is good for image compression of line art, can be used for efficient size of image database. Yet for better file compression, algorithm of DJVU file format can be further applied by splitting a tabular page into many layers for separate compression. However, the best current possible and practical solution to both the problems is to have real-time font rasterization from font files like outline font or vector font storing all the Unicode graphic symbols to the monitor display.

5.1.6 Resistance Techniques to Shoulder-Surfing Attack

Another problem of multilingual key is shoulder-surfing attack from a person or camera nearby the monitor and able to watch and record the image area selection of sequential Unicode graphic symbols. The first solution relies on the human memorizability limit and asks a user to do false selection of image areas by toggling a key on the keyboard, or single-double or left-middle-right clicking of mouse. The second solution is to allow a user to enter a textual password into a key field at any interim session during the input of a graphical key. In other words, the second solution is a hybrid method combining the textual and graphical passwords/keys.

5.1.7 Fast Search for a Unicode Graphic Symbol

Yet another problem of multilingual key is its huge key space causes the search of a graphic symbol to be slow if only images of Unicode graphic symbols are stored. To solve this problem, there can be some tabular pages specially designed to list and show the frequently used Unicode graphic symbols, especially Latin and Han characters, or Latin and other languages, to speed up the image area selection of a Unicode graphic symbol. A second solution is to have a fast input method and system of Unicode graphic symbol to search and locate the tabular page and specific location of a particular graphic symbol, which is now possible for Latin languages and CJKV languages using Han characters.



Optionally activate the anti-keylogging software.

Open the multilingual key application software for its input interface.

User enters a secret of multilingual key:

(1) Search for the specific tabular page containing the Unicode graphic symbol
(2) Real-time font rasterization displays the specific tabular page containing the Unicode graphic symbol
(3) User optionally clicks on a selected Unicode graphic symbol to access the (16+1)-color scheme
(4) User clicks on the partitioned area based on digit secret and optional color secret
(5) User optionally cancels for false signal to resist shoulder-surfing attack or confirms on the selected secret of Unicode graphic symbol together with its secrets of digit and color
(6) Repeat steps (1) to (6) of Box **903** in sequential order until sufficient key entropy has been achieved

User optionally enters another textual password/key into a password/key space to resist shoulder-surfing attack.

Undergo secret processing technique(s) as in Box **404**. Then, apply the finally generated secret(s) for various applications.

Clear the memory storing the initial, intermediate, and final secrets. Then, close all the application software.

Figure 5.5 Operation of multilingual key input method and system

5.1.8 Applications for Symmetric and Asymmetric Key Cryptosystems

Subsequently, big memorizable secret for cryptographic, information-hiding, and non-cryptographic applications in information engineering can be created from multilingual key as in Figure 5.5 according to the specific demand thresholds for various key sizes in Table 3.3. More importantly, MePKC using fully memorizable private key can be specifically realized. For more information, please refer to Chapter 9.

Multilingual key input can have optional anti-keylogging application software (Log This, No date; McNamara, 2003) to achieve higher security. The input method of multilingual key is normally a mouse, where it can also be other input devices like touch screen, tablet, stylus, keyboard, sound recognition, eye-tracking technology, Microsoft Surface, etc. The key space is increased using pictorial colourful Unicode graphic symbols with 17 background colours and 16 foreground colours, which can also be increased using special effects like directional shadow, 3D styles, lighting, enclosed character using shapes like circle, square, triangular, or diamond, as well as typeface variation like font type, font size, and font format.

The (16+1) colours of colourful multilingual key in Figure 5.4 are black, brown, red, orange, yellow, green, blue, violet, gray, white, silver, tan, salmon, gold, khaki, and cyan for 16 foreground colours, and black, brown, red, orange, yellow, green, blue, violet, gray, white, silver, tan, salmon, gold, khaki, cyan, and pink for 17 background colours. The first 10 colours of the (16+1)-colour scheme has good memorizability based on the colour code of resistor. The next 6 colours are lighter colours than the corresponding colours modulus 10. The last colour pink is used as the front-slash-wise diagonal background colour. Other colour combinations may also be possible.

A software prototype of this multilingual key (Lee & Tan, 2006a) has been built up by using the Microsoft Visual Studio (Marshall, 2003). To get a copy of this software, please email the author at [Email: E96LKW@hotmail.com]. Whenever the conditions like time, money, and online archival service allow, the author shall mail you a CD copy or upload the software.

5.2 Multi-Tier Geo-Image Key

To create big memorizable secret, a second new type of graphical password/key is invented using a hybrid combination of recognition-based cognometrics and locimetrics over a map, as well as recall-based textual password/key of a space name and characteristics. This space map can be continents of Earth, seafloor of oceans, constellations of star sky, and so on.

Let's take the Earth map of continents as an example for multi-tier geo-image key. The current best GPS (Global Positioning System) resolution for civilian usages is about 15 meters (m) per pixel. For the latest Google super-satellite GeoEye-1 (Chen, 2008; Wikipedia Contributor, 2008ay) launched on 6 September 2008, it captured the first image at a resolution of 41 cm on 7 October 2008 (EDT). For spy satellite (Wikipedia Contributors, 2008ax), the best resolution may be less than 2 cm.

Let's take 15 meters (m) per pixel in the calculation. The radius of Earth globe is $r = 6.37 \text{ x } 10^6$ m and its surface area is $S_{Earth} = 4\pi r^2 = 5.099 \text{ x } 10^{14} \text{ m}^2$. Assume only 2^{-7} of Earth surface is memorizable populated areas like metropolis, city, town, village, etc. Assume also a pixel represents an area of 15^2 m^2, and a partitioned area of Earth map at the first tier has 20 * 20 pixels. At a monitor image resolution of 800 * 600 pixels, there are 1200 partitioned areas at the first tier of Earth map. Simple estimation will show that four to five tiers of map are needed to locate a specific location on the Earth surface after subsequently selected image areas.

Through some calculation, the whole Earth surface including continents and oceans has a surface area per pixel of $S_{pixel} = 4\pi r^2 / 15^2 = 2.266 \text{ x } 10^{12} \text{ m}^2$/pixel, or entropy of $E_{Earth} = 41.04$ bits. Considering a click area of 20 x 20 pixels after image partitioning, the surface area per click area is $S_{click} = 4\pi r^2 / (15^2 \text{ x } 20^2) = 5.665 \text{ x } 10^9$ m^2/click area, or entropy of 32.40 bits. When the factor of easily memorizable Earth space like populated areas is included, the usable Earth surface to create a big memorizable secret is $S_{memorizable} = 2^{-7} \text{ x } S_{click} = 4.426 \text{ x } 10^7 \text{ m}^2$/click area, or entropy of 25.40 bits. Hence, a partial image secret of multi-tier geo-image key has about 25.40 bits.

In addition to a partial image secret of a space, a user is also required to enter a second partial textual secret related to the name and/or characteristics of that particular selected image space or location. This is used to increase the key entropy and to resist the shoulder-surfing attack. For every partial image secret, there shall be a partial textual secret. Preferably, the key length of the partial textual secret is at least 6 characters.

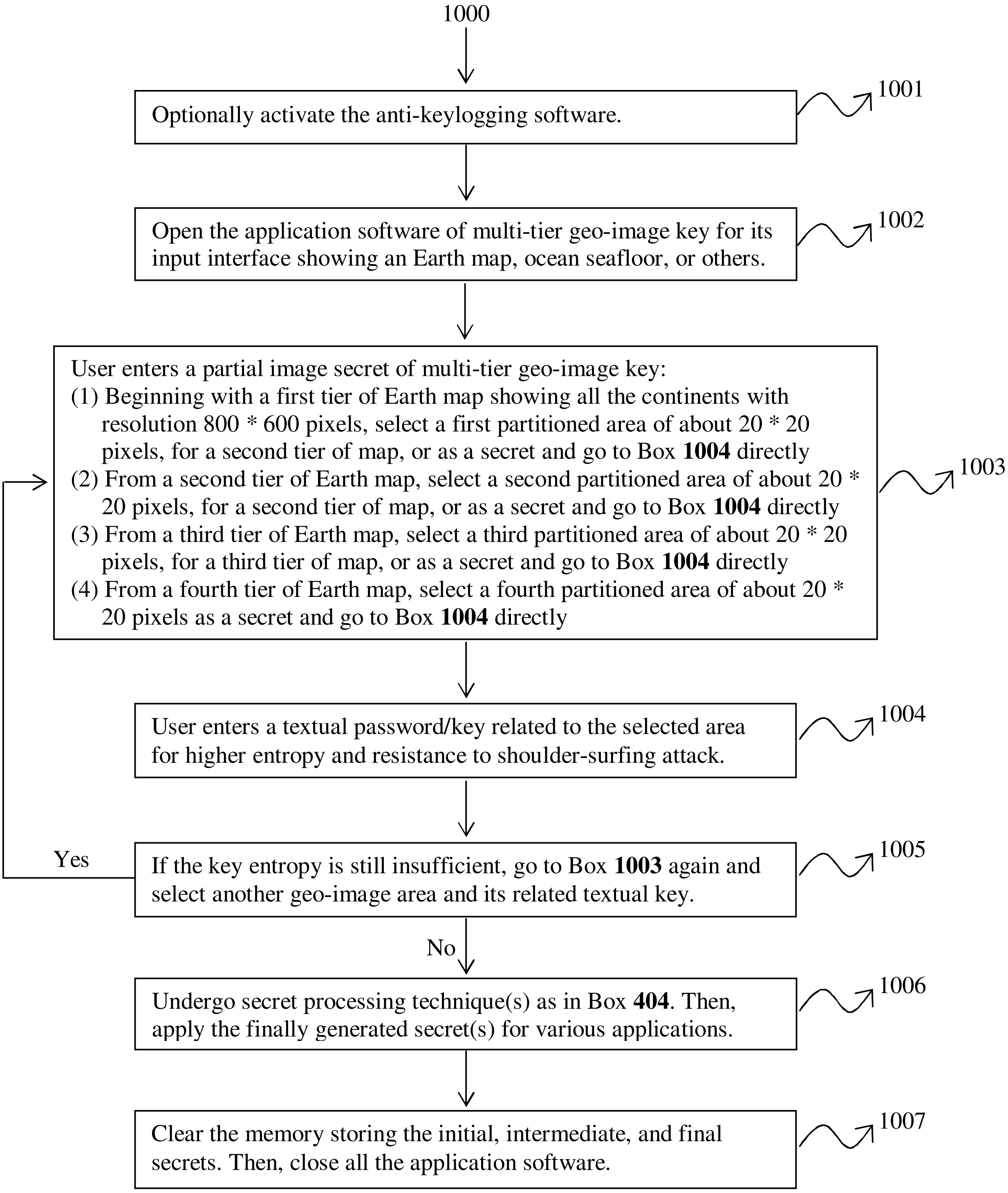

1000
Optionally activate the anti-keylogging software.
1001
Open the application software of multi-tier geo-image key for its input interface showing an Earth map, ocean seafloor, or others.
1002
User enters a partial image secret of multi-tier geo-image key:
(1) Beginning with a first tier of Earth map showing all the continents with resolution 800 * 600 pixels, select a first partitioned area of about 20 * 20 pixels, for a second tier of map, or as a secret and go to Box 1004 directly
(2) From a second tier of Earth map, select a second partitioned area of about 20 * 20 pixels, for a second tier of map, or as a secret and go to Box 1004 directly
(3) From a third tier of Earth map, select a third partitioned area of about 20 * 20 pixels, for a third tier of map, or as a secret and go to Box 1004 directly
(4) From a fourth tier of Earth map, select a fourth partitioned area of about 20 * 20 pixels as a secret and go to Box 1004 directly
1003
User enters a textual password/key related to the selected area for higher entropy and resistance to shoulder-surfing attack.
1004
Yes
If the key entropy is still insufficient, go to Box 1003 again and select another geo-image area and its related textual key.
1005
No
Undergo secret processing technique(s) as in Box 404. Then, apply the finally generated secret(s) for various applications.
1006
Clear the memory storing the initial, intermediate, and final secrets. Then, close all the application software.
1007

Figure 5.6 Operation of multi-tier geo-image key input method and system

If ASCII encoding is used, then the textual password/key adds another 39.42 bits. In total, a unit of multi-tier geo-image key has entropy of 64.82 bits. Some units of multi-tier geo-image key are sufficient for many applications using secret. To specifically realize the MePKC, three and four units of multi-tier geo-image key can support 160- and 256-bit MePKC, respectively, using the ECC. The monitor tend towards wide-screen LCD at lower cost shall popularize the multi-tier geo-image key as well.

Table 3.3 shows the required unit of geo-image key for various key sizes, and Figure 5.6 illustrates the operation of this method. The space map can optionally have invisible and/or visible grid lines for easy references. The input method is normally a mouse, where it can also be other input devices like touch screen, stylus, keyboard, sound recognition, eye-tracking technology, Microsoft Surface, etc. To further increase the key space of this method, the preceding tiers of geo-image key before the last tier can be included, and early secret selection of larger geographical area is allowed. Multi-tier geo-image key input can also have optional anti-keylogging (Log This, No date) application software to achieve higher security.

To further increase the key space of this method, the preceding tiers of geo-image key before the last tier can be included, and early secret selection of larger geographical area is allowed. Yet another method to increase the key space is to invest more resources to recruit the architects to draw the geographical map of populated areas using the architectural normal scaling of 1:500 (or 1 cm : 500 cm, or 1 cm : 5 m), which is a resolution better than the civilian GPS resolution 15 m/pixel.

5.3 Multi-Factor Multimedia Key Using Software Token

To create big memorizable secret, especially for MePKC realization in Chapter 9, the key sizes larger than 256 bits, such like 384 and 512 bits, are hard to be memorizable, and a possible solution is multi-factor multimedia key using software token as in Figures 5.7-5.8. For instance, 512-bit MePKC using ECC is needed to realize the bits of security at 256 bits and to resist future quantum computer attack. Hence, multi-factor multimedia key using software token is invented to halve the memorizable key sizes at equivalent security levels, especially designed for MePKC operating on the FFC or ECC.

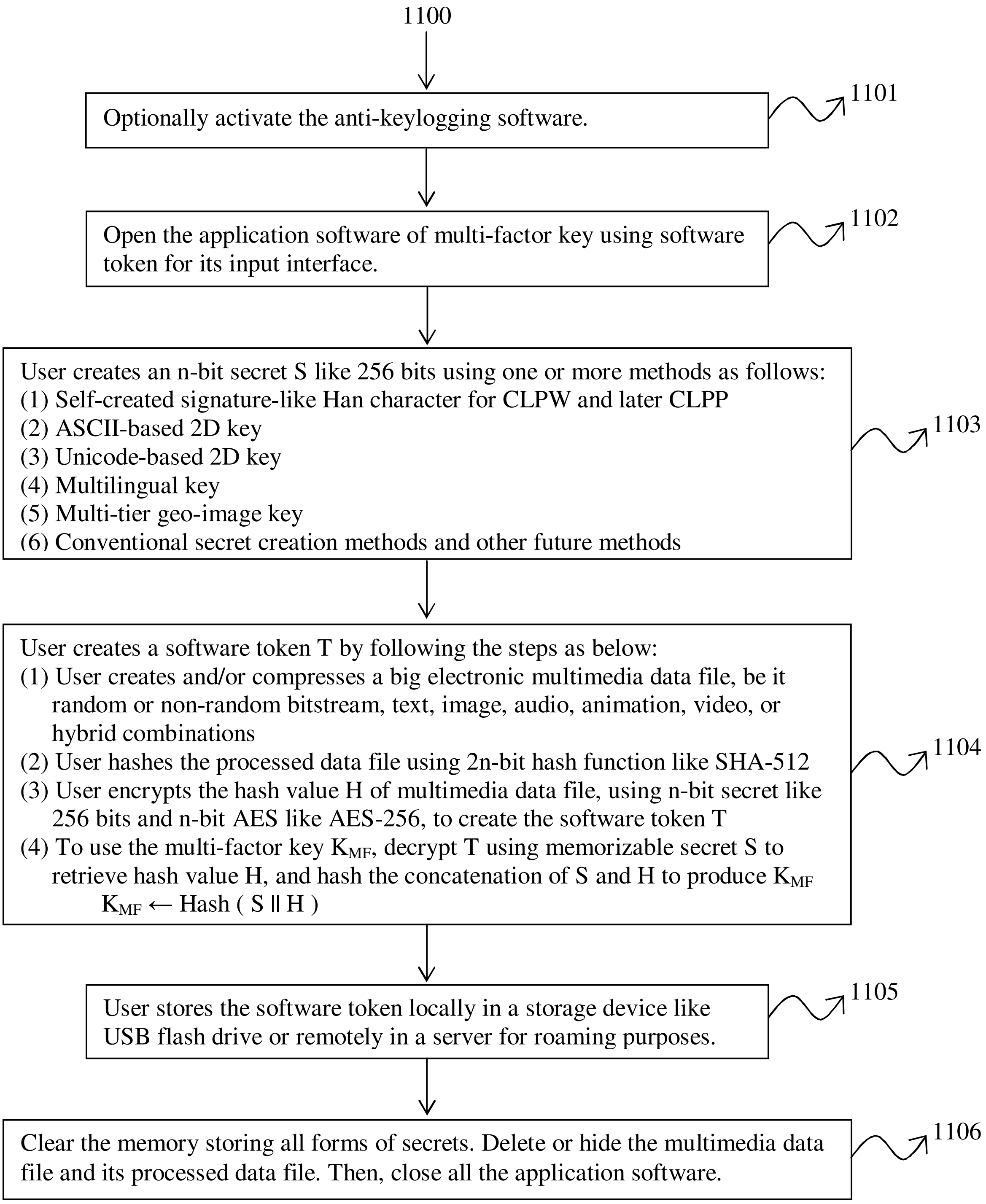

1100
Optionally activate the anti-keylogging software.
1101
Open the application software of multi-factor key using software token for its input interface.
1102
User creates an n-bit secret S like 256 bits using one or more methods as follows:
(1) Self-created signature-like Han character for CLPW and later CLPP
(2) ASCII-based 2D key
(3) Unicode-based 2D key
(4) Multilingual key
(5) Multi-tier geo-image key
(6) Conventional secret creation methods and other future methods
1103
User creates a software token T by following the steps as below:
(1) User creates and/or compresses a big electronic multimedia data file, be it random or non-random bitstream, text, image, audio, animation, video, or hybrid combinations
(2) User hashes the processed data file using 2n-bit hash function like SHA-512
(3) User encrypts the hash value H of multimedia data file, using n-bit secret like 256 bits and n-bit AES like AES-256, to create the software token T
(4) To use the multi-factor key K_{MF}, decrypt T using memorizable secret S to retrieve hash value H, and hash the concatenation of S and H to produce K_{MF}
$K_{MF} \leftarrow \text{Hash} (S \| H)$
1104
User stores the software token locally in a storage device like USB flash drive or remotely in a server for roaming purposes.
1105
Clear the memory storing all forms of secrets. Delete or hide the multimedia data file and its processed data file. Then, close all the application software.
1106

Figure 5.7 Software token generation of multi-factor multimedia key input method and system

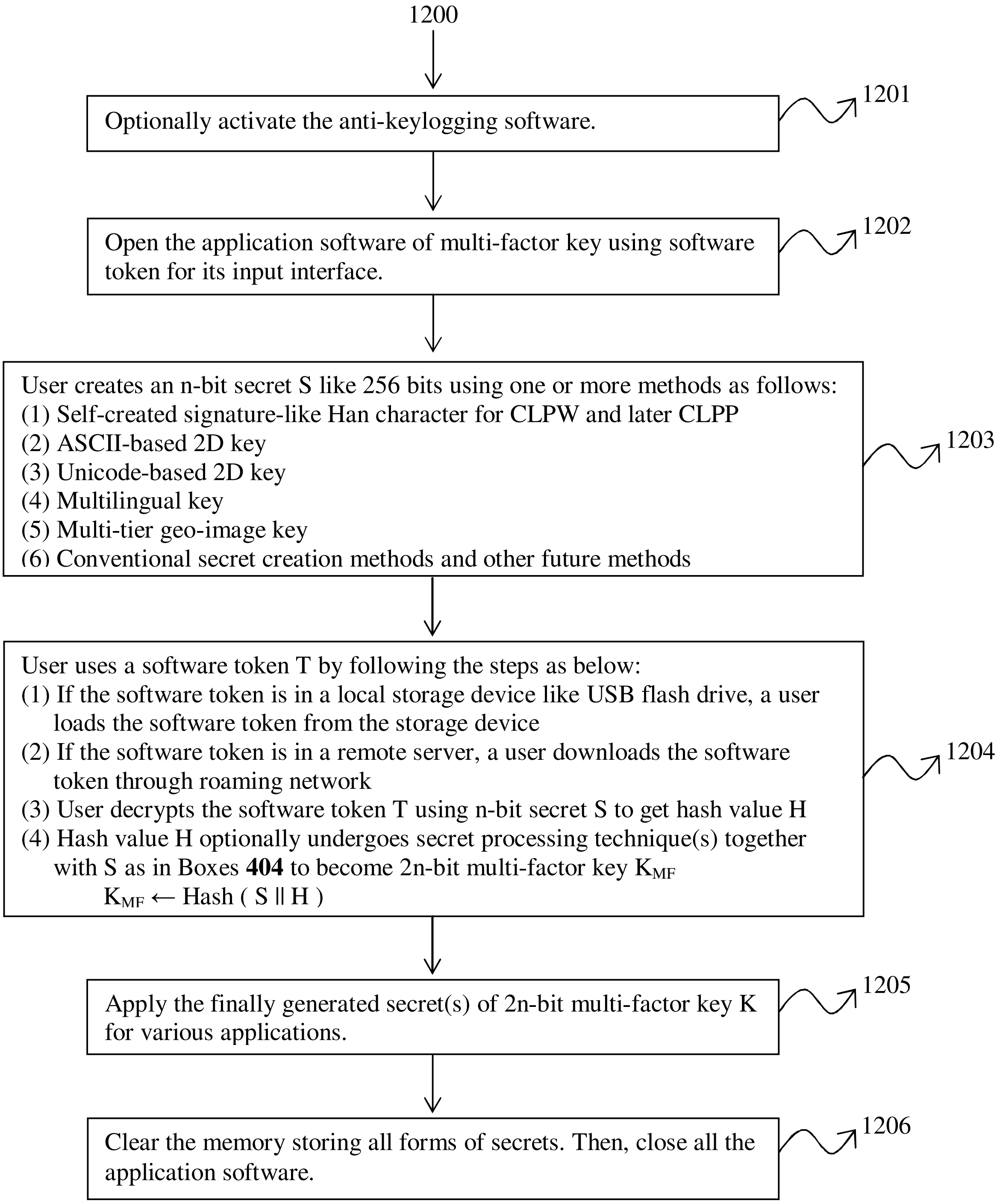

1200
Optionally activate the anti-keylogging software.
1201
Open the application software of multi-factor key using software token for its input interface.
1202
User creates an n-bit secret S like 256 bits using one or more methods as follows:
(1) Self-created signature-like Han character for CLPW and later CLPP
(2) ASCII-based 2D key
(3) Unicode-based 2D key
(4) Multilingual key
(5) Multi-tier geo-image key
(6) Conventional secret creation methods and other future methods
1203
User uses a software token T by following the steps as below:
(1) If the software token is in a local storage device like USB flash drive, a user loads the software token from the storage device
(2) If the software token is in a remote server, a user downloads the software token through roaming network
(3) User decrypts the software token T using n-bit secret S to get hash value H
(4) Hash value H optionally undergoes secret processing technique(s) together with S as in Boxes **404** to become 2n-bit multi-factor key K_{MF}
$K_{MF} \leftarrow$ Hash (S || H)
1204
Apply the finally generated secret(s) of 2n-bit multi-factor key K for various applications.
1205
Clear the memory storing all forms of secrets. Then, close all the application software.
1206

Figure 5.8 Software token acquisition and application of multi-factor multimedia key input method and system

5.3.1 Related Work

A closest related work to multi-factor multimedia key has been published and found after the patent filing of multi-factor multimedia key. This related work is multimedia password (Mannan & van Oorschot, 2008) using hash value of a digital object as password.

5.3.2 Using Key the Secret and USB Flash Drive

For 2n-bit ECC, where 2n can be as big as 512, its 2n-bit private key can be derived from a memorizable secret and a 2n-bit hash value. This 2n-bit hash value is obtained from the hashing of a big multimedia data file with its size at least 512 bits by 2n-bit hash function like SHA-512. This multimedia data file may be random or non-random bitstream, text, image, audio, animation, video, or hybrid combinations. The 2n-bit hash value is encrypted by an n-bit memorizable symmetric key using n-bit AES like AES-256 to create a software token. Here, 2n-bit ECC and n-bit AES have equivalent bits of security strength at n bits in the scale of symmetric key cryptosystem. This software token is then stored in a local storage device like USB flash drive, floppy disk, CD-ROM, DVD, etc., or in a remote server.

5.3.3 MePKC Application with Optional Split Key Method

Whenever a user needs to use the 2n-bit MePKC like 2n-bit ECC, one is either to get the local device storing the software token or to download it from a server through roaming network. Then, by using n-bit memorizable symmetric key S, one decrypts the 2n-bit software token to get 2n-bit hash value, which is later used together with S to derive the 2n-bit private key of 2n-bit MePKC. Hence, this bi-factor key using an n-bit symmetric key and 2n-bit software token can halve the key sizes of MePKC by sacrificing some mobility.

This method can be extended to become multi-factor key easily by undergoing the similar processes in split private key cryptography (Ganesan, 1996b). For instance, the software token may require bi-factor or multi-factor authentication, including at least a biometrics factor to access the software token. Also, during the input, multi-factor multimedia key using software token can have optional anti-keylogging application software (Log This, No date) to achieve higher security.

5.3.4 Software Prototype

A software prototype of this multi-factor multimedia key using software token (Lee, 2008k) with the function of multihash key (Lee, 2007a) has been built up by using the Microsoft Visual Studio (Marshall, 2003). The multimedia key input has optional anti-keylogging application software (Log This, No date) to achieve higher security. To get a copy of this software, please visit [URL: www.xpreeli.com].

5.4 Hybrid Combinations

So far, there are five methods discussed independently and slightly linked on how to create big and yet memorizable secret. These five methods are (i) Chinese-character-encoded passphrase, (ii) 2-dimensional (2D) key, (iii) multilingual key, (iv) multi-tier geo-image key, and (v) multi-factor multimedia key using software token. Here, it is to note that these five methods can be applied in hybrid mode together with the optional keys of password(s) and passphrase(s).

From psychological studies (Standing, Conezio & Haber, 1970; Standing, 1973; Jansen, Gavrila, Korolev, Ayers & Swanstrom, 2003; de Angeli, Coventry, Johnson & Renaud, 2005), graphical secret key has better memorizability. However, there are special attacks on graphical password/key (Thorpe & van Oorschot, 2007).

Nevertheless, there is a human memorizability limit on the number of unique secrets that a human can have. Adams and Sasse (1999) reported a limit of 4 to 5 keys that were unrelated and regularly used. Meanwhile, Florencio and Herley (2007) reported a limit at an average of 6.5 textual keys without any relevancy condition on the selection of a key, except that the set of keys of a user has to be different.

Coming to here, there is a problem on how to increase the number of unique secrets from a few master keys to some slave keys for both the offline and online accounts. Please refer to Chapter 7 for a proposed solution called multihash key.

CHAPTER 6 MULTIMEDIA NOISES FOR MORE RANDOM SECRET

6.1 Introduction

This chapter talks on the various types of multimedia noises or errors, which can be used to create more random secret. Higher randomness means higher complexity and harder secret cracking. Then, an enhanced frequency analysis to decrypt the English text is presented. It is followed by the discussion of information rates and unicity distances for some big secret creation methods. Lastly, a proof is shown on why secret with noise(s) has higher information rate and unicity distance.

6.2 Semantic Error Occurrences in the Multimedia Communications

Since the introduction of communications technologies from analogue communications to digital communications, the communication networks have evolved into the ubiquitous network of Internet that triggers the Information Era. The data transmission covers all types of media from bit stream, text, image, audio, to video. To increase the accuracy rate of the communications, the processes of error detection and error correction are carried out over the received data, which is majority-wise in the binary form for nowadays communications.

Nevertheless, in this chapter, the second error level occurring in the multimedia communications is discussed. The first error level is the bit stream; whereas the second error level is an advanced error called as semantic error (Lee, 2005a). The semantic error occurrence is explained corresponding to the type of media. Considering the semantic error occurrence, a protection layer can be obtained in mining the information from the multimedia communications. It is not only applicable to the digital media in the Silicon World, but the daily physical media as well. Hence, we can avoid the critical errors in addition to the higher accuracy rate during the intake process of information. It is highly useful in the field of science, commerce, history, archaeology, etc. Specifically here, the semantic errors are applied to add more randomness to the big secret.

6.2.1 Introduction

Mass, energy, and information are the three basic elements of the universe. Meanwhile, the communications of information in the human society is one of the two main components in the process of human civilization from Stone Age to the Modern Age. These two components are the languages to communicate the information and the tools to generate the functions. In the nineteen century, the first electronic communications technology of telegraph was initialized in year 1837 (Tomasi, 1998; IEEE Communications Society, 2002). Then the analog communications evolve to the digital communications. A good example of the ubiquitous network will be the Internet that triggers the Information Era.

Nowadays, the data communications covers all types of media from bit stream, text, image, audio, to video. The majority of the received data is in the binary form via the computing devices. Both of the processes of error detection and error correction are carried out over the received data to improve the accuracy rate of the communications. Both of these processes have now been very well improved to handle the first error level of the multimedia errors (Wu, Cheng & Xiong, 2001; Lu, 2002), which is the bit stream.

The works on the first error level of bit stream have been quite abundant and very successful for the data transmission in the secure communication systems (Schneier, 1996, 2000; Stallings, 2006b, 2007) via the applications of many types of codes and protocols.

Nevertheless, the focus of this section is to discuss on the second error level occurring in the multimedia communications. This second error level is an advanced error called semantic error. The capability to detect semantic error will be somehow able to differentiate among the neutral, true, and false statements. The semantic error occurrence is explained corresponding to various types of media. With the consideration of semantic error occurrence, a protection layer is obtained to mine the information from the multimedia communications.

It is not only applicable to the digital media in the Silicon World, but the daily physical media as well. Hence, critical errors can be avoided. In addition, higher accuracy rate during the intake process of information can be achieved. It is highly useful in the fields of science, commerce, history, archaeology, etc. By mining the critical information of the essential constraints, various models in these fields can be set up by using the practical intelligence. Specifically here, the semantic errors are applied to add more randomness to the big secret.

6.2.2 Multimedia Type of Bit Stream

For the multimedia type of bit stream, it is not common in the human society, except the I Ching (aka Yi Jing) （易经）, until the advent of computer technologies in the 20th century. There are basically three groups of them in categorizing the digital binary bit stream (B.1-B.3):

(B.1) Self-executable execution file

(B.2) Non-self-executable execution file

(B.3) Non-executable data file

The examples of (B.1) and (B.2) are the installation and execution files like self-extractable and non-self-extractable WinZip files. For (B.3), the example is like the Microsoft Office Word file with the file extension of .doc.

The advancement of error detection codes, error correction codes, and cryptology (Schneier, 1996, 2000; Stallings, 2006b, 2007) have allowed the full recovery of binary bit stream without considering the intrinsic information of the transferred data via the secure communication systems. Therefore, the consideration of first error level is normally sufficient for this type of multimedia. The demand for second error level of semantic error for advanced error checking is normally not an essential issue.

6.2.3 Multimedia Type of Text

For the multimedia type of text, it can be of monolingual and multilingual articles. Anyway, one language system represents one knowledge system or hyperspace knowledge library. Hence to learn and understand a civilization and culture comprehensively, the text version has to be learnt in the original language. For translated version into other language, the factors of polysemy and multimedia paradox will make the original version and translated version to have various disputable issues in semantics. Further categorization has two main groups (T.1-T.2):

(T.1) Hard copy

(T.2) Digital copy

Hard copy may be of handwriting and printed copies. Meanwhile the digital copy is due to the advent of computer technologies. Nevertheless, first error level checking is not enough for the both of them, especially for the text group of hard copy. As an information, first error level checking here means to detect and to correct the text in the style of one word after another word by treating the word as an independent symbol as for the binary bit stream for its block data. Nevertheless, it is not enough due to the associative factors of languages (Crystal, 1999; Finegan, 2004) and linguistics (Matthews, 1997) when the context of text is taken into consideration.

Subsequently, it is essential for the second error level checking to be implemented on the multimedia type of text. Considering the possible semantic errors of text (T.a-T.j) as listed below, it is very important to include them in mining the information from the multimedia type of text.

(T.a) Readability due to handwriting and printing qualities

(T.b) Copying or duplication process

(T.c) Translation process: Phonetics-, symbolics-, semantics-based

(T.d) Punctuation marks

(T.e) Erosion and dirt that change parts of the articles

(T.f) Book typesetting

(T.g) Book binding

(T.h) Usage of ancient, intermediate, and modern language

(T.i) Missing supporting references

(T.j) Polysemy where a word has two or more meanings

The significant examples of readability due to handwriting (T.a) are as follows:

(T.a.1) “Dao De Jing”（道德经）(Book of Ethics) and “Yi Jing”（易经）(Book of Changes) in ancient Chinese language writing that is being transformed into the modern Chinese language writing

The significant examples of translation process (T.c) are as follows:

(T.c.1) Books of “Greek Mythology”, “Homer’s Epics”, “Al-Quran” (Tareq Rajab Museum, 1998-2002), and “DunHuang Story” in the oral form of human brain memory and human sound

(T.c.2) Books of “The Holy Bible: New Testament” (“The Holy Bible,” 1997) from English language, “The Book of God: Old Testament” (Wangerin, 1999a) from Hebrew language, and “The Book of God: New Testament” (Wangerin, 1999b) from Greek language

(T.c.3) Book of “Diamond Sutra” in the translated version of Chinese language from the Sanskrit language

(T.c.4) Books of “Rig Veda”, “Puranas”, “Upanishad” (Xu, 1984), “Mahabharata” (Buck, 2000a), and “Ramayana” (Buck, 2000b) in the English and Chinese languages translated from the Sanskrit language

(T.c.5) Books of “Arabian Nights”, “Kama Sutra”, and “The Perfumed Garden” in the translated version of English language from the Arabian language

A good example is given as follows to show the effect of punctuation mark (T.d) via a series of sentences:

(T.d.1) Woman without her man is a savage.

(T.d.2) Woman without her, man is a savage.

(T.d.3) Woman without her man, is a savage.

(T.d.4) Woman, without her, man is a savage.

(T.d.5) Woman without her man is a savage?

(T.d.6) Woman without her, man is a savage?

(T.d.7) Woman without her man, is a savage?

(T.d.8) Woman, without her, man is a savage?

(T.d.9) Woman without her man is a savage!

(T.d.10) Woman without her, man is a savage!

(T.d.11) Woman without her man, is a savage!

(T.d.12) Woman, without her, man is a savage!

The good examples of usage of ancient, intermediate and modern language (T.h) are as follows:

(T.h.1) Book of "Sejarah Melayu" (The Malay Annals) ("Sejarah Melayu," 1954) in the ancient Malay language

(T.h.2) Book of "Sejarah Melayu" (Shellabear, 1975) in the intermediate Malay language

A few good examples are given below to show the effect of polysemy (T.j):

(T.j.1) Good

(T.j.2) Well

(T.j.3) Shell

(T.j.4) Lay

(T.j.5) Heart

6.2.4 Multimedia Type of Image

For the multimedia type of image, graphics, or picture, there are hard copy and digital copy after the advent of computer technologies. The various images can be categorized into eight groups (I.1-I.8) as follows:

(I.1) Hard copy with simple black-and-white colours

(I.2) Hard copy with complex black-and-white colours

(I.3) Hard copy with simple multiple colours

(I.4) Hard copy with complex multiple colours

(I.5) Digital copy with simple black-and-white colours

(I.6) Digital copy with complex black-and-white colours

(I.7) Digital copy with simple multiple colours

(I.8) Digital copy with complex multiple colours

The first error level checking is very useful for digital copy of image. Nevertheless, the tool of computing devices is required as an intermediate agent between the human and the image. In addition to the first error level, the advanced error of image as the second error level is important to be taken into consideration for information mining. These semantic errors of image (I.a-I.h) may possibly occur as listed below:

(I.a) Light source(s)

(I.b) Other light source(s)

(I.c) Optical properties of the viewed object

(I.d) Viewer

(I.e) Distance between the viewed object and viewer

(I.f) Medium conditions for the space of light of sight

(I.g) Relative position and velocity of light sources, viewed object and viewer

(I.h) Material decaying factor for the surface of the viewed object

The example of light source(s) (I.a) and optical properties of the viewed object (I.c) are as follows:

(I.a.1) The star of Sun emits the visible light beam of white colour. It is to note here that the basic colours of red, green and blue will compose the white colour. The black colour will be the complement to the white colour.

(I.c.1) The satellite of Moon is a viewed object reflecting the visible light beam of white colour from the star of Sun. The human bare eyes will see the Moon to be in yellow colour being the mixture of red colour and green colour.

Hence, the multimedia type of image is not self independent for bit stream and text, but it depends on the environment factors and viewer as well. In other words, the information mining of image is not an objective issue but a subjective issue.

6.2.5 Multimedia Type of Audio

For the multimedia type of audio or sound, it is either directly communicated from the originator or indirectly communicated via a storage device. The audio data storage can be either in the analogue form or the digital form. In the categorization process, there are four groups (A.1-A.4) as follows:

(A.1) Aural sound (audible to human ears)

(A.2) Vocal sound (speech from human mouth)

(A.3) Environmental sound

(A.4) Music (Jacobs, 1998)

The first error level checking is basically over the bit stream of audio data that is stored digitally after the advent of computer technologies. For the information mining of audio data, it is important to consider the occurrence of second error level, which has the semantic errors of audio (A.a-A.j) as listed below:

(A.a) Homograph where a spelling has many meaning

(A.b) Variation of a human voice like biological factor

(A.c) Pronunciation variation due to contextual factor

(A.d) Musical intonation

(A.e) Environmental factors and noises

(A.f) Situation of speaker

(A.g) Situation of listener

(A.h) Distance between the speaker and listener

(A.i) Medium conditions for the transmission of sound

(A.j) Relative position and velocity of speaker, listener and others

A few good examples are given below to show the effect of homograph (A.a):

(A.a.1) “Sea” and “see”

(A.a.2) “Tree” and “three”

(A.a.3) “Plain” and “plane”

(A.a.4) “Flower” and “flour”

(A.a.5) “Five” and “fine”

For different musical instrument, the musical intonation will be different. The examples of musical intonation (A.d) are as follows:

(A.d.1) Piano

(A.d.2) Guitar

(A.d.3) Saxophone

(A.d.4) Drum

Hence, the multimedia type of audio has lower accuracy rate than the multimedia types of text and image in term of information mining. This is especially true when it comes to the advanced error level of semantic error in the human-machine, human-human, and human-machine-human communication systems.

6.2.6 Multimedia Type of Video

For the multimedia type of video or animation, it is in fact a combination of text, image, and/or audio with the inclusion of time synchronization factor. Video is in fact a word meaning “I see”. The basic video is the body language as like the hand signal, facial expression, gait, etc. This type of multimedia data becomes popular after the advent of television technologies and computer technologies. There are three main groups of video (6.1-6.3) as follows:

(V.1) Body language

(V.2) Analogue video data

(V.3) Digital video data

For the consideration of first error level of video, it is alike the error checking process of the bit stream. For the second error level of video data, it includes all the advanced error of semantic error as for multimedia types of text [(T.a) – (T.j)], image [(I.a) – (I.h)], and audio [(I.a) - (I.j)]. Other than these presented semantic error, there are other semantic errors of video data (V.a-V.c) as follows during the process of information mining:

(V.a) Ambiguous meanings of the body language

(V.b) Time synchronization factor for text, image and audio

(V.c) Material decaying factor of the video storage media

The example of ambiguous meanings of the body language (V.a) is like the polite protocol in welcoming the guest.

The example of time synchronization (V.b) is like the situation of movie-like watching, where the sound system is not designed in good manner and the physical distance creates multimedia differential error of image and audio.

In summary, the semantic error occurrences of video data have about 28 entries. Hence, the accuracy level for the multimedia type of video is lower than the other types of multimedia in the process of information mining.

6.2.7 Duplication of Multimedia Data

After considering the semantic error occurrences for the various types of multimedia data, it is hereby to study the ease of duplication for the multimedia. In simple sentences, there are two types of multimedia data duplication as follows:

(D.1) Hard copy duplication directly usable by human

(D.2) Digital copy or others which require intermediate tool(s) for multimedia data retrieval to be usable by human

Those in the subset of (D.1) are multimedia groups of text [(T.1)] and image [(I.1) – (I.3)]. Meanwhile, those in the subset of (D.2) are multimedia groups of bit stream [(B.1) – (B.3)], text [(T.2)], image [(I.4) – (I.8)], audio [(A.1) – (A.4)] and video [(V.1) – (V.3)]. Hence, it can be concluded here that subset of (D.1) is more human-friendly than the subset of (D.2).

6.2.8 Conclusion

In a nutshell, the multimedia error occurrences have been discussed for the multimedia types of bit stream, text, image, audio, and video. The first error level checking is not enough if we wish to increase the accuracy level of information mining.

Subsequently, the advanced error at the second error level, which is semantic error, is presented here for the various types of multimedia data. Also, the ease of multimedia data duplication has been discussed.

It is hoped that Section 6.2 can help increase the accuracy level of information mining during the multimedia communications. Expected fields for implementation are science, commerce, history, archaeology, etc. One of the few examples for possible practical implementation will be the decryption project for the discovered manuscripts from the Dunhuang caves and other Silk Road sites (International Dunhuang Project (IDP), No date).

6.3 Decrypting English Text Using Enhanced Frequency Analysis

Frequency analysis is the fundamental cryptanalytic technique besides brutal force, threat, blackmail, torture, bribery, etc. (Schneier, 1996, 2000). Conventionally, it is applied on monogram, bigram, and trigram together with anagramming technique (Bauer, 2002). The efficiency of anagramming depends on the cryptanalyst's depth of knowledge in different language features. Hence anagramming is a trial-and-error approach due to huge deviation in different fields, e.g. military, diplomatic, commercial, legal, and daily languages.

The frequency fluctuations among the normal frequency order of the characters trigger the crossover problem for monogram frequency analysis. Meanwhile, bigram and trigram frequency analyses are found to give little help. Higher orders are unlikely to be useful. In year 1994, George Hart (1994) introduced a novel frequency analysis approach based on word frequency to decode the enhanced frequency analysis for systematic decryption of short English ciphertext.

The proposed protocol (Lee, Teh & Tan, 2006) in this Section 6.3 uses combined techniques of monogram frequency analysis, keyword rules, and dictionary checking. It is successfully tested faster on monoalphabetic substitution cipher. Moreover, it solves the weakness of Hart's approach for complete failure when neither one of the top 135 words is found in the ciphertext. Further extrapolation is expected to be applicable to other cryptanalytic algorithms and languages. Specifically here, the possible cracking of short password ciphertext can be reflected.

6.3.1 Introduction

In the scientific world of *secret writing*, there are both the branches of *steganography* using the hiding techniques and *cryptography* using the scrambling techniques (Singh, 1999). To complement the cryptography, which is to reveal the scrambled messages, the human culture achieves the first historical breakthrough in the East via a brilliant mixture of linguistics, statistics, and religious devotion (Kahn, 1996a, 1996b). This new knowledge arena is called *cryptanalysis* by a brilliant cryptologist William Friedman in 1920 (Wrixon, 1998). The etymology of this word is from Latin "crypta", Greek "kryptē" for "crypto" meaning "secret, hidden"; the Greek "ana" for "ana" meaning "up, throughout"; and the Greek "lysys" for "lysis" meaning "a loosing". In other words, cryptanalysis is the interception and solution of a message by a third party. Both cryptography and cryptanalysis are collectively known as *cryptology*.

6.3.2 Justification on Frequency Analysis

Among so many cryptanalytic techniques, *frequency analysis* or *frequency count* is the most basic one other than brutal force, threat, blackmail, torture, and bribery. The frequency analysis is in fact the anatomy of a language. According to a book "*Trattati in cifra*" published in year 1470 and written by Leone Battista Alberti, who is known as "Father of Western Cryptology", the aspect of cryptanalysis using frequency analysis can be traced back to al-Kindī, who is "The Philosopher of the Arabs" and author of 290 books on medicine, astronomy, mathematics, linguistics and music (Kahn, 1996a, 1996b; Bauer, 2002).

In 1987, the Arabic scientist al-Kindī's treatise was discovered in the Sulaimaniyyah Ottoman Archive in Istanbul and entitled "*A Manuscript on Deciphering Cryptographic Message*" (Singh, 1999). It is believed that this manuscript is the first ever known oldest description of cryptanalysis by frequency analysis. The al-Kindī's manuscript was written during the ninth century AD containing discussions on statistics, Arabic phonetics, and Arabic syntax. Al-Kindī's explanation on the revolutionary system of cryptanalytic technique, i.e. frequency analysis, is translated from Arabic language into English language (Singh, 1999) as shown below. It best describes the fundamental operating principles of conventional frequency analysis technique.

> *"One way to solve an encrypted message, if we know its language, is to find a different plaintext of the same language long enough to fill one sheet or so, and then we count the occurrences of each letter. We call the most frequently occurring letter the 'first', the next most occurring letter the 'second', the following most occurring letter the 'third', and so on, until we account for all the different letters in the plaintext sample.*
>
> *Then we look at the ciphertext we want to solve and we also classify its symbols. We find the most occurring symbol and change it to the form of the 'first' letter of the plaintext sample, the next most common symbol is changed to the form of the 'second' letter, and the following most common symbol is changed to the form of the 'third' letter, and so on, until we account for all the cryptogram we want to solve."*

Figure 6.1 Al-Kindī's explanation on the revolutionary system of cryptanalytic technique

All the other cultures then only knew about cryptography and not cryptanalysis. Therefore, it is said that cryptology is born among the Arabs. In addition to al-Kindī's manuscripts, the Arabic knowledge of cryptology was fully set forth in the section of a 14-volume encyclopaedia "*Subh al-a 'sha*" written by al-Qalqashandi in 1412 (Kahn, 1996b). Al-Qalqashandi attributed most of his cryptologic information to be from Ibn ad-Duraihim (1312-1361). Ibn ad-Duraihim-Qalqashandi described the more analytical frequency analysis technique by using the monogram first, then the two-letter word, three-letter word, and so on. Then the concept of two-letter word gave rise to the concept of contacts of letters.

Thereafter, the Arabic cryptologic knowledge in finance, diplomacy, and military fell for 250 years later. Nevertheless, the frequency analysis techniques using monogram and contacts of letters developed by Arabians have become the most universal, most basic cryptanalytic procedures. It is a prerequisite for understanding all subsequent substitution cryptanalysis techniques (Schneier, 1996).

Almost at the same time of Ibn ad-Duraihim, the Western civilization began the record of cryptology around 1326 (Kahn, 1996b). Since then, the Western cryptographic systems merged both of the code and cipher using substitution and transposition for 650 years in both Europe and America. We call this as *secret key cryptosystem* or *classical cryptography* (Kaufman, Perlman & Speciner, 1995). During this period, the substitution cipher evolved from monoalphabetic substitution cipher, to homophonic substitution cipher in 1401, to polygram substitution cipher and polyalphabetic substitution cipher in year 1568 (Schneier, 1996). With the advent of computer, the classical cryptography can easily be decrypted using the frequency analysis and anagramming.

Subsequently, a new cryptographic field called as *public key cryptosystem* (Stallings, 2006b) or *modern cryptography* was introduced by Diffie and Hellman in 1976 (Diffie & Hellman, 1976). In practical life nowadays, both classical cryptography and modern cryptography are used together under a hybrid scheme, e.g. IBM mainframes (Le, Matyas, Johnson & Wilkins, 1993), to gain an optimization between the key security and computational load. Public key cryptosystem is used for exchanging the master keys. Meanwhile, the master keys are used for distributing the secret session keys. The session keys will then be used under the secret key cryptosystem. Now, most of the good cryptographic algorithms combine both of the cryptosystems with the mixed usages of substitution and transposition ciphers.

This is applicable and conveniently implemented with the handy help of nowadays computing facilities. The only difference is just that instead of the 26 elements of alphabetical characters, now we have only two elements, i.e. binary '1' and '0' (Schneier, 1996). Although it looks more complicated, it is basically the same. It is just like the analogy of numerical computations between binary, decimal, and hexadecimal numerical systems. Hence the frequency analysis remains as the prevailing basic requisite for cryptanalysis techniques. This paper presents an improved approach of modified frequency analysis upon the Hart's approach (1994) for the decryption of English ciphertext. The improved cryptanalytical approach is a ciphertext-only attack using the combined techniques of monogram frequencies, keyword rules, and dictionary checking.

6.3.3 Drawbacks of Conventional Frequency Analysis

At a first look into the sentences that describe the frequency analysis on monogram, the method shall work theoretically for sufficient long texts (Bauer, 2002). However, it is found in practical tests that the frequency distribution of English language is a science fiction.

This is due to the fact that the English texts in different circumstances deviate between fields, e.g. military, diplomatic, commercial, legal, literary, etc. Besides, different person may use different words at different frequencies. In addition, an energetic language is alive and always grows with time. The vocabulary is getting more and more abundant. It may even undergo mutation due to the enrichment of foreign words, e.g. *paddy*, *spaghetti*, *kimono*, etc.

Table 6.1 Frequency analyses of various sources [ascribe to Bauer (2002)]

Frequency Counts Descending Order	Sources
eaoid hnrst uycfg lmwbk pqxz	E.A. Poe 1843
etaoi nshrd lucmf wypvb gkqjx z	O. Mergenthaler 1884
etoan irshd lcfum pywgb vkxjq z	P. Valērio 1893
etaon isrhl dcupf mwybg vkqxj z	H.F. Gaines & O.P. Meaker 1939
etoan irshd lcwum fygpb vkxqj z	L.D. Smith 1943
etoan irshd lufcm pywgb vkxzj q	L. Sacco 1951
etaon irshd lucmp fywgb vjkqx z	D. Kahn 1967
etaon rishd lfcmu gpywb vkxjq z	A.G. Konheim 1981
etaoi nsrhl dcumf pgwyb vkxjq z	C.H. Meyer & S.M. Matyas 1982

Hence, these factors may cause the frequency fluctuations for the alphabetic letters in the texts. The fluctuations may then create the crossover problems when we try to match the empirical frequency counts with the ideal frequency counts. Table 6.1 above lists the old frequency analyses for English text with only 10,000 or fewer letters adopted from Bauer (2002). It can be observed in Table 6.1 that the frequency counts depend on the source file at certain deviation degree. The crossover problems in frequency analysis can be ameliorated if the ciphertext is big. A huge ciphertext can reduce the fluctuations of letters as practically tested by Bauer (2002) on German texts. However, it is hard to have a chance to encounter with huge ciphertext.

6.3.4 Combined Methods of Frequency Analysis

To deal with these problems, Bauer (2002) proposed the *cliques* and *partition matching* method. Cliques are groups of letters with almost equi-frequency that are hard to be separated. This method is proven to be good on the conditions that the cliques are not overlapping and having a clear gap between the members of a clique. Hence, it normally works for intermediate-size ciphertexts and not the short ciphertexts.

In addition, Bauer (2002) suggested other advanced frequency analysis methods. The frequency of multigrams, such like *bigram frequencies* of 676 symbols and *trigram frequencies* of 17,576 symbols, are found to be helpful in solving the short ciphertexts that the cliques are not separated well by monogram frequencies. However, higher orders of *n*-gram frequencies with $n \geq 4$ are almost helpless.

Other combined methods proposed by Bauer (2002) were *word frequencies* between two spaces, *position frequencies* of a letter in a word, *average word length*, and *word formation* by alternating and/or accumulating vowels and consonants. Although there are numerous methods here, the only aim is to mechanize the decryption of monoalphabetic simple substitution ciphers. Instead of huge and intermediate texts, our aim also involves the short texts. Bauer's methods fade for short texts.

For short cryptograms or ciphertexts, Hart (1994) proposed an efficient algorithm using the language model. His method of [26!] partial permutation searched through a manageable tree of word assignments. These words are from a listing of 135 top-ranked words of modern American English in descending order (Kučera & Francis, 1967). Firstly, starting from the smallest *n*-letter words with same length and same pattern of repeated letters if any repeat, we compare the *n*-letter words of ciphertext with the 135-word dictionary to check for any possible deciphering. The minimum value of *n* is 1 and the maximum value is 7. "QUESTION" is ranked 358th. In other words, it means there is no 8-letter word in the 135-word dictionary.

Nevertheless, there are some drawbacks in this method as well. While dozens of words match the general 2-letter and 3-letter patterns, higher orders of *n*-letter words after that are of low possibilities to have repeated letters. "THAT" is the only case having the first and fourth letters repeated for 4-letter words. The small size of 135-word dictionary also causes the problem of missing letters like letters 'J', 'Q', 'X', and 'Z'. Hence, one may need to do anagramming via the trial-and-error method to get the ciphertexts in proper meaning by inspection. Therefore, the approach to increase the size of the word dictionary is helpful. It will help decipher the missing letters from the word dictionary.

However, this method has no usage of grammatical information and relies on word counts. In the special case of unusual letter distributions, this cryptanalytic technique may be confused and fail. For the worst case if no plaintext is in the word dictionary, it will definitely fail completely at a possibility of 2^{-n} for *n*-word texts using a 135-word dictionary. Consequently, it is our purpose here to propose an enhanced frequency analysis technique to mechanize the deciphering of this type of ciphertext.

6.3.5 Proposed Method: Enhanced Frequency Analysis

In this proposed enhanced frequency analysis technique, it is a three hierarchical approach. The monogram frequencies, keyword rules, and dictionary checking are implemented one by one to mechanize the full deciphering of monoalphabetic substitution cipher.

In the first step, the monogram frequencies are computed for all of the letters in the ciphertext. Basically, at least the letters of ‘E’ and ‘T’ can be identified or deciphered. During the second step, the keyword rules are conducted to decipher the other unknown secret letters one by one. Each keyword rule is supposed to perform only on cipher words that have only one secret letter. This secret letter will then be deciphered. The sequence of the keyword rules is illustrated in Table 6.2 below. It is to note here that some of the keyword rules may have the sequence swapped. Besides, for letters that are still not yet deciphered in the second step, they will be identified in the third step.

Table 6.2 Sequence of keyword rules for modified frequency analysis

Sq.	Letter	Keyword	Sq.	Letter	Keyword
1	o	to	11	u	our, out
2	n	no, on, not	12	c	can, could
3	f	of	13	w	we, was, with
4	r	or, for	14	g	go, get
5	a	a, at, are	15	l	all, will, could
6	d	do, and	16	b	be, but
7	h	he, the	17	y	by, may, yes
8	i	it, in, if	18	v	have, very
9	s	so, is	19	p	up, people
10	m	from, am, me	20	k	like, back, make

In the last step, there are at least four letters “JQXZ” that have to be recognized by using the dictionary checking or inspection. If the ciphertext still has lots of secret letters, dictionary checking will be in favour by having the cipher words with the least number of secret letters tested first. If the ciphertext has very few secret letters left as for alphabets “xjqz”, then the inspection method is already sufficient to decrypt them.

This three-tier approach has been tested on two short cryptograms with the dimensions of 9006 letters for an excerpt from the storybook of “*Arabian Nights: Ali*

Baba and the Forty Thieves" (Penguin Popular Classics, 1997) and 2802 letters for an excerpt randomly chosen from a daily newspaper article entitled "Bangsar". It is found that both achieve very successful deciphering results at promising computational time. The results are plotted in Figure 6.2. Hence, the deciphering model to mechanize the cryptanalysis of monoalphabetic substitution cipher has been successfully developed and tested.

It is found that this enhanced frequency analysis approach performs at more systematic and faster decryption than the Hart's approach. Moreover, due to the special combined properties of keyword rules and dictionary checking, the Hart's approach weakness (1994) in facing total failure when neither one of the top 135 words are in the English ciphertext could be hindered.

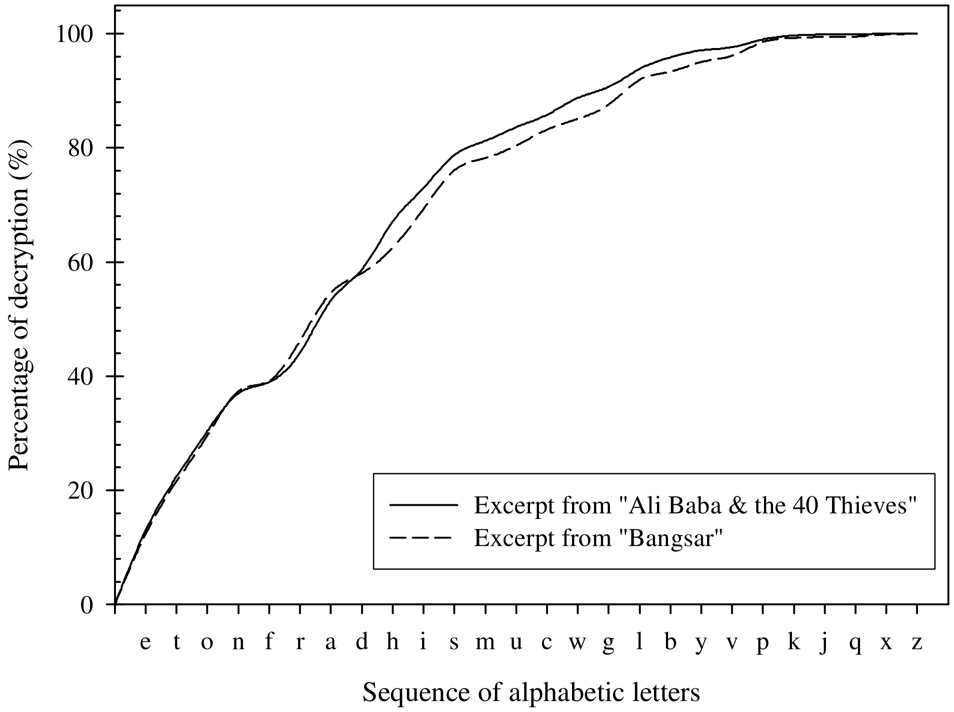


Figure 6.2 Implementation of enhanced frequency analysis

6.3.6 Conclusion

In a nutshell, the cryptanalytic technique of enhanced frequency analysis has been successfully developed by using the combined techniques of monogram frequencies, keyword rules, and dictionary checking. The proposed three-tier approach in Section 6.3 mechanizes the cryptanalysis of monoalphabetic simple substitution cipher. It is also discovered in the research that this enhanced approach shows improved performance from the Hart's approach in the context of English texts. The improvements are faster ciphertext decryption and the avoidance from the chances of total failure when none a single top 135 words is within the ciphertext.

In future, the keyword rules may be refined to have a faster deciphering speed by rearranging the sequence order and/or the selected keyword for specific letters. Furthermore, finer and more advanced algorithms could be established from here to decipher the other advanced substitution and transposition ciphers, e.g. homophonic substitution cipher, polygram substitution cipher, polyalphabetic substitution cipher, and simple columnar transposition cipher.

It is our wish that other advanced ciphers like Playfair Cipher could be decoded. Moreover, there are possibilities that this model could be applied to other languages as well. The advent of Internet has created an e-life society that is not only multidisciplinary but also multilingual. Perhaps, the *Scrabble Players Dictionary* ("The Official Scrabble," 2002) may become another source of idea towards the direction of further improvement. Specifically here, Section 6.3 reflects some facts on the possible cracking of short password ciphertext.

6.4 Passphrase with Semantic Noises and a Proof on Its Higher Information Rate

Key size becomes very important to a cryptographic algorithm according to Kerckhoff's law where a civilian cryptosystem shall depend fully on key secrecy. Currently, there are 4 passphrase generation methods: Sentence, acronym, diceware, and coinware. Unicity distance is the minimum ciphertext size for unique ciphertext decipherability when number of spurious keys is zero. A key with size less than unicity distance is good where there are spurious keys which allow a protection method using limited unsuccessful logins. Here (Lee & Ewe, 2007d), stronger forms of passphrases using textual semantic noises like punctuation marks, mnemonic substitution, misspelling, and associative morphing, which improve the key entropy, are proposed. An ASCII mutual substitution table is presented together with its proof on information rate increment. Higher information rate has lower redundancy, and hence bigger unicity distance ensures encrypted keys the short cryptogram in a key vault, like Password Safe, cannot be cracked within certain limited login attempts.

6.4.1 Introduction

Key is one of the four main groups of entity authentication for identification. These four groups are “something known”, “something possessed”, “something inherent”, and “someone known” (Menezes, Oorschot & Vanstone, 1996). Key is a secret only known to the authenticated entity, where its low cost, mobility, and wide compatibility, makes it to be the most popular authentication method.

Kerckhoff’s law is applied in civilian cryptosystem, where the strength of a cryptosystem is fully dependent on the key secrecy (Schneier, 1996). In other words, key size is the main factor of a cryptographic algorithm. Short key is called password and long key is called passphrase. FIPS PUB 112 (Federal Information Processing Standards Publication 112) dated 30 May 1985 on password usage (NIST, 1985b) defined password to be a key with a length of 4 to 8 characters and passphrase as a key with a length of 9 to 64 characters.

To estimate the key entropy, NIST (National Institute of Standard and Technology), USA, published an electronic authentication guideline (Burr, Dodson & Polk, 2004, 2006; Burr, Dodson, Perlner, Polk, Gupta & Nabbus, 2008). However, the estimation of user-chosen key entropy is based on Shannon’s information theory (Shannon, 1948) and English information rate (Shannon, 1951; Cover & King, 1978) using 26 English alphabets plus one space character. For more accurate figures, the English information rate is in fact shall be based on 95 ASCII printable characters, where its limiting (infinite-length history) conditional entropy is available in (Brown, V. J. D. Pietra, Mercer, S. A. D. Pietra & Lai, 1982; Tsou, Lai & Chow, 2004).

A good key shall be strong and memorizable. A strong key has high entropy and high randomness. Meanwhile, a memorizable key has reasonable secrets to be remembered. Weak key is not in favourite (Klein, 1990; Spafford, 1992a, 1992b). Random key is strong but not memorizable. Hence, this thesis proposes keys with balanced features of strength and memorability (Kurzban, 1985; Bugaj, 1996; Yan, Blackwell, Anderson & Grant, 2004).

Due to the long key size demand for symmetric key algorithm and asymmetric key algorithm (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b; Gehrmann & Näslund, 2005, 2006, 2007), password is no longer enough and passphrase (Schneier, 1996; Menezes, Oorschot & Vanstone, 1996; Stallings, 1995) is needed. For the popular email encryption software, PGP (Pretty Good Privacy) version 9.0, the allowed key size comes to a maximum of 255 characters (PGP Corporation, 2006). Nowadays, many modern operating systems support a maximum key field of 255 characters.

For the entry of passphrase, there are currently four input methods: Sentence, acronym, diceware (PGP Corporation, 2006) and coinware (Lee & Ewe, 2006). In this section, a stronger form of passphrase is proposed to have more spurious keys by using the semantic noises or semantic errors (Lee, 2005a) like punctuation marks, misspelling, mnemonic substitution, and associative morphing (Bugaj, 1996).

6.4.2 Key Sizes and Passphrases

The minimum symmetric and asymmetric key sizes for different protection periods are given by (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b; Gehrmann & Näslund, 2005, 2006, 2007) as in Tables 3.1 and 3.2.

The ASCII entropy is 6.57 bits/letter. Hence, many key sizes are challenging the human memorizability limit. For asymmetric key algorithm before the MePKC, the long key sizes make it impossible to be memorizable in ASCII encoding code and require encrypted private key stored in the computing device. A user has to remember the shorter symmetric key used to protect the encrypted private key. Hence, asymmetric key algorithm before the MePKC is normally having the portability and mobility problems of private key. From Table 3.1, the minimum key size for smallest general purpose level with a maximum of 2-year protection or before year 2010 is 80 bits. If ASCII is used, it needs 13 characters to fulfil the key size requirement. Hence, passphrase shall be used as compared to password.

There are four types of passphrase generation methods (Lee & Ewe, 2006): Sentence, acronym, diceware (PGP Corporation, 2006), and coinware (Lee & Ewe, 2006). Sentence-type and acronym-type passphrases are subject to computational analysis of word frequency distribution. Meanwhile, diceware and coinware are immune to the computational analysis due to the feature of random word selection from dictionary and self-created word list(s).

Sentence-type passphrase uses an entire phrase or full sentence to form a key. Acronym-type passphrase applies abbreviation of first, second, last, etc., letters of each word in a sentence. Diceware uses dice to choose a word from an ordered word list. The word list can be in any language and based on senary or base-6 numeral system. Coinware is similar to diceware except it uses coin to select from a word list that can be monolingual, bilingual, or multilingual. It is especially efficient for word list in binary, octal, and hexadecimal orders. There are readily built word lists for Han characters in Unicode-encoded CJK languages.

6.4.3 Unicity Distance and Passphrases with Semantic Noises

Capitalization and *permutation* are the prior arts to increase the passphrase entropy (Stallings, 1995). *Mnemonic substitution* and *associative morphing* are other forms of prior arts. The latter two methods are presented in very brief manner without a proof (Bugaj, 1996). Here, these four methods are generalized together with another two methods from the author's research project, i.e. *punctuation marks* and *misspelling*, as passphrase with semantic noises, and propose a user template of ASCII mutual substitution table, which comes together with a proof on the information rate increment.

Passphrase with semantic noises has higher information rate (r), lower redundancy (D) (Ritter, 2001; Wikipedia Contributors, 2008f), and hence bigger unicity distance (n_0). Unicity distance (Stinson, 2002; Wagstaff, 2003) is the minimum ciphertext size for unique decipherability of ciphertext given sufficient decryption time, when number of spurious keys (F) is zero. The larger the difference between the unicity distance and key length, the more the spurious keys, and the stronger is the protection method using limited login attempts. Special key management algorithms allow multiple site keys (aka slave keys) to be created from a master key (Yee & Sitaker, 2006; Lee & Ewe, 2007a). This further permits each short cryptogram of keys in a key vault like Password Safe to be encrypted by different slave keys. The decoding of short cryptogram has been studied by Hart (1994). Faster decryption can be achieved using an enhanced frequency analysis (Lee, Teh & Tan, 2006). The relationships of information rate, absolute rate like random signal (R), key entropy (H(K)), ciphertext size (n), and unicity distance are given in Equations (6.1-6.4).

$$D = R - r \tag{6.1}$$

$$F \geq 2^{H(K) - nD} - 1 \tag{6.2}$$

$$n_0 \geq H(K) / D \quad \text{when } F = 0 \tag{6.3}$$

$$r \uparrow \Rightarrow D \downarrow \Rightarrow F \uparrow, n_0 \uparrow. \tag{6.4}$$

For instance, English text has r = 1.3 bits/letter for 27 symbols (a, b, c, …, z, space) (Shannon, 1951; Cover & King, 1978). The redundancy is R = 4.75 bits. If AES-128 is used, H(K) = 128 bits. Hence, n_0 = 128/3.45 = 38 characters. For 95 ASCII printable characters, the upper bound of r based on English language becomes 1.75 bits/letter (Brown, V. J. D. Pietra, Mercer, S. A. D. Pietra & Lai, 1982). The revised n_0 = 128/(8-1.75) = 21.

Below are the examples of passphrases with semantic noises using punctuation marks, misspelling, mnemonic substitution, associative morphing, capitalization, and permutation. Punctuation mark is the easiest. A user is encouraged to embed semantic noises for all types of passphrases: Sentence, acronym, diceware, and coinware. Wingate and Sinden (2002) have written a book on how to create short text messages. SMS (Short Message Service) language (Wikipedia Contributors, 2008ag) and text messaging (Wikipedia Contributors, 2008ah) are other good references. The presence of these spurious keys is very useful for the case of limited login attempts, where unique cryptanalysis is impossible.

Actual key: Woman without her man is a savage.

Semantic noises: *Punctuation marks* and *permutation* {

Woman without her, man is a savage.

Woman without her man, is a savage.

Woman, without her man is a savage.

Woman without her, man is a savage?

Woman without reh man, si a savage?

Woman, without reh man si a savage?

Woman without her man is a savage!

Woman without her, man is a savage!

Woman without reh man, si a savage!

Woman, without reh man si a savage!

}

Actual key: To be, or not to be: That is the question.

Semantic noises: *Misspelling* and *capitalization* {

To be? or not to be? That is the question?

To be, or not to be? that is the question!

T0 6e? 0r n0t t0 6e? th@t i5 the que5ti0n?

To be! Or not to be! That is the question!

TO BE! OR NOT TO BE! THAT IS THE Question!

To we, or not to we: That is the question.

To me, or not to me: That is the question.

To be, of not to be: That is the question.

}

Actual key: Ballon, Address? Atmel. ~Star~

Semantic noise: *Mnemonic substitution* {

B@!!0n, Address? Atmel. ~Star~

Ballon, Address? @mail. ~Star~

}

Semantic noise: *Associative morphing* {

Ballon, +++re$$? Atmel. ~Star~

Ballon, Address? Atmel. ~****~

B@!!0n, +++re$$? @mail. ~****~

}

Table 6.3 ASCII mutual substitution table

<table>
<tr><td>aA</td><td>bB</td><td>cC</td><td>dD</td><td>eE</td><td>fF</td><td>gG</td><td>hH</td><td>iI</td><td>jJ</td></tr>
<tr><td>^</td><td>6</td><td><</td><td>o|</td><td>3</td><td>|=</td><td>9</td><td>|-|</td><td>!</td><td>?</td></tr>
<tr><td>kK</td><td>lL</td><td>mM</td><td>nN</td><td>oO</td><td>pP</td><td>qQ</td><td>rR</td><td>sS</td><td>tT</td></tr>
<tr><td>|<</td><td>1</td><td>TV|</td><td>TV</td><td>0</td><td>|o</td><td>&</td><td>|-</td><td>5</td><td>+</td></tr>
<tr><td>uU</td><td>vV</td><td>wW</td><td>xX</td><td>yY</td><td>zZ</td><td>0</td><td>1</td><td>2</td><td>3</td></tr>
<tr><td>[_]</td><td>\/</td><td>vv</td><td>><</td><td>`/</td><td>2</td><td>O</td><td>l</td><td>Z</td><td>E</td></tr>
<tr><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>+</td><td>-</td><td>*</td><td>/</td></tr>
<tr><td>h</td><td>S</td><td>b</td><td>L</td><td>B</td><td>g</td><td>t</td><td>_</td><td>x</td><td>|</td></tr>
<tr><td>%</td><td>=</td><td>[</td><td>]</td><td>{</td><td>}</td><td>(</td><td>)</td><td><</td><td>></td></tr>
<tr><td>o/o</td><td>eq</td><td>{</td><td>}</td><td>[</td><td>]</td><td><</td><td>></td><td>(</td><td>)</td></tr>
<tr><td>!</td><td>“</td><td>#</td><td>$</td><td>&</td><td>‘</td><td>,</td><td>.</td><td>:</td><td>;</td></tr>
<tr><td>i</td><td>„</td><td>n</td><td>m</td><td>Q</td><td>,</td><td>‘</td><td>*
dot</td><td>;</td><td>:</td></tr>
<tr><td>?</td><td>@</td><td>\</td><td>^</td><td>_</td><td>`</td><td>|</td><td>~</td><td colspan="2">space</td></tr>
<tr><td>j</td><td>at</td><td>`</td><td>A</td><td>-</td><td>\</td><td>/</td><td>^v</td><td colspan="2">CamelCase</td></tr>
</table>

6.4.4 Proof of Higher Information Rate

Here, an ASCII mutual substitution table is presented as user template to create passphrase with semantic noises. A user can modify any mutual substitution of these ASCII characters in Table 6.3. CamelCase (Wikipedia Contributors, 2008g) makes compound words or phrases in which the words are joined without spaces and are capitalized within the compound. The ASCII substitution is a token with one or more characters. The probability of the initial token letter is used as the token probability, where the difference is assumed to be small and negligible.

The upper bound of information rate (r) (Shannon, 1951; Cover & King, 1978) is given by Equations. (6.5-6.6), where $q^N{}_i$ is the probability for predictor to discover the correct letter following a sequence of N-1 symbols in i guesses. i indexes one of the 95 ASCII printable characters.

$$q_i^N = \sum p(j_1, j_2, ..., j_{N-1}, j_N) \tag{6.5}$$

$$\sum_{i=1}^{95} i(q_i^N - q_{i+1}^N)\log_2 i \le r \le -\sum_{i=1}^{95} q_i^N \log_2 q_i^N \tag{6.6}$$

Due to the mutual substitution of ASCII printable characters, 95 $q^N{}_i$ becomes about 48 pairs. Every two different $q^N{}_{i-}$ with different probabilities are paired to share the same probability. Let one of these pairs has probabilities A and B before mutual substitution, and probability C after mutual substitution. Let other ASCII characters have a combined probability D, where (A + B + D = 1) and (A + B)/2 = C. The inequality Equation (6.7) can be proven using differentiation of calculus dy_2/dA in Equation (6.11) as derived from Equations (6.8-6.10). y_1 is a constant and y_2 has an absolute minimum value equal to y_1 at A = B = (1 – D)/2. The other two critical points, A = 0 and 1 – D, share the same absolute maximum value. Table 6.4 shows how to determine the critical points of function y_2.

$$-\,2C * \log_2 C \ge -\,A * \log_2 A - B * \log_2 B \tag{6.7}$$

$$C^{2C} \le A^A B^B \tag{6.8}$$

$$((1 - D)/2)^{1-D} \le A^A\,(1 - D - A)^{1-D-A} \tag{6.9}$$

$$y_1 = ((1 - D)/2)^{1-D} \quad (6.10)$$

$$y_2 = A^A (1 - D - A)^{1-D-A} \quad (6.11)$$

Inequality Equation (6.6) can be further extended to three or more mutually substituted ASCII characters for higher information rate increment. The best case is all the ASCII characters can be mutually substituted, which creates the highest information rate like absolute rate the random signal, where unicity distance will become infinite. However, this is just an ideal dream. What we can do is to approach the dream as close as possible. As in Equation (6.4), higher information rate has more spurious key.

$$y_2'(A) = dy_2/dA = y_2 \ln (A / (1 - D - A)) \quad (6.12)$$

$$y_2''(A) = d^2y_2/dA^2 \quad (6.13)$$

$$y_2''(A) = (dy_2/dA) * \ln (A / (1 - D - A)) + y_2 * (1/A + 1/(1 - D - A)) \quad (6.14)$$

When $y_2'(A) = 0$, then $A = (1 - D)/2$. (6.15)

Other critical points are $A = 0$ and $(1 - D)$. (6.16)

Since $y_2''(A = (1 - D)/2) = +ve$,

then $A = (1 - D)/2$ is an absolute minimum value. (6.17)

$$y_2(A = 0) = y_2(A = 1 - D) \quad (6.18)$$

Table 6.4 Determination of the critical points of function y_2

Critical Points	A = 0	A = (1 – D)/2	A = 1 – D
Left of $y_2'(A)$		-ve	+ve
Right of $y_2'(A)$	-ve	+ve	
Types of Extreme Values	Absolute Maximum	Absolute Minimum	Absolute Maximum

6.4.5 Conclusion

Here, stronger form of passphrase is proposed using semantic noises generalizing the punctuation marks, capitalization, permutation, mnemonic substitution, associative morphing, and misspelling. Passphrase with semantic noises has higher information rate, bigger unicity distance, and more spurious keys, which strengthens the login protection with limited attempts. In addition, an ASCII mutual substitution table and its proof on information rate increment is provided.

6.5 Information Rates and Unicity Distances

Below are the denotation to show the information rates and unicity distances of ASCII-based 2D key, Unicode-based 2D key, black-and-white multilingual key, and colourful multilingual key. Unicity distance is the minimum size of ciphertext for the unique decipherability of ciphertext when given enough time.

$$n_0 = \text{unicity distance} \tag{6.19}$$

$$F = \text{number of spurious keys} \tag{6.20}$$

$$K = \text{key} \tag{6.21}$$

$$H(K) = \text{key entropy} \tag{6.22}$$

$$D = R - r \tag{6.23}$$

where D = redundancy, R = absolute rate like random signal, r = information rate

$$F \geq 2^{H(K) - nD} - 1 \tag{6.24}$$

$$n_0 \geq H(K) / D \quad \text{when } F = 0 \tag{6.25}$$

For limiting conditional entropies of English language (Shannon, 1951; Cover & King, 1978; Brown, V. J. D. Pietra, Mercer, S. A. D. Pietra & Lai, 1982; Yannakoudakis & Angelidakis, 1988), Chinese language (Fossum, No date; Vines & Zobel, 1998; Zhang, Xu & Huang, 2000; Tan & Yap, 2001; Tsou, Lai & Chow, 2004; Li, Hu, Wang & Dai, 2005), Malay language (Tan, 1981), and various languages (Behr, Fossum, Mitzenmacher & Xiao, 2002, 2003), these are the references.

$$\text{English} = 1.3 \text{ bits/letter (26 alphabets + space), } 1.75 \text{ bits/ASCII character} \tag{6.26}$$

$$\text{Chinese} = 4.0462, 4.1, 4.5, 5.17 \text{ bits/character from four different studies} \tag{6.27}$$

For the information rate and unicity distance of ASCII-based 2D key:

$$D = 8 - 1.75 = 6.25 \tag{6.28}$$

$$n_0 \text{ (128-bit key)} = 128/6.25 = 20.48 \tag{6.29}$$

$$n_0 \text{ (256-bit key)} = 256/6.25 = 40.96 \tag{6.30}$$

Number of required ASCII characters (128-bit key) = 128/6.57 = 19.48 (6.31)

Number of required ASCII characters (256-bit key) = 256/6.57 = 38.96 (6.32)

For the information rate and unicity distance of Unicode-based 2D key:

$$D = 16 - 4.1 = 11.9 \tag{6.33}$$

$$n_0 \text{ (128-bit key)} = 128/11.9 = 10.76 \tag{6.34}$$

$$n_0 \text{ (256-bit key)} = 256/11.9 = 21.51 \tag{6.35}$$

Number of required Unicode symbols (128-bit key) = 128/16 = 8.00 (6.36)

Number of required Unicode symbols (256-bit key) = 256/16 = 16.00 (6.37)

For the information rate and unicity distance of black-and-white multilingual key:

$$R = 17 + 3 \tag{6.38}$$

$$r = 4.1 + 3 \tag{6.39}$$

$$D = 20 - 7.1 = 12.9 \tag{6.40}$$

$$n_0 \text{ (128-bit key)} = 128/12.9 = 9.92 \tag{6.41}$$

$$n_0 \text{ (256-bit key)} = 256/12.9 = 19.84 \tag{6.42}$$

Number of required image selections (128-bit key) = 128/20 = 6.40 (6.43)

Number of required image selections (256-bit key) = 256/20 = 12.80 (6.44)

Number of required image selections (236-bit key with 20-bit key strengthening)

= 236/20 = 11.80 (6.45)

For the information rate and unicity distance of (16+1) colourful multilingual key:

$$R = 17 + 3 + 8 \tag{6.46}$$

$$r_1 = 4.1 + 3 + 8 \text{ (colours encoded as digits)} \tag{6.47}$$

$$\text{or } r_2 = 4.1 + 3 + 1.75 \text{ (colours encoded as alphabets)} \tag{6.48}$$

$$D_1 = 28 – 15.1 = 12.9 \text{ or } D_2 = 28 – 8.85 = 19.15 \tag{6.49}$$

$$n_{01} \text{ (128-bit key)} = 128/12.9 = 9.92 \tag{6.50}$$

$$n_{01} \text{ (256-bit key)} = 256/12.9 = 19.84 \tag{6.51}$$

$$n_{02} \text{ (128-bit key)} = 128/19.15 = 6.68 \tag{6.52}$$

$$n_{02} \text{ (256-bit key)} = 256/19.15 = 13.37 \tag{6.53}$$

$$\text{Number of required image selections (128-bit key)} = 128/28 = 4.57 \tag{6.54}$$

$$\text{Number of required image selections (256-bit key)} = 256/28 = 9.14 \tag{6.55}$$

$$\text{Number of required image selections (236-bit key with 20-bit key strengthening)} = 236/28 = 8.43 \tag{6.56}$$

If the numbers of required ASCII/Unicode characters and image selections (T) are less than the respective unicity distance, then there will be no unique decipherability. The more spurious keys, the better it is. For higher information rate and smaller T, then there are more spurious keys when T < n_0. Setting an attempt limit to open a key vault using a master key can then become a possible method of protection. Equations (6.57-6.60) show the required formulas to estimate the unicity distance and T.

$$\text{Let T = numbers of required ASCII/Unicode characters and image selections.} \tag{6.57}$$

$$F \geq 2^{H(K) – TD} – 1 \qquad \text{when } n = T, T < n_0 \tag{6.58}$$

$$T \geq (H(K) – \log_2(F + 1)) / D = n_0 – (\log_2(F + 1)) / D \tag{6.59}$$

$$\log_2(F + 1) \geq D(n_0 - T) \tag{6.60}$$

Moreover, to know the information content of selected images as secret key in some of the big secret creation methods, please refer to Tavakoli (1991).

CHAPTER 7 MULTIHASH KEY

7.1 Introduction

There are lots of situations that require a user to have many online and offline accounts. Examples of online and offline accounts are login access and file encryption, respectively. For safer security, a secret cannot be re-used to avoid password domino cracking effect (Ives, Walsh & Schneider, 2004), where an attacker starts the password cracking process from the weakest link. However, according to R. Kanaley (2001), an Internet user manages an average 15 keys on a daily basis. Yet in another survey by Adams and Sasse (1999), a user can only be expected to handle 4 to 5 unrelated and regularly used keys. For user's unique keys without the constraint of relevancy, Florencio and Herley's survey (2007) reported an average 6.5 keys, repeated 3.9 times each for 25 accounts and typing 8 keys daily. Hence, there is a memory burden to the user unless these secrets are written down somewhere. However, important password the secret is discouraged to be jotted down somewhere.

7.2 Multiple hashes of single key with passcode for multiple accounts

A human's e-life needs multiple offline and online accounts. It is a balance between usability and security to set keys or passwords for these multiple accounts. Password reuse has to be avoided due to the domino effect of malicious administrators and crackers. However, human memorizability constrains the number of keys. Single sign-on server, key hashing, key strengthening, and petname system are used in the prior arts to use only one key for multiple online accounts. The unique slave keys (aka site keys) are derived from the common master secret and specific domain name. These methods cannot be applied to offline accounts such as file encryption. New method and system are invented to be applicable to offline and online accounts. It does not depend on http server and domain name, but numeric 4-digit passcode, key hashing, key strengthening, and hash truncation. Domain name is only needed to resist spoofing and phishing attacks of online accounts.

7.2.1 Introduction

For friendly environment, cost effectiveness, and efficiency, human civilizations are heading towards a paperless and electronic society. Every human is getting numerous offline and online accounts. These accounts require authentication to gain system access. There are four types of authentication approaches: Secret, token, biometrics, and introducer.

Secret is about something you know like password or key. Token is about something you have like smart card. Biometrics is about something you are like fingerprint. Introducer is whom you know. For the sake of cost and compatibility, secret in the form of key is the most popular authentication approach.

According to Forrester Research (Kanaley, 2001), an active Internet user manages an average of 15 keys on a daily basis. Most people, who are majority-wise, not using the password management tools, either maintain the same key for all the accounts, write down different keys for different accounts, or keep closely related keys for various accounts. These are all poor password management practices.

The HTTP basic authentication protocol (even over SSL) (Franks, Hallam-Baker, Hostetler, Lawrence, Leach, Luotonen & Stewart, 1999) allows a server to know the key of each account. This causes possible malicious server attacks from the administrators and crackers. The server may be untrustworthy or compromised.

For another HTTP specification, i.e. HTTP digest authentication protocol, challenge-response protocol is used (Franks, Hallam-Baker, Hostetler, Lawrence, Leach, Luotonen & Stewart, 1999). The server can still see the clients' keys. Since the response from a client to a server is not specific to the server, HTTP digest authentication protocol is vulnerable not only to malicious server attacks, but password file compromise attacks, spoofing attacks, and phishing attacks.

If a key is reused, the success of an attack on an account in a weak system may cause a strong system to be compromised. This password reuse can trigger a domino effect from the weakest system to the strongest system (Ives, Walsh & Schneider, 2004).

Therefore, every key has to be uniquely set for each account, regardless of weak or strong system, to get rid of the risk when one system is compromised. However, according to (Adams & Sasse, 1999), users can only be expected to cope with a maximum of four or five keys that are unrelated and regularly used. When key relevancy is allowed, a user can cope with average 6.5 unique keys (Florencio & Herley, 2007). This reflects the need to balance the usability and security.

To address this problem, some key management tools are invented. These tools allow users to remember only one master secret as master key and assign unique keys to multiple accounts. They allow users either to choose their own master key and then store the site keys (aka slave keys) somewhere safe, or to assign fixed keys to each website that can be computed whenever they are needed.

The examples of the first approach are Password Safe and Windows Live ID. The examples of the second approach are LPWA (Lucent Personal Web Assistant), HP Site Password, Password Multiplier, SPP (Single Password Protocol), PwdHash, and Passpet. A special example using the hybrid approach is CPG (Compass Password Generator).

Password Safe is a password vault that can be used for offline and online accounts. However, its mobility is low due to the requirement to have a safe storage for multiple keys encrypted by a common master key. Another form of solution for online accounts only is to use a single sign-on server and its proxy servers. Microsoft Windows Live ID (aka Microsoft Passport Network) is one of these examples. Its weaknesses are single point of failure and high cost of integration.

Another method to reduce the memory burden of online account passwords uses key hashing and key strengthening (aka key stretching) of a master key concatenated with a domain name and optional username. Exemplary applications of this method are (i) LPWA (Lucent Personal Web Assistant) (Gabber, Gibbons, Matias & Mayer, 1997); (ii) HP Site Password (aka System-Specific Passwords or Site-Specific Passwords) (Karp & Poe, 2002; Karp, 2003); (iii) Password Multiplier (Halderman, Waters & Felten, 2005); (iv) PwdHash (Ross, Jackson, Miyake, Boneh & Mitchell, 2005); and (v) Passpet (Yee & Sitaker, 2006).

There is also a method using unique random number assignment to different online accounts called CPG (Compass Password Generator) (aka Common Password Method) (Luo & Henry, 2003). Yet there is another method using the key hashing of one-time ticket, server name, and master password to generate different site keys (aka slave keys) called SPP (Single Password Protocol) (Gouda, Liu, Leung & Alam, 2005).

All these methods of single master key generating multiple site keys or slave keys apply only to online accounts having a domain name. Its weakness is a change of master key requires all the accounts to be updated one by one, which is required by some key management strategies.

For offline account, the current prior art uses a password vault to store all the unique passwords the secret. These password vaults can be simply an encrypted spreadsheet or document file, or application software like Password Safe by Bruce Schneier [URL: http://www.schneier.com/passsafe.html]. The disadvantage of password vault is its low mobility and danger of disclosing the ciphertext of password vault to the public domain. Hence, there exists a need to have a method to generate multiple slave keys of online and offline accounts from a master key, and yet an individual slave key can be changed without changing the master key and other slave keys.

With the realization of big memorizable secret for cryptographic, information-hiding, and non-cryptographic applications, especially MePKC, there are even more types of offline accounts like asymmetric private key, stego-key, symmetric watermarking key, asymmetric watermarking private key, and PRNG seed. Among them, for MePKC cryptographic applications like encryption, signature, authentication, key exchange, and other schemes, different schemes require a different pair of asymmetric key pair, by the technical and law requirements to have a safer electronic information society. Hence, there exists a need to generate multiple private keys as slave keys from a common memorizable master key.

The present invention (Lee & Ewe, 2007a) can be applied to offline and online accounts with good mobility. Domain name is not necessary but optionally needed to resist phishing attacks and spoofing attacks. A single sign-on server is also not needed. The required components are numeric 4-digit passcode, key hashing, key strengthening, and hash truncation.

To allow diversity of site keys from a single master key, there are two optional entries: Username ID and domain name (or website) URL. Domain name that is also used to resist phishing attacks can be replaced by adopting an anti-phishing tool. In other words, the proposed new method and system can be used together with an anti-phishing tool.

These anti-phishing tools are SpoofStick, Netcraft Toolbar, Earthlink Toolbar, SiteKey, DSS with SRP (Dynamic Security Skins with Secure Remote Password Protocol), Petname Tool, TrustBar, and Passpet (Yee & Sitaker, 2006).

7.2.2 Related Works

Here, the prior arts of key management tools are discussed, where a single key can be used for multiple accounts, in a deeper context. Anti-phishing tools will not be discussed. Accounts are divided into two types: Offline and online. Offline accounts have no domain name while online accounts have domain name. Example of offline accounts is file encryption; whereas example of online accounts is email.

Password Safe is an application software originally developed by Bruce Schneier [URL: http://www.schneier.com/passsafe.html]. It uses the Twofish encryption algorithm to protect the stored passwords by a master password. Users need only to remember one master password to access multiple passwords. Its mobility depends on the available password database. It can be used for both offline and online accounts, but cannot resist spoofing attacks and phishing attacks.

Windows Live ID (No date) is also known as "Microsoft Passport Network" [URL: http://www.passport.net]. Users need a master password to sign on a central server. This central server will authenticate users for multiple servers which have joint network. Besides single point of failure, it has high cost of integration. Some security loopholes are reported (Kormann & Rubin, 2000). It can be used for online accounts only, but can resist phishing and spoofing attacks.

LPWA (Gabber, Gibbons, Matias & Mayer, 1997; Matias, Mayer & Silberschatz, 1997) uses key hashing of master password and domain name to generate a specific site password via a server. It has single point of failure but not the high cost of integration. However, the malfunction of central authority will mean the breakdown of all services. It can be used for online accounts only and can resist phishing and spoofing at-tacks. Nowadays, it has stopped providing the services.

HP Site Password (Karp, 2003; Karp & Poe, 2004) is also called “System-Specific Passwords” or “Site-Specific Passwords”. A master password and a system name are concatenated, hashed using MD5 (Rivest, 1992) and converted into Base64 encoding (Borenstein & Freed, 1992) to get a site password. It is not centralized using a server but operates as stand-alone application in the terminal computers. It can be used for online accounts only and cannot resist phishing and spoofing attacks.

It is important to note here there were few successful collision attacks over the MD5 in the years 2004-2006 (Wikipedia, 2008r). The successor of MD5, which is SHA-1, is also discovered to be subject to collision attacks on its reduced version in the years 2004-2006 (Wikipedia, 2008w). Consequently, NIST announced that SHA-1 would be phased out by the year 2010 in favour of SHA-2 variants: SHA-224, SHA-256, SHA-384, and SHA-512 (NIST, 1995a, 2002b; 2007b; Lilly, 2004).

CPG (Luo & Henry, 2003) is also called “Common Password Method”. It assigns unique random numbers to different website accounts. The random number is hashed using MD5 and converted using a binary-to-text transform to generate a specific password for multiple accounts. The random number is encrypted and stored in an account server or proxy server. When a user needs to access a specific account, the encrypted random number is retrieved from the server, decrypted, hashed, and converted into a specific password to authenticate the access. Therefore, it has the weakness of single point of failure, but does not involve the high integration cost like LWPA. It is for online accounts only and can resist phishing and spoofing attacks.

Password Multiplier (Halderman, Waters & Felten, 2005) uses key hashing and key strengthening. There are two levels of hash iterations using the inputs of username, master password and site name. Both the numbers of hash iterations are fixed for 100 seconds and 1/10 second, respectively. It is a stand-alone application without using a server and implemented using browser extension to Mozilla Firefox. It can be used for online accounts only and can resist phishing and spoofing attacks.

SPP (Gouda, Liu, Leung & Alam, 2005) is also a stand-alone application. It applies the techniques of challenge-response protocol, one-time server-specific ticket and key hashing using MD5 or SHA-1. The site password is hashed from the one-time ticket, server name, and master password. The one-time ticket and site password will be updated after every login access. It can be used for online accounts only and can resist phishing and spoofing attacks.

PwdHash (Ross, Jackson, Miyake, Boneh & Mitchell, 2005) is implemented using browser extensions to Mozilla Firefox, Internet Explorer, and Opera. Its key hashing inputs the domain name of remote site into a pseudo-random function controlled by user's master password. The domain name acts as a hash salt. It can be used for online accounts only and resist phishing and spoofing attacks.

Passpet (Yee & Sitaker, 2006) is also implemented using browser extension to Mozilla Firefox. It applies the techniques of petname system, key hashing, key strengthening, and UI customization. Petname system is a naming system possessing the properties of globality, security and memorizability (Wikipedia, 2008k). It is used for anti-phishing attacks. Key hashing and key strengthening in Passpet are alike the Password Multiplier using the SHA-256, except that its first level of hash iterations is flexible in amount allowing updates according to the computer technology advancement without changes of software. It uses local storage for login access via a fixed machine, and remote storage in a server for login access with mobility feature. The remote server stores the first level of hash iterations and site label file that is encrypted from the site label list. Due to the dependency of server for newly used machines, Passpet has some risks of single point of failure. However, there is no high cost of integration. It can be used for online accounts only and can resist phishing and spoofing attacks.

7.2.3 Basic Model of Multihash Key

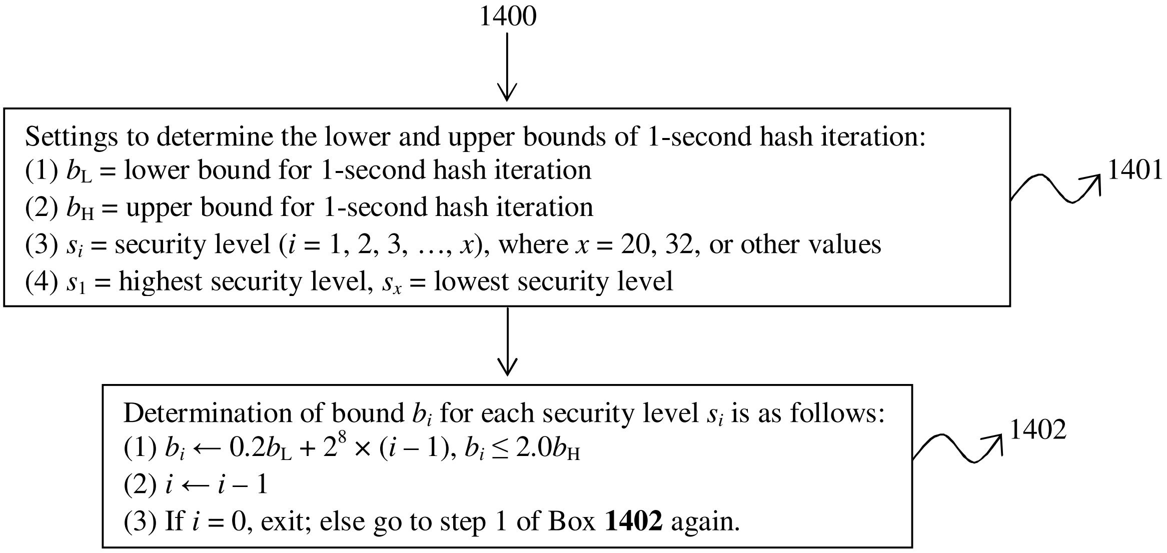
The proposal here requires users to remember an at least 128-bit master key and a numeric 4-digit passcode. This method and system is named "multihash key".

The passcode is used together with key hashing, key strengthening (Manber, 1996; Abadi, Lomas & Needham, 1997, 2000; Kelsey, Schneier, Hall & Wagner, 1997) and hash truncation to generate exemplary 20 unique hashes at 20 security levels for 20 accounts. Each security level has one account. These hashes are site keys. All the security levels are ranked from the highest security (#1) to the lowest security (#20). This is because knowing the multihash key at the higher level can reveal the multihash key at the lower level, but not the reverse.

From Kanaley's survey (2001), 20 accounts are set since an active Internet user manages an average of 15 keys daily. Five accounts are added by assuming that there are five offline accounts. The number of accounts can be increased by changing the settings or remembering another pair of (master key, passcode).

There are three pseudo-codes for multihash key to show how the method and system work: Determination of hash iterations of multiple security levels, generation of multihashes as site keys, and changes of key pair (master key, passcode).

As an example, Figure 7.1 shows the determination of 20 security levels via the experiments to locate the lower bound b_L and upper bound b_H for 1-second hash iterations for an old computer that is slow but still popular. Each security level is partitioned by 2^8.

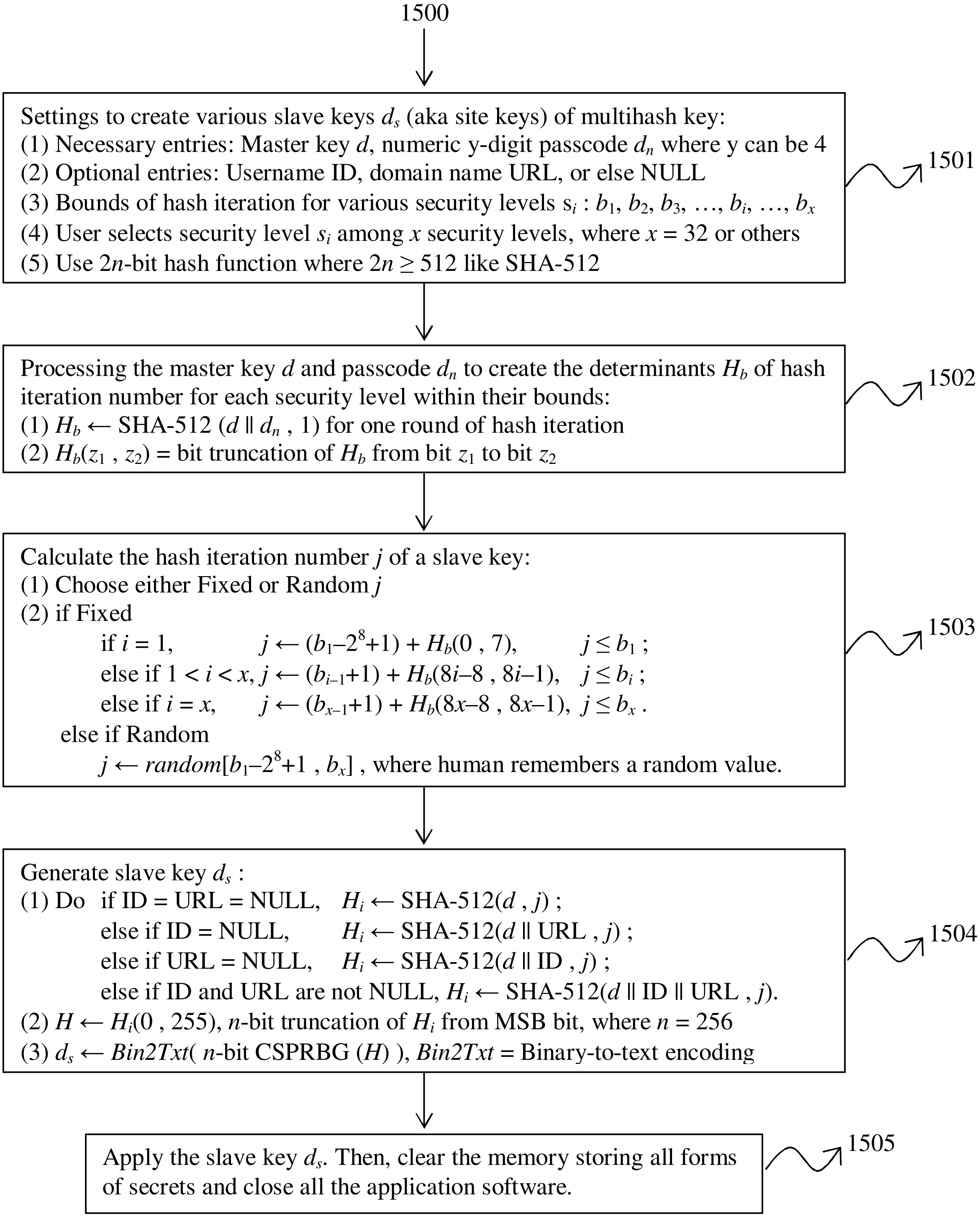


Settings to determine the lower and upper bounds of 1-second hash iteration:
(1) b_L = lower bound for 1-second hash iteration
(2) b_H = upper bound for 1-second hash iteration
(3) s_i = security level (i = 1, 2, 3, …, x), where x = 20, 32, or other values
(4) s_1 = highest security level, s_x = lowest security level

Determination of bound b_i for each security level s_i is as follows:
(1) $b_i \leftarrow 0.2b_L + 2^8 \times (i - 1)$, $b_i \leq 2.0b_H$
(2) $i \leftarrow i - 1$
(3) If $i = 0$, exit; else go to step 1 of Box **1402** again.

Figure 7.1 Pseudo-code to determine the numbers of hash iteration for multiple security levels of multihash key methods and systems

Figure 7.2 presents the basic model of multihash key to generate multihashes as site keys. A user needs to remember the selected security level for a specific account. In case of forgetfulness, all the 20 security levels shall be tried one by one.



Settings to create various slave keys d_s (aka site keys) of multihash key:
(1) Necessary entries: Master key d, numeric y-digit passcode d_n where y can be 4
(2) Optional entries: Username ID, domain name URL, or else NULL
(3) Bounds of hash iteration for various security levels s_i : $b_1, b_2, b_3, \ldots, b_i, \ldots, b_x$
(4) User selects security level s_i among x security levels, where $x = 32$ or others
(5) Use $2n$-bit hash function where $2n \geq 512$ like SHA-512

Processing the master key d and passcode d_n to create the determinants H_b of hash iteration number for each security level within their bounds:
(1) $H_b \leftarrow$ SHA-512 ($d \parallel d_n$, 1) for one round of hash iteration
(2) $H_b(z_1 , z_2)$ = bit truncation of H_b from bit z_1 to bit z_2

Calculate the hash iteration number j of a slave key:

```
(1) Choose either Fixed or Random $j$
(2) if Fixed
        if $i = 1$,            $j \leftarrow (b_1-2^8+1) + H_b(0 , 7)$,           $j \leq b_1$ ;
        else if $1 < i < x$, $j \leftarrow (b_{i-1}+1) + H_b(8i-8 , 8i-1)$,  $j \leq b_i$ ;
        else if $i = x$,       $j \leftarrow (b_{x-1}+1) + H_b(8x-8 , 8x-1)$,  $j \leq b_x$ .
    else if Random
        $j \leftarrow random[b_1-2^8+1 , b_x]$ , where human remembers a random value.
```

Generate slave key d_s :

```
(1) Do  if ID = URL = NULL,    $H_i \leftarrow$ SHA-512($d$ , $j$) ;
        else if ID = NULL,        $H_i \leftarrow$ SHA-512($d \parallel$ URL , $j$) ;
        else if URL = NULL,     $H_i \leftarrow$ SHA-512($d \parallel$ ID , $j$) ;
        else if ID and URL are not NULL, $H_i \leftarrow$ SHA-512($d \parallel$ ID $\parallel$ URL , $j$).
(2) $H \leftarrow H_i(0 , 255)$, $n$-bit truncation of $H_i$ from MSB bit, where $n = 256$
(3) $d_s \leftarrow Bin2Txt($ $n$-bit CSPRBG ($H$) ), $Bin2Txt$ = Binary-to-text encoding
```

Apply the slave key d_s. Then, clear the memory storing all forms of secrets and close all the application software.

Figure 7.2 Operation of the basic model of multihash key method and system

Necessary entries are master key d and numeric 4-digit passcode d_N. Optional entries are username ID and website (or domain name) URL. The username and website are used to create diversity of multihash key from a key pair (master key, passcode). Domain name can also help to resist phishing and spoofing attacks.

The 512-bit hash of the concatenated master key and passcode is truncated into 20 partitions with 8-bit each from the MSB bit. This increases the randomness of specific keys for different accounts. If an attacker does not know the exact security, then 5120 hashes have to be checked for any key pair (master key, passcode). If the attacker knows about the security level, then 28 hashes have to be validated for any key pair (master key, passcode).

For the settings of bound b_i, it can be either fixed or random. If the fixed option is chosen, the number of hash iterations will use the standard settings. A user is mobile and can use this method without remembering the number of hash iterations while accessing offline and online login account from different computing systems.

If the random option is chosen, the number of hash iterations will be randomly selected by a user within a given range. User's mobility is weakened unless one can remember the random values of hash iterations while accessing offline and online logins. However, if a user can remember the hash iterations, this option offers stronger resistance to dictionary attack. The best option is a hybrid scheme. Choose fixed option for lower security levels and random option for higher security levels.

Depending on the value existence of username ID and domain URL, the master key undergoes different key hashing and key strengthening using SHA-512 to generate hash H_i. H_i is then encoded from binary to text to fulfil the demands of password requirements such as alphanumeric, mixed lowercase and uppercase, and with punctuation marks.

Here, a binary-to-text encoding of *Bin2Txt*(*H*) is proposed as in Table 7.1. Base64 encoding is not used as there are only two punctuation marks included (Borenstein & Freed, 1992). *Bin2Txt*(*H*) converts 6 binary bits into one 8-bit ASCII character. It has a bit expansion of 33%. All types of ASCII characters are included: lowercase, uppercase, digit, and punctuation marks. The last group of 4 binary bits of *H* from 253rd to 256th is padded with 2 binary bits of 0 at the right or LSB side. The output of *Bin2Txt*(*H*) is a string of 43 ASCII characters and is used as key hash.

Table 7.1 Binary-to-text encoding Bin2Txt(H) of multihash key

Bin	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
Txt	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
Bin	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Txt	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Bin	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
Txt	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
Bin	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Txt	!	“	#	$	%	&	‘	(	)	*	+	,	-	.	/	@
Bin	Padding															
Txt	~															

N.B.: Bin: For easy understanding, decimal value is shown to represent binary values

Lastly, copy the hash as site key into the clipboard and paste it on the prompt key field for authentication access. **Remember to clear the clipboard before leaving the computer.**

On how to change from an old key into a new one, a user can change either the master key, passcode, security level, username, or the domain name. There are also proposed usages of 20 security levels as shown in Figure 7.3.

New methods and systems called multihash key and its variants are presented here to generate multiple slave keys (aka site keys) from a single master key for both the offline and online accounts. Among various cryptographic, information-hiding, and non-cryptographic applications needing secrets for various types of key, here are some of the popular applications of secret key: (i) Master key for password vault hiding various keys; (ii) Internet banking; (iii) online stock trading; (iv) insurance; (v) tax; (vi) office, school and home email accounts; (vii) instant messengers; (viii) encrypted files; (ix) database accounts at the office and school; (x) library accounts; and (xi) verification key for credit card. Hence, the impact contribution of multihash key shall be very high in the aspects of reducing the human memorization burden and system operating costs.

The multihash key method and system uses the hash iteration and hash truncation, followed by optional n-bit CSPRBG to increase the randomness, as for a basic model of multihash key as in Figure 7.2, to generate slaves keys from a master key and an optional passcode. The master key and hash function shall be at least $2n$ bits. The passcode shall be at least 4 digits or more. The hash iteration applies the key strengthening for a period ranging from 0.2 to 2 seconds, or longer to 10 seconds in some of the variants of multihash key. Hash truncation halves the hash value or message digest. Multihash key supports infinite number of online accounts and limited number of offline accounts depending on the performance of the computer. Examples of online accounts are webmail, login, email, and instant messenger. Examples of offline accounts are encrypted file, public-key certificate, bank ATM card, and software token.

Security levels: usages
1 Password file and key management tool like password vault.
2 Finance=>Very important Internet banking.
3 Finance=>Important Internet banking.
4 Finance=>Stock trading.
5 Finance=>Insurance, income tax, ...
6 Very important personal encrypted files, email accounts, instant messengers, …
7 Important personal encrypted files, email accounts, instant messengers, …
8 Very important accounts in working/studying place like email.
9 Important accounts in working/studying place like database.
10 Other accounts in working/studying place like library.
11~20 Other not frequently used offline and online accounts.

Figure 7.3 Proposed usages of 20 security levels

Table 7.2 Comparisons of key management tools

Features\Key management tools	Plain browser	Password autofill	Password safe	Windows live ID	LPWA	HP site password	CPG	Password multiplier	SPP	Pwd Hash	Passpet	Multihash key
Usability												
1. Make logging in more convenient	No	Yes	No	Yes	Yes	No	?	Yes	?	No	Yes	?
2. Work with existing websites	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
3. Allow site-by-site migration to tool	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
4. Change individual site keys	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes	–	Yes	Yes
5. Log in from other computers	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
6. Only need to memorize one secret	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	–	Yes	Yes
7. Enable changing the master secret	–	–	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
8. Applicability to offline accounts	–	–	Yes	No	No	No	No	No	No	No	No	Yes
9. Applicability to online accounts	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
10. Integrate usages together with other tools	–	–	Yes	No	No	Yes	No	No	No	No	No	Yes
Security												
1. Unique key for each account	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
2. Resist offline dictionary attacks	No	No	No	No	No	No	No	Yes	No	No	Yes	Yes
3. Adapt to increasing CPU power	No	No	No	No	No	No	No	Yes	No	No	Yes	Yes
4. Avoid storing keys	Yes	No	No	No	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
5. Avoid a single central authority	Yes	Yes	Yes	No	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
6. Resist phishing by fake login forms	No	No	No	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
7. Resist mimicry of browser UI	No	No	No	No	No	No	No	No	No	No	Yes	No
8. Help the user identify websites	No	No	No	No	No	No	No	No	No	No	Yes	No
9. Stop entering secrets in webpages	No	No	No	No	No	Yes	?	Yes	?	No	Yes	?
Possible implementation												
1. Stand-alone application	No	No	Yes	No	No	Yes	No	Yes	Yes	Yes	No	Yes
2. Single sign-on server	No	No	No	Yes	Yes	No	Yes	No	No	No	No	No
3. Browser extension	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

?: Unknown situation depending on implementation

7.2.4 Discussions

Table 7.2 compares various key management tools with multihash key from the aspects of usability, security, and possible implementation. A lot of comparisons are attributed by Yee and Sitaker (2006) on Passpet. New features used for comparisons are applicability to offline and online accounts, integrated usages together with other key management tools and possible implementations. It is important to note here that multihash key can be used together with Passpet to earn "Yes" for items [I.7-I.9] under the security features in Table 7.2.

Multihash key can be used for both offline and online accounts. Possible implementations are stand-alone application and browser extension. These are simple interfaces to input a password or key with unique key images for multiple accounts. Memorizability is improved since there is only one secret for various login accounts.

Server is not used and hence there is no central authority. There are no single point of failure and high cost of integration. It is mobile and there is no encrypted storage of site keys. Since there is no integration, multihash key can be used for any existing computer systems.

The passcode is optional to be remembered by a user because it can be converted to be an 8-bit password supplement in one of the two methods of key strengthening (Manber, 1996; Abadi, Lomas & Needham, 1997, 2000). Master key is the password, and when it is combined with the password supplement, they form the full password. Another key strengthening method is also called key stretching, which uses a large amount of hash iterations (Kelsey, Schneier, Hall & Wagner, 1997).

The variant of SHA-2, which is SHA-512, is used in the key hashing and key strengthening. This is because there are possible collision attacks to MD5 and SHA-1. The hash truncation creates a 256-bit hash as site key. The unused truncated bit creates a 128-bit security strength (256/2=128) preventing the compromised site keys at the higher security level from revealing the site keys at the lower security level. The passcode also has this feature but is very much less powerful.

For the experimental data of lower bound b_L and upper bound b_H of some computer systems, there are reported as follows. For instance, for the first computer system of desktop PC [Example 1: Pentium II 266MHz, 192MB RAM, running on Windows XP Professional Edition], the lower and upper bounds for 1-second hash iteration, as in Figure 7.1, are 7600 and 8200, respectively. In other words, the first computer system can only support 20 offline accounts for a security level partitioning of 8 bits or 2^8.

Yet in the second computer system of laptop PC [Example 2: Centrino Duo 1.66GHz, 1.5GB RAM, running on Windows XP Home Edition], the lower and upper bounds for 1-second hash iteration are 81,700 and 93,700 respectively. For this specification, the second computer system can support 256 offline accounts for a security level partitioning of 8 bits or 2^8.

Yet in the third computer system of desktop PC [Example 3: Pentium D 2.80GHz, 512MB RAM, running on Windows XP Professional Edition SP3], the lower and upper bounds for 1-second hash iteration are 52,500 and 122,500 respectively. For this specification, the third computer system can support 256 offline accounts for a security level partitioning of 8 bits or 2^8.

Using the proposed settings, the key strengthening has an access time from 0.2 second to 2 seconds. This is an efficient range of acceptable login processing time. It can be calibrated to be parallel with the advances in computer technologies for new releases of multihash key. Moore's Law is a good rule to judge the calibration, which is about one bit faster for every two years (Wikipedia, 2008v).

Key hashing and key strengthening are also good techniques to resist offline and online dictionary attacks as well as pre-computation attacks. To prevent phishing and spoofing attacks, multihash key can either be used together with other anti-phising tool like petname system and Passpet, or include domain name URL in its key hashing. Malicious server attack is also prevented as different accounts have unique passwords. For homograph attack due to visually similar Unicode graphic symbols, the implementation of multihash key shall support the Unicode characters.

Up to here, the basic model of multihash key can support infinite number of online account. Meanwhile, the number of supported offline account by multihash key is given by Equation (7.1). From Figure 7.2, the security level x can be increased up to the maximum of hash iteration number $j_{\max}$. Also, hash functions beyond 512 bits like 768 and 1024 bits may be needed.

$$\mathrm{S}_{\mathrm{AC0}} = x \tag{7.1}$$

7.2.5 Limitations

Multihash key can be implemented as a stand-alone application with no change of setting at the server side. However, it is vulnerable to password file compromise attacks and message log file attacks. Nevertheless, domino effect of password reuse can be avoided. To get rid of password file compromise attacks and message log file attacks, some countermeasures (Gouda, Liu, Leung & Alam, 2005) can be adopted by changing the settings of authentication approach at client and server sides.

Acting as a stand-alone application, multihash key requires a user to perform extra steps. These steps are creating a key, copying, and pasting it to a login prompted textbox. The user also needs to remember the security level of an account, an at least 128-bit master key, and a numeric 4-digit passcode. These cause the solution to be not user-friendly.

To facilitate the application, multihash key has to be integrated into the user interface of each authentication application. Therefore, the item [I.1] of usability in Table 7.2 about convenient logging in depends on implementation.

For security level, it can be jotted into a notebook in plaintext form because it is not an essential secret. Alternatively, for online account, the user can be reminded about the security level whenever the user sends the username to the server. This allows an attacker to reduce the number of hash testing by 20 times, or 4.32 bits ($=\log_2 20$).

For numeric 4-digit passcode, it gives an extra security of 13.29 bits ($=\log_2 10\ 000$) and is not an essential secret. This passcode can be made constant for user with poor memory. For user who can remember 128-bit master key, 4-digit passcode and security level, the effective security strength is 145.61 bits. For user who can remember only the 128-bit master key, its security strength is 128 bits. Hence, security can be compensated for better usability.

Using multihash key, limited number of multiple offline and online accounts can be supported as compared to the almost infinite number of online accounts for LPWA, HP Site Password, CPG, Password Multi-plier, SPP, PwdHash, and Passpet. For more accounts, faster computer system is needed to have larger bound range. Or else, the partition between any two security levels has to be reduced.

7.2.6 Conclusion

The proposed invention of multihash key requires users to remember a master key and passcode to generate unique key hashes or site keys for multiple accounts. For security level, username, and domain name of a specific account, users can choose to write them down somewhere as there are not critical secrets. This is a balance between the usability and security.

Multihash key can be used for offline and online accounts, where existing similar key management tools without encrypted site key storage can only be applied to online accounts. It is hoped that this proposal can release the human memory burden on required passwords or keys for various types of increasing accounts. To have better resistance to phishing and spoofing attacks, try to use multihash key together with an anti-phishing tool like petname system and Passpet.

7.3 More Slave Keys for Offline Accounts per Master Key of Multihash Key

Key hashing of master key and domain name generates site keys for online account only like login. Meanwhile, multihash key using hash truncation can be applied for both online account and offline account like password encryption of electronic file using symmetric key cryptography and partial memorizable private key of the asymmetric key cryptography. Nevertheless, even though infinite online account can be supported by multihash key, only limited offline account is possible. The current computer speed limits it to 20 offline accounts unless we have faster computer system, reduced partition length, or more passcodes.

Here, three methods are proposed to support more offline accounts per master key of multihash key using filename, random number, or two-tier structure. Multihash key using a random number can now have up to 2^{256} site keys for electronic file of offline account. Meanwhile, multihash key using a two-tier structure can support multiple partial memorizable private keys of the split private key cryptography up to 400 offline site keys.

To support more offline accounts, especially the various cryptographic schemes of MePKC, multihash key is further enhanced. Firstly, hashing the concatenation of a master key and a filename is proposed as in Figure 7.4a. As long as the filename is unique, infinite offline accounts can be supported. However, the problem is name clashing and renaming.

Secondly, a random number is used without and with multihash key, respectively, as in Figures 7.4b-7.4c, where this random number is concatenated with master key in a hashing process to generate a slave key. For a ciphertext encrypted using this slave key, the random number has to be retrieved first. Hence, this random number is encrypted using the master key and stored as a concatenation to a file ciphertext encrypted by the slave key to become an output file. When a user wants to open the file ciphertext, one splits the output file to get the ciphertexts of file and random number. Decrypt the ciphertext of random number using the master key. Then, generate the slave key using the master key and the recovered random number. Subsequently, the file ciphertext is decrypted by the slave key. Using AES-256, this method using a random number can support 2^{256} offline accounts. However, its drawbacks are major modification to the current computer systems and no support for secrets of offline accounts without any ciphertext storage, such as split private key cryptosystem and MePKC.

Then, a third method, as in Figure 7.4d, using a two-tier structure of multihash key is proposed. For the examples of the first and second computer systems, 400 and 65536 offline accounts, respectively, can be supported. This method is compatible with the current computer system. Yet the special advantage of this method is its support for secrets of offline accounts without any ciphertext storage. In other words, the partially and fully memorizable private keys of split private key cryptosystem and MePKC can now be supported.

1600

<table>
<tr><td colspan="2"><b>Using Filename</b></td></tr>
<tr><td colspan="2">P = plaintext, K_M = master key, F = file name, K_S = slave key, C_1 = file ciphertext,</td></tr>
<tr><td><b>Encryption</b>
$K_S \leftarrow$ Hash (K_M || F)
$C_1 \leftarrow$ Encrypt (P , K_S)</td><td><b>Decryption</b>
$K_S \leftarrow$ Hash (K_M || F)
P $\leftarrow$ Decrypt (C_1 , K_S)</td></tr>
</table>

Figure 7.4a First method: Using filename

1601

Using Random Number without Multihash Key

P = plaintext, K_M = master key, R = random number, K_S = slave key,
C_R = R ciphertext, C_1 = file ciphertext, C_2 = output file,
Truncation = *n*-bit truncation, *j* = hash iteration number

Encryption	**Decryption**
$K_S \leftarrow$ Truncation (Hash ($K_M \parallel R$))	$[C_R , C_1] \leftarrow$ Truncation(C_2)
$C_1 \leftarrow$ Encrypt (P , K_S)	$R \leftarrow$ Decrypt (C_R , K_M)
$C_R \leftarrow$ Encrypt (R , K_M)	$K_S \leftarrow$ Truncation (Hash ($K_M \parallel R$))
$C_2 \leftarrow (C_R \parallel C_1)$	$P \leftarrow$ Decrypt (C_1 , K_S)

Figure 7.4b Second method: Using random number without multihash key

1602

Using Random Number with Multihash Key

Encryption	**Decryption**
$K_S \leftarrow$ Truncation (Hash ($K_M \parallel R , j$))	$[C_R , C_1] \leftarrow$ Truncation (C_2)
$C_1 \leftarrow$ Encrypt (P, K_S)	$R \leftarrow$ Decrypt (C_R , K_M)
$C_R \leftarrow$ Encrypt (R, K_M)	$K_S \leftarrow$ Truncation (Hash ($K_M \parallel R , j$))
$C_2 \leftarrow (C_R \parallel C_1)$	$P \leftarrow$ Decrypt (C_1 , K_S)

Figure 7.4c Second method: Using random number with multihash key

1603

Using Two-Tier Structure of Multihash Key

Settings: Multihash = multihash key function, K_M = master key, K_N = passcode K_{S1} = first slave key, K_{S2} = second slave key, j_1 = first hash iteration number, j_2 = second hash iteration number, K_S = final slave key for various offline accounts

(1) $K_{S1} \leftarrow$ Multihash (K_M , K_N , j_1)
(2) $K_{S2} \leftarrow$ Multihash (K_{S1} , K_N , j_2)
(3) Stored in notebook $\leftarrow$ (j_1 , j_2)
(4) $K_S \leftarrow K_{S2}$

Figure 7.4d Third method: Using two-tier structure of multihash key with manually selected security levels

Figure 7.4 Methods and systems to support more offline accounts for multihash key

7.3.1 Related Works

In the information era, human has many keys or passwords for various offline and online accounts. On 4 February 2001, Kanaley (2001) with Forrester Research reported that an active Internet user averagely manages 15 keys on a daily basis. On the other hand, Adams and Sasse (1999) reported that a user can only be expected to cope with 4 to 5 keys that were unrelated and regularly used. For user's unique keys without the constraint of relevancy, Florencio and Herley's survey (2007) reported an average 6.5 keys, repeated 3.9 times each for 25 accounts and typing 8 keys daily. This reflects a demand to enhance human memorizability to support many accounts needing a key. Online account is like email and webpage login. Offline account is like file encryption, encrypted private key, and memorizable private key of MePKC.

Key hashing of master key and domain name (or server name) is used to generate site keys for online accounts (Gabber, Gibbons, Matias & Mayer, 1997; Karp, 2003; Karp & Poe, 2004; Halderman, Waters & Felten, 2005; Gouda, Liu, Leung & Alam, 2005; Ross, Jackson, Miyake, Boneh & Mitchell, 2005; Yee & Sitaker, 2006). Other methods of single master key for multiple online site keys are single sign-on server (Windows Live ID, No date) and CPG (Common Password Method) using website account with assigned random number (Luo & Henry, 2003).

Meanwhile, multihash key (Lee & Ewe, 2007a) using key strengthening and hash truncation can be applied for both online account and offline account. Nevertheless, even though infinite online account can be supported by multihash key, only limited offline account is possible. The current computer performance limits it to 20 offline accounts unless the computer system is faster or reduced the partition length between the two security levels.

However, the computer speed increases by 1 bit for every 24 months. It is not surprising where one may have a few hundreds of accounts up to 512. We may need to wait for 10 years if computer speed is expected to increase the supported offline account. For the reduction of partition length, it decreases the attacking time because the security level of those account holders with poor memory is public. The current partition length at 2^8 is already considered short. It is not encouraged to be shorter.

Instead, the better approach is to increase the partition length in order to increase the attacking time whenever the security level is known.

Hence, the only current possible practical way to have over 20 offline accounts is to change the passcode (Lee & Ewe, 2007a). However, this method is not appropriate for those with poor memory. Moreover, the increment of offline account is additive, which means an increment of 20 offline accounts for every additional passcode. Here, three methods for multihash key are proposed to support more offline accounts using filename, random number, or two-tier structure.

7.3.2 First Method: Using Filename

Domain name or server name is commonly used to generate infinite slave keys (aka site keys) for online ac-counts from a master key (Gabber, Gibbons, Matias & Mayer, 1997; Karp, 2003; Karp & Poe, 2004; Halderman, Waters & Felten, 2005; Gouda, Liu, Leung & Alam, 2005; Ross, Jackson, Miyake, Boneh & Mitchell, 2005; Yee & Sitaker, 2006). The general equation is in Equation (7.2).

$$\text{Slave Key} = \text{Hash (Master Key} \parallel \text{Domain Name)} \tag{7.2}$$

For offline account like file encryption and encrypted private key, a tagged filename informs that no change can be made. Then, Equation (7.3) can be used to support over 20 offline accounts, where filename replaces the domain name. This method using filename is shown in Figure 7.4a.

$$\text{Slave Key} = \text{Hash (Master Key} \parallel \text{Filename)} \tag{7.3}$$

As long as the filename is different, infinite offline account can be supported. The disadvantage of this method is that the computer file system has to be restructured so that there is an option to ban the change of filename. Having another file to hint that a specific filename of a file cannot be changed may have problem because a user may forget to transfer and store these two files together.

Another problem is that due to different computer operating systems, the filename with some special characters like space mark and some punctuation marks tends to change automatically when it is transferred and saved from one computer to another computer.

Thirdly, for non-ASCII multilingual filename like CJK filename, it may have problem as well when the file is transferred and saved among different computing systems. For encrypted private key in the form of smartcard, there exists the problem that no filename is available. Consequently, this method using filename can only be conditionally implemented independent of or together with multihash key.

7.3.3 Second Method: Using a Random Number

CPG (Luo & Henry, 2003) assigned a random number for every website account and stored it in a server to generate unique site keys for different online accounts. It is not designed for offline account at all. Here, a new algorithm is designed using random number as well to support over 20 offline accounts up to the set size of the random number. This new algorithm can be implemented independent of or together with multihash key. Figure 7.4b shows an independent implementation from multihash key; whereas Figure 7.4c shows an implementation together with multihash key.

Let P = file plaintext, R = random number, K_M = master key, K_S = slave key, C_R = ciphertext of R, C_1 = file ciphertext, C_2 = output, Hash(X || Y) = hashing the concatenation of X and Y for one round, Hash(X || Y , *b*) = hashing the concatenation of X and Y for *b* round, Enc = encryption, Dec = decryption, Trunc = truncation, || = concatenation, and B(x) = bit length of x. Equation (7.4) is the setting.

$$B(K_S) = B(K_M), \quad B(\text{Hash}) \geq 3B(K_M), \quad B(R) \leq B(\text{Hash}) - B(K_M) \qquad (7.4)$$

For security analysis of both the algorithms using random number in Figures 7.4b-7.4c, an attacker will try to recover the random number R and file plaintext P. The ciphertext of random number C_R cannot be cryptanalyzed as long as it is random, which means there is no loophole in the pseudorandom number generator (PRNG).

Another condition for the random number R is that its bit length is maximized as in Equation (7.3) with promising random number generation speed.

When the cryptanalysis of C_R is impossible, the only attacking place is the cryptanalysis of file ciphertext C_1. C_1 can be obtained easily because the truncation function of C_2 to get C_R and C_1 is public. For algorithm in Figure 7.4b independent of multihash key, guessing attack and pre-computation attack are possible and take less computing efforts than the algorithm in Figure 7.4c. This is because the pre-computation attack over the algorithm in Figure 7.4b requires only one round of hashing to generate the slave key K_S. Meanwhile, algorithm with multihash key in Figure 7.4c requires many rounds of hashing. The extra security strength of Figure 7.4c algorithm from Figure 7.4b algorithm is because of the key strengthening (Manber, 1996; Abadi, Lomas & Needham, 1997, 2000; Kelsey, Schneier, Hall & Wagner, 1997). Key strengthening, which is also called key stretching, is a process to hash an input value for many rounds until a response time of about 1 second.

The hash truncation in both Figures 7.4b-7.4c introduces more difficulty to get master key K_M from compromised K_S. If an attacker can recover P and K_S from C_R, it is still very hard for him to recover K_M. Whenever the K_M is not compromised, the compromise of a single offline account will not affect the other offline accounts. Of course, the security of the hash function is also very important. The security strength of a hash function is half of its bit length. In case of compromised K_S, B(Hash) has to be at least triple of B(K_M) as in Equation (7.3). For instance, if SHA-384 is used, the K_M, K_S, and system security strength are all 128 bits. For hash function longer than 512 bits, a developer can refer to scalable polymorphic hash (Roellgen, No date, 2005).

For computation load of algorithm in Figure 7.4b, it runs faster than algorithm in Figure 7.4c. However this faster speed is not good because it also means faster possible cryptanalysis. Figure 7.4c algorithm runs slower but it has acceptable response time with key strengthening set at about 1 second. The delay of response time is significant as it also freezes the requirement of longer key size as the computer technologies are advancing. The number of hash iteration can be simply increased to maintain a response time at about 1 second. For random number generation, Le Quere (2004) invented a fast RNG with 100 Mbps. Its drawback is that a special physical RNG is needed. Another alternative without special hardware is a PRNG based on discrete logarithm problem by Gennaro (2005) with 860 bits per 160-bit exponent exponentiation when typical parameters are n = B(modulo prime) = 1024 and c = B(exponent) = 160.

The first weakness of this method using random number is that it cannot be implemented on the current encryption system. Only new encryption system with updated software program and hardware device can adopt it. Also, for some special cases where there is no ciphertext, the ciphertext of a random number has no place to be stored. This happens to the private key storage technology like split private key (Sandhu, deSa & Ganesan, 2005c, 2006c, 2006f). Different asymmetric key pairs may be needed for different purposes like encryption, digital signature, key exchange, authentication, digital timestamping, and multipartite public key cryptography for electronic commerce.

7.3.4 Third Method: Using a Two-Tier Structure

The third method is proposed to get rid of the disadvantages of the second method. It can only be used together with multihash key. For the mean time, the increment of supported offline account is additive with extra 20 site keys for every additional passcode. Here, the increment of the slave keys for offline accounts can be made to be multiplicative by having a two-tier structure.

For this two-tier multihash key, there are two rounds of multihash key. The slave key from the first round of the multihash key (intermediate site key) will become the master key of the second round of the multihash key (intermediate master key). The slave key from the second round of the multihash key (final slave key) will be used for the offline account like file encryption and private key.

Subsequently, there are 20*20 = 400 site keys for offline accounts. This will be sufficient for the current usage demand of asymmetric key pair. In case it is not enough, the two-tier structure can become three-tier structure or higher order. In case it is a three-tier multihash key, there are 20*20*20 = 8000 site keys for offline accounts. Figure 7.4d shows a two-tier multihash key.

For this method, there exists a problem to remember the combination of security level. A user may need to save this combination somewhere in a paper notebook or encrypted electronic file. Consequently, portability is a weakness in case the notebook is not around. However, its advantages are support to old encryption system and applicability to offline account without ciphertext storage like the memorizable partial private key of split private key technology (Sandhu, deSa & Ganesan, 2005c, 2006c, 2006f) and fully memorizable private key of MePKC.

7.3.5 Conclusion

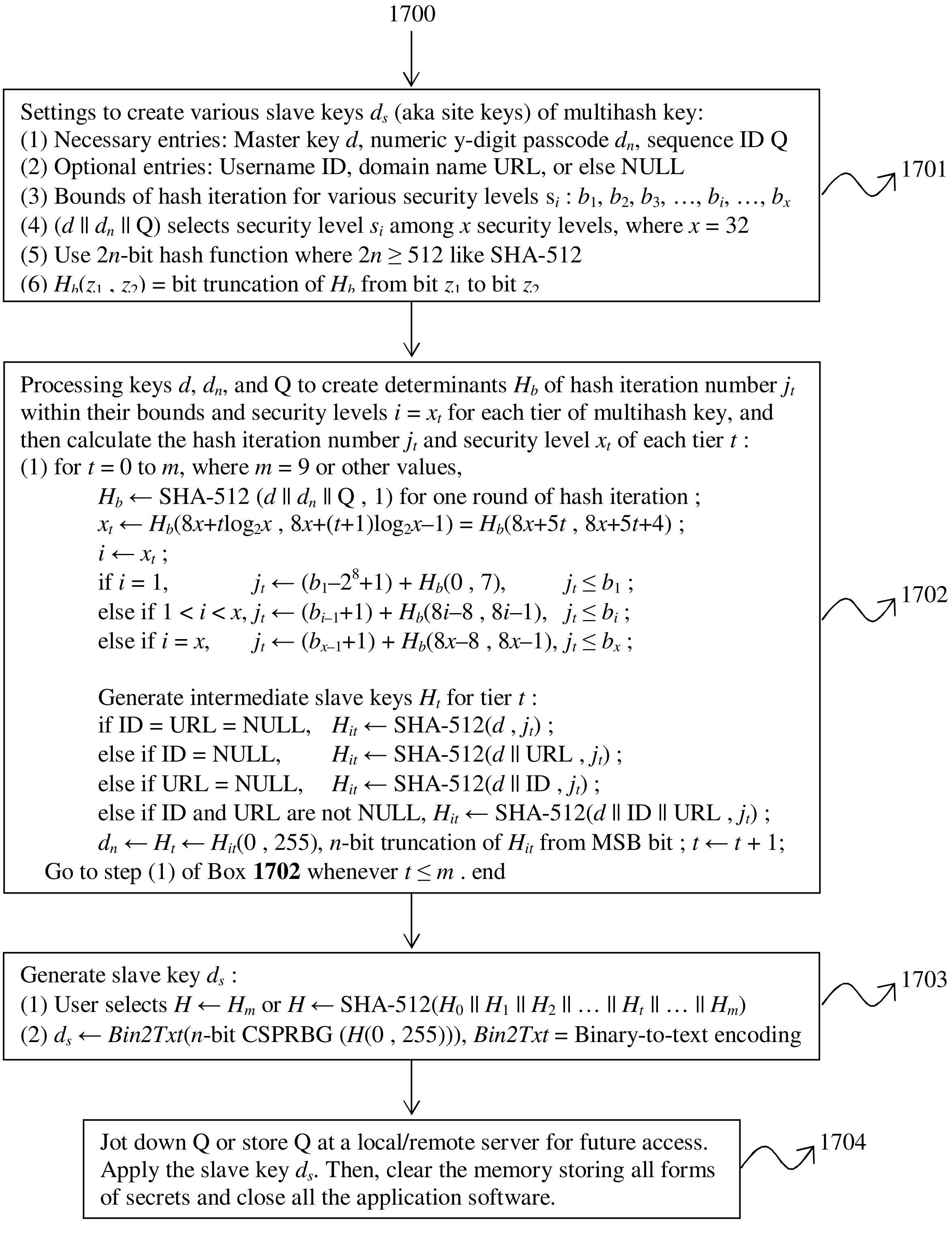
Three methods to increase the number of slave keys for offline accounts have been proposed: Using a filename, random number, and two-tier structure. The first method is the simplest, but has impractical implementation conditions. The second method is good for new file encryption system with ciphertext storage. Complementarily with the second method, the third method can be used for old and new file encryption system without ciphertext storage. In a nutshell, human needs memorizability enhancement like multihash key to have various slave keys from a master key for both the offline and online accounts.

The software implementations (Lee, 2007a, 2008i, 2008k) of the multihash key have been built up by using the Microsoft Visual Studio (Marshall, 2003) and can be evaluated by downloading a copy of this software from [URL: www.xpreeli.com].

7.4 First Variant of Multihash Key Using Automatically Selected Tiers and Security Levels

Besides the basic model, multihash key has been innovated to have some variants. The first variant in Figure 7.5 supports more offline accounts by using automatically selected tiers and security levels.

Box **1701** gives the settings to create various slave keys d_s (aka site keys) of multihash key. Necessary entries are master key d, numeric y-digit passcode d_n, where y can be 4, and sequence ID Q.



Settings to create various slave keys d_s (aka site keys) of multihash key:
(1) Necessary entries: Master key d, numeric y-digit passcode d_n, sequence ID Q
(2) Optional entries: Username ID, domain name URL, or else NULL
(3) Bounds of hash iteration for various security levels s_i : $b_1, b_2, b_3, \ldots, b_i, \ldots, b_x$
(4) ($d \parallel d_n \parallel$ Q) selects security level s_i among x security levels, where $x = 32$
(5) Use $2n$-bit hash function where $2n \geq 512$ like SHA-512
(6) $H_b(z_1, z_2)$ = bit truncation of H_b from bit z_1 to bit z_2

Processing keys d, d_n, and Q to create determinants H_b of hash iteration number j_t within their bounds and security levels $i = x_t$ for each tier of multihash key, and then calculate the hash iteration number j_t and security level x_t of each tier t :

```
(1) for $t = 0$ to $m$, where $m = 9$ or other values,
        $H_b \leftarrow$ SHA-512 ($d \parallel d_n \parallel$ Q , 1) for one round of hash iteration ;
        $x_t \leftarrow H_b(8x+t\log_2 x , 8x+(t+1)\log_2 x-1) = H_b(8x+5t , 8x+5t+4)$ ;
        $i \leftarrow x_t$ ;
        if $i = 1$,           $j_t \leftarrow (b_1-2^8+1) + H_b(0 , 7)$,        $j_t \leq b_1$ ;
        else if $1 < i < x$, $j_t \leftarrow (b_{i-1}+1) + H_b(8i-8 , 8i-1)$,   $j_t \leq b_i$ ;
        else if $i = x$,      $j_t \leftarrow (b_{x-1}+1) + H_b(8x-8 , 8x-1)$, $j_t \leq b_x$ ;

        Generate intermediate slave keys $H_t$ for tier $t$ :
        if ID = URL = NULL,    $H_{it} \leftarrow$ SHA-512($d$ , $j_t$) ;
        else if ID = NULL,     $H_{it} \leftarrow$ SHA-512($d \parallel$ URL , $j_t$) ;
        else if URL = NULL,    $H_{it} \leftarrow$ SHA-512($d \parallel$ ID , $j_t$) ;
        else if ID and URL are not NULL, $H_{it} \leftarrow$ SHA-512($d \parallel$ ID $\parallel$ URL , $j_t$) ;
        $d_n \leftarrow H_t \leftarrow H_{it}(0 , 255)$, $n$-bit truncation of $H_{it}$ from MSB bit ; $t \leftarrow t + 1$;
   Go to step (1) of Box **1702** whenever $t \leq m$ . end
```

Generate slave key d_s :
(1) User selects $H \leftarrow H_m$ or $H \leftarrow$ SHA-512($H_0 \parallel H_1 \parallel H_2 \parallel \ldots \parallel H_t \parallel \ldots \parallel H_m$)
(2) $d_s \leftarrow Bin2Txt$(n-bit CSPRBG (H(0 , 255))), $Bin2Txt$ = Binary-to-text encoding

Jot down Q or store Q at a local/remote server for future access. Apply the slave key d_s. Then, clear the memory storing all forms of secrets and close all the application software.

Figure 7.5 First variant of multihash key method and system to support more offline accounts using automatically selected tiers and security levels

Sequence ID Q can be in plaintext and is used to create multiple unique offline and online slave keys. Q can be jotted down into a notebook, or stored at local and remote servers for future acknowledgment to the user about the Q value of one's account. Optional entries are username ID, domain name URL, or else NULL. Bounds of hash iteration for various security levels s$_i$ are b_1, b_2, b_3, …, b_i, …, b_x. Concatenation of ($d \parallel d_n \parallel$ Q) selects security level s_i among x security levels, where x = 20, 32 or others. This method uses $2n$-bit hash function, where $2n \geq 512$ like SHA-512. $H_b(z_1 , z_2)$ means bit truncation of H_b from bit z_1 to bit z_2.

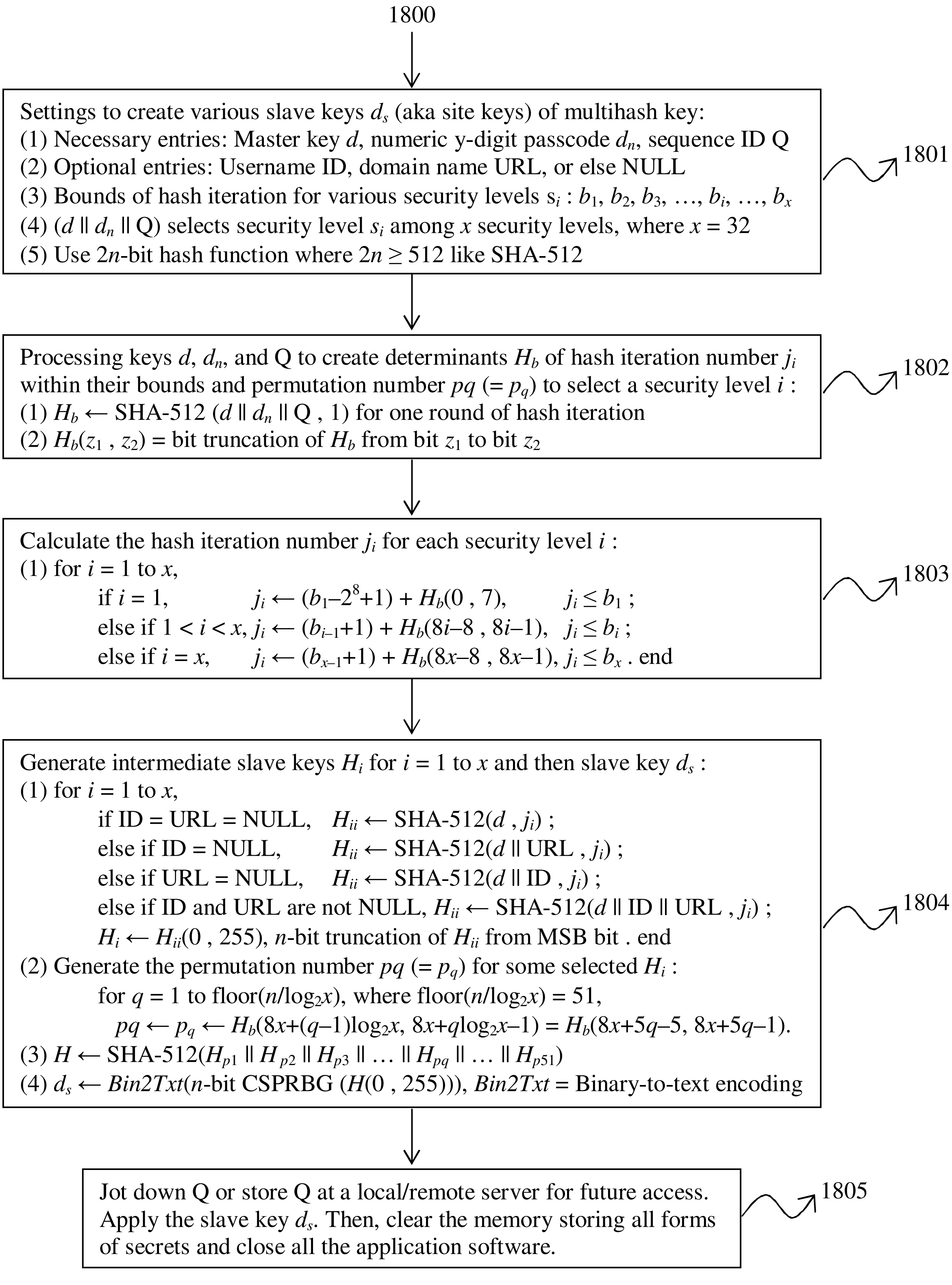
At Box **1702**, master key d, passcode d_n, and sequence ID Q are processed to create the determinants H_b of hash iteration number j_t within their bounds and security levels $i = x_t$ for each tier of multihash key, and then calculate the hash iteration number j_t and security level x_t of each tier t. Here, an intermediate slave key H_t is derived at each tier and replaces the d_n. Repeat step (1) in Box **1702** whenever the maximum number of tier m has not been reached.

At Box **1703**, final slave key d_s is generated by directly taking the slave key at the final tier or hashing the concatenation of derived secrets from each tier. At Box **1704**, jot down Q or store Q at a remote server as like salt for future access, apply the slave key d_s, clear the memory storing all forms of secrets, and then close all the application software. The passcode here can be optionally replaced by a big memorizable secret for more randomness to support more offline accounts up to Equation (7.5). Security level x can be increased up to the maximum of hash iteration number $j_{\max}$. Also, hash functions beyond 512 bits like 768 and 1024 bits may be needed.

$$S_{AC1} = x^m \tag{7.5}$$

7.5 Second Variant of Multihash Key Using Automatically Selected Permutation Sequence of Security Levels

The second variant in Figure 7.6 also supports more offline accounts by using automatically selected permutation sequence of security levels.



Settings to create various slave keys d_s (aka site keys) of multihash key:
(1) Necessary entries: Master key d, numeric y-digit passcode d_n, sequence ID Q
(2) Optional entries: Username ID, domain name URL, or else NULL
(3) Bounds of hash iteration for various security levels $s_i : b_1, b_2, b_3, \ldots, b_i, \ldots, b_x$
(4) ($d \| d_n \|$ Q) selects security level s_i among x security levels, where $x = 32$
(5) Use $2n$-bit hash function where $2n \geq 512$ like SHA-512

Processing keys d, d_n, and Q to create determinants H_b of hash iteration number j_i within their bounds and permutation number pq (= p_q) to select a security level i :
(1) $H_b \leftarrow$ SHA-512 ($d \| d_n \|$ Q , 1) for one round of hash iteration
(2) $H_b(z_1 , z_2)$ = bit truncation of H_b from bit z_1 to bit z_2

Calculate the hash iteration number j_i for each security level i :

```
(1) for $i = 1$ to $x$,
        if $i = 1$,             $j_i \leftarrow (b_1 - 2^8 + 1) + H_b(0 , 7)$,            $j_i \leq b_1$ ;
        else if $1 < i < x$, $j_i \leftarrow (b_{i-1} + 1) + H_b(8i-8 , 8i-1)$,   $j_i \leq b_i$ ;
        else if $i = x$,        $j_i \leftarrow (b_{x-1} + 1) + H_b(8x-8 , 8x-1)$, $j_i \leq b_x$ . end
```

Generate intermediate slave keys H_i for $i = 1$ to x and then slave key d_s :

```
(1) for $i = 1$ to $x$,
        if ID = URL = NULL,    $H_{ii} \leftarrow$ SHA-512($d$ , $j_i$) ;
        else if ID = NULL,         $H_{ii} \leftarrow$ SHA-512($d \|$ URL , $j_i$) ;
        else if URL = NULL,     $H_{ii} \leftarrow$ SHA-512($d \|$ ID , $j_i$) ;
        else if ID and URL are not NULL, $H_{ii} \leftarrow$ SHA-512($d \|$ ID $\|$ URL , $j_i$) ;
        $H_i \leftarrow H_{ii}(0 , 255)$, $n$-bit truncation of $H_{ii}$ from MSB bit . end
(2) Generate the permutation number $pq$ (= $p_q$) for some selected $H_i$ :
        for $q = 1$ to floor($n/\log_2 x$), where floor($n/\log_2 x$) = 51,
            $pq \leftarrow p_q \leftarrow H_b(8x+(q-1)\log_2 x,\ 8x+q\log_2 x-1) = H_b(8x+5q-5,\ 8x+5q-1)$.
(3) $H \leftarrow$ SHA-512($H_{p1} \| H_{p2} \| H_{p3} \| \ldots \| H_{pq} \| \ldots \| H_{p51}$)
(4) $d_s \leftarrow Bin2Txt$($n$-bit CSPRBG ($H$(0 , 255))), $Bin2Txt$ = Binary-to-text encoding
```

Jot down Q or store Q at a local/remote server for future access. Apply the slave key d_s. Then, clear the memory storing all forms of secrets and close all the application software.

Figure 7.6 Second variant of multihash key method and system to support more offline accounts using automatically selected permutation sequence of security levels

Box **1801** gives the settings to create various slave keys d_s (aka site keys) of multihash key. Necessary entries are master key d, numeric y-digit passcode d_n, where y can be 4, and sequence ID Q. Sequence ID Q can be in plaintext and is used to create multiple unique offline and online slave keys. Q can be jotted down into a notebook, or stored at local and remote servers for future acknowledgment to the user about the Q value of one's account. Optional entries are username ID, domain name URL, or else NULL. Bounds of hash iteration for various security levels s_i are $b_1, b_2, b_3, \ldots, b_i, \ldots, b_x$. Concatenation of $(d \,||\, d_n \,||\, Q)$ selects security level s_i among x security levels, where x = 20, 32 or others. This method uses $2n$-bit hash function, where $2n \geq 512$ like SHA-512.

At Box **1802**, master key d, passcode d_n, and sequence ID Q are processed to create the determinants H_b of hash iteration number j_i within their bounds and permutation number pq (= p_q) to select a security level i. $H_b(z_1\ , z_2)$ means bit truncation of H_b from bit z_1 to bit z_2. At Box **1803**, calculate the hash iteration number j_i for each security level i.

At Box **1804**, generate intermediate slave keys H_i at each security level and then slave key d_s. To select H_i, permutation number p_q is used. The final slave key is the hashing of the concatenation of multiple H_i based on p_q. There may be a special permutation number meaning NULL value where no bitstream is concatenated. If all the selected H_i are NULL, then select another d_n and repeat all the steps.

At Box **1805**, jot down Q or store Q at a remote server as like salt for future access, apply the slave key d_s, clear the memory storing all forms of secrets, and then close all the application software. Let T be the maximum number of concatenated H_i based on p_q. The passcode here can be optionally replaced by a big memorizable secret for more randomness to support more offline accounts up to Equation (7.6). Security level x can be increased up to the maximum of hash iteration number $j_{\max}$. Also, hash functions beyond 512 bits like 768 and 1024 bits may be needed.

$$\mathrm{S_{AC2}} = \sum_{y=1}^{y=T} x^{y} \tag{7.6}$$

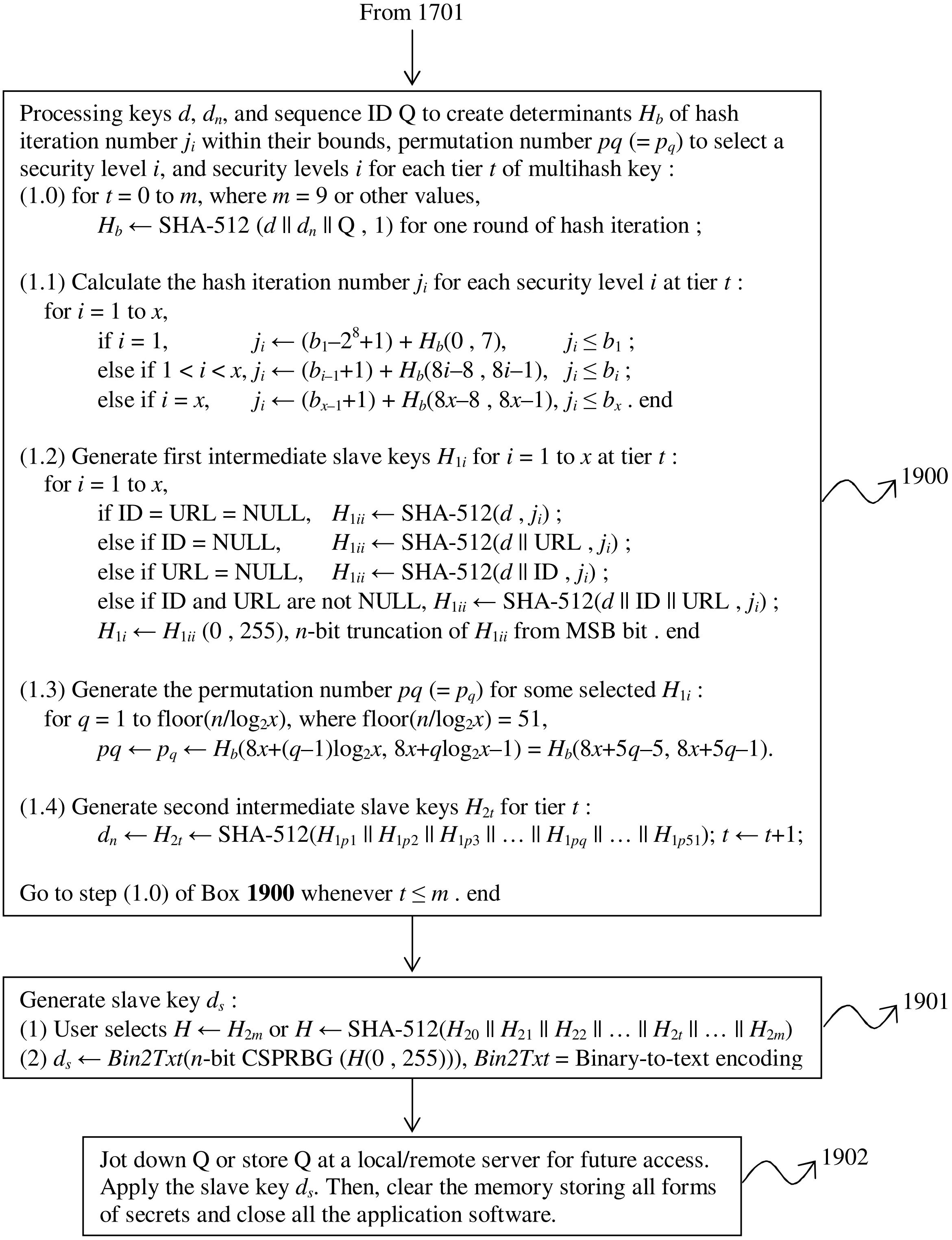
7.6 Third Variant of Multihash Key Using Hybrid Combination

The third variant in Figure 7.7 is a hybrid combination of the first and second variants to support more offline accounts using a hybrid combination of automatically selected tiers and security levels, and automatically selected permutation sequence of security levels.

Firstly, do the operations in Box **1701**. Then, at Box **1900**, master key *d*, passcode d_n, and sequence ID Q are processed to create the determinants H_b of hash iteration number j_i within their bounds, permutation number *pq* (= p_q) to select a security level *i*, and security levels *i* for each tier *t* of multihash key. Here, calculate the hash iteration number j_i for each security level *i* at tier *t*. Generate first intermediate slave keys H_{1i} for $i = 1$ to *x* at tier *t*. Generate the permutation number *pq* (= p_q) for some selected H_{1i} at tier *t*. Generate second intermediate slave keys H_{2t} for tier *t* and replaces the d_n. Repeat steps (1.0-1.4) in Box **1900** whenever the maximum number of tier *m* has not been reached. There may be a special permutation number meaning NULL value where no bitstream is concatenated. If all the selected H_i are NULL, then select another d_n and repeat all the steps.

At Box **1901**, final slave key d_s is generated by directly taking the slave key at the final tier or hashing the concatenation of derived secrets from each tier. At Box **1902**, jot down Q or store Q at a remote server as like salt for future access, apply the slave key d_s, clear the memory storing all forms of secrets, and then close all the application software. Sequence ID Q can be in plaintext and is used to create multiple unique offline and online slave keys. Q can be jotted down into a notebook, or stored at local and remote servers for future acknowledgment to the user about the Q value of one's account. Let T be the maximum number of concatenated H_{1i} based on p_q. The passcode here can be optionally replaced by a big memorizable secret for more randomness to support more offline accounts up to Equation (7.7). Security level *x* can be increased up to the maximum of hash iteration number $j_{\max}$. Also, hash functions beyond 512 bits like 768 and 1024 bits may be needed.

$$\mathrm{S}_{\mathrm{AC3}} = \left(\sum_{y=1}^{y=T} x^{y} \right)^{m} \tag{7.7}$$



Processing keys d, d_n, and sequence ID Q to create determinants H_b of hash iteration number j_i within their bounds, permutation number pq ($= p_q$) to select a security level i, and security levels i for each tier t of multihash key :

```
(1.0) for $t = 0$ to $m$, where $m = 9$ or other values,
        $H_b \leftarrow$ SHA-512 ($d \parallel d_n \parallel$ Q , 1) for one round of hash iteration ;

(1.1) Calculate the hash iteration number $j_i$ for each security level $i$ at tier $t$ :
   for $i = 1$ to $x$,
        if $i = 1$,            $j_i \leftarrow (b_1 - 2^8 + 1) + H_b(0 , 7)$,         $j_i \le b_1$ ;
        else if $1 < i < x$, $j_i \leftarrow (b_{i-1} + 1) + H_b(8i - 8 , 8i - 1)$,  $j_i \le b_i$ ;
        else if $i = x$,       $j_i \leftarrow (b_{x-1} + 1) + H_b(8x - 8 , 8x - 1)$, $j_i \le b_x$ . end

(1.2) Generate first intermediate slave keys $H_{1i}$ for $i = 1$ to $x$ at tier $t$ :
   for $i = 1$ to $x$,
        if ID = URL = NULL,   $H_{1ii} \leftarrow$ SHA-512($d$ , $j_i$) ;
        else if ID = NULL,       $H_{1ii} \leftarrow$ SHA-512($d \parallel$ URL , $j_i$) ;
        else if URL = NULL,    $H_{1ii} \leftarrow$ SHA-512($d \parallel$ ID , $j_i$) ;
        else if ID and URL are not NULL, $H_{1ii} \leftarrow$ SHA-512($d \parallel$ ID $\parallel$ URL , $j_i$) ;
        $H_{1i} \leftarrow H_{1ii}$ (0 , 255), $n$-bit truncation of $H_{1ii}$ from MSB bit . end

(1.3) Generate the permutation number $pq$ ($= p_q$) for some selected $H_{1i}$ :
   for $q = 1$ to floor($n/\log_2 x$), where floor($n/\log_2 x$) = 51,
        $pq \leftarrow p_q \leftarrow H_b(8x+(q-1)\log_2 x, 8x+q\log_2 x-1) = H_b(8x+5q-5, 8x+5q-1)$.

(1.4) Generate second intermediate slave keys $H_{2t}$ for tier $t$ :
        $d_n \leftarrow H_{2t} \leftarrow$ SHA-512($H_{1p1} \parallel H_{1p2} \parallel H_{1p3} \parallel \ldots \parallel H_{1pq} \parallel \ldots \parallel H_{1p51}$); $t \leftarrow t+1$;

Go to step (1.0) of Box 1900 whenever $t \le m$ . end
```

Generate slave key d_s :
(1) User selects $H \leftarrow H_{2m}$ or $H \leftarrow$ SHA-512($H_{20} \parallel H_{21} \parallel H_{22} \parallel \ldots \parallel H_{2t} \parallel \ldots \parallel H_{2m}$)
(2) $d_s \leftarrow$ *Bin2Txt*(n-bit CSPRBG (H(0 , 255))), *Bin2Txt* = Binary-to-text encoding

Jot down Q or store Q at a local/remote server for future access. Apply the slave key d_s. Then, clear the memory storing all forms of secrets and close all the application software.

Figure 7.7 Third variant of multihash key method and system to support more offline accounts using a hybrid combination of automatically selected tiers and security levels, and automatically selected permutation sequence of security levels

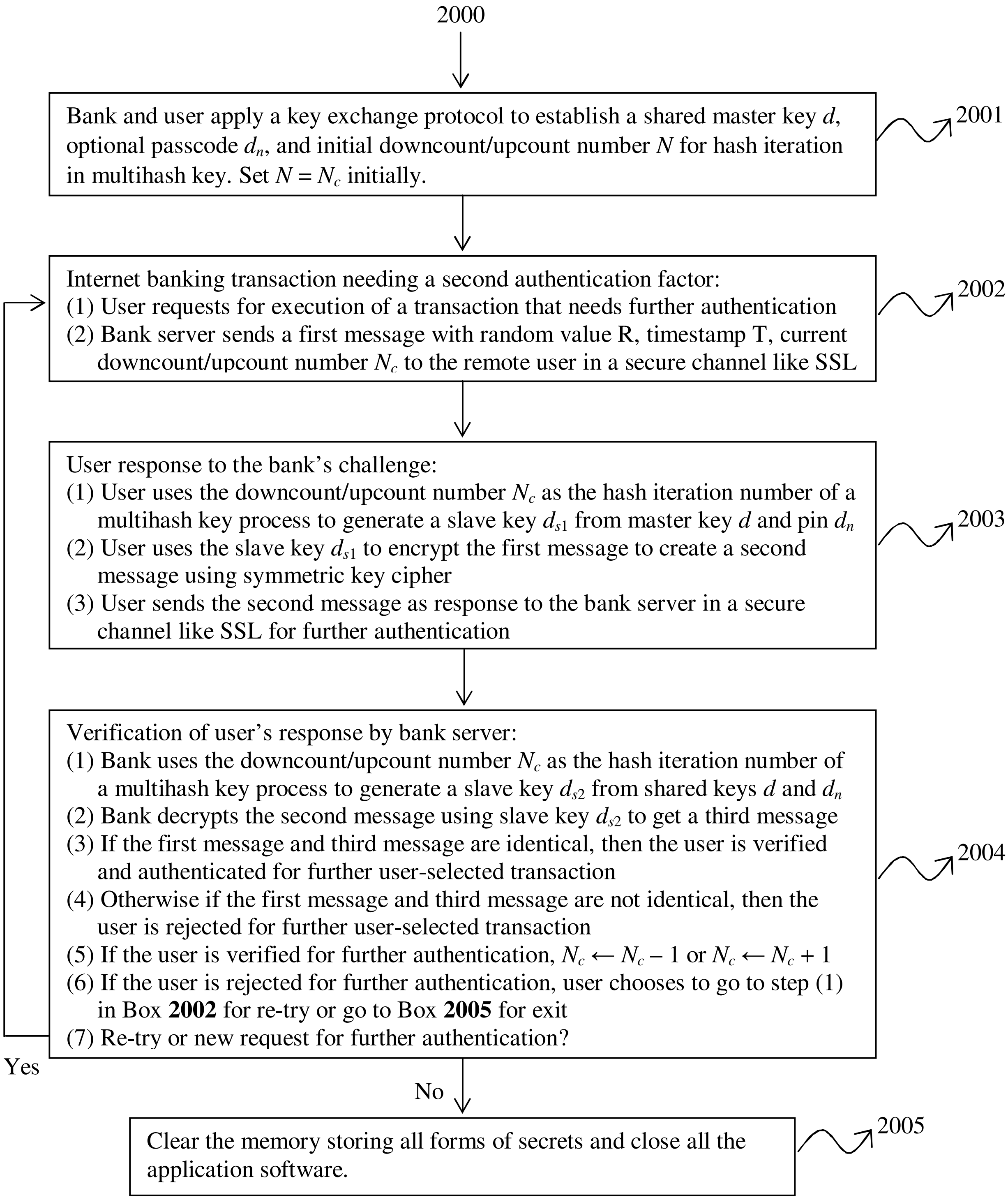
7.7 Fourth Variant of Multihash Key as a Further Authentication Factor

For the fourth variant in Figure 7.8, it is a specific application of multihash key to act as a further authentication factor in the Internet banking, online share trading, or other situations.

At Box **2001**, bank and user apply a key exchange protocol to establish a shared master key d, optional passcode d_n, and initial downcount/upcount number N for hash iteration in multihash key. Set $N = N_c$ initially. At Box **2002** for Internet banking transaction needing a second authentication factor, it is triggered by a user requesting for execution of a transaction that needs further authentication. Bank server then sends a first message with random value R, timestamp T, current downcount/upcount number N_c to the remote user in a secure channel like SSL.

At Box **2003** for user response to the bank's challenge, user uses the downcount/upcount number N_c as the hash iteration number of a multihash key process to generate a slave key d_{s1} from master key d and pin d_n. Then, user uses the slave key d_{s1} to encrypt the first message to create a second message using symmetric key cipher. Later, user sends the second message as response to the bank server in a secure channel like SSL for further authentication.

At Box **2004** for verification of user's response by bank server, bank uses the downcount/upcount number N_c as the hash iteration number of a multihash key process to generate a slave key d_{s2} from shared keys d and d_n. Then, bank decrypts the second message using slave key d_{s2} to get a third message. If the first message and third message are identical, then the user is verified and authenticated for further user-selected transaction. Otherwise if the first message and third message are not identical, then the user is rejected for further user-selected transaction. If the user is verified for further authentication, decrement the N_c by one unit for downcount, or increment the N_c by one unit for upcount. If the user is rejected for further authentication, user chooses to go to step (1) in Box **2002** for re-try or go to Box **2005** for exit. For re-try or new request for further authentication, go to step (1) in Box **2002**. Otherwise, go to Box **2003** to clear the memory storing all forms of secrets and close all the application software.



Bank and user apply a key exchange protocol to establish a shared master key *d*, optional passcode d_n, and initial downcount/upcount number *N* for hash iteration in multihash key. Set $N = N_c$ initially.

Internet banking transaction needing a second authentication factor:
(1) User requests for execution of a transaction that needs further authentication
(2) Bank server sends a first message with random value R, timestamp T, current downcount/upcount number N_c to the remote user in a secure channel like SSL

User response to the bank's challenge:
(1) User uses the downcount/upcount number N_c as the hash iteration number of a multihash key process to generate a slave key d_{s1} from master key *d* and pin d_n
(2) User uses the slave key d_{s1} to encrypt the first message to create a second message using symmetric key cipher
(3) User sends the second message as response to the bank server in a secure channel like SSL for further authentication

Verification of user's response by bank server:
(1) Bank uses the downcount/upcount number N_c as the hash iteration number of a multihash key process to generate a slave key d_{s2} from shared keys *d* and d_n
(2) Bank decrypts the second message using slave key d_{s2} to get a third message
(3) If the first message and third message are identical, then the user is verified and authenticated for further user-selected transaction
(4) Otherwise if the first message and third message are not identical, then the user is rejected for further user-selected transaction
(5) If the user is verified for further authentication, $N_c \leftarrow N_c - 1$ or $N_c \leftarrow N_c + 1$
(6) If the user is rejected for further authentication, user chooses to go to step (1) in Box **2002** for re-try or go to Box **2005** for exit
(7) Re-try or new request for further authentication?

Clear the memory storing all forms of secrets and close all the application software.

Figure 7.8 Fourth variant of multihash key method and system for the specific application to act as a further authentication factor in the Internet banking or other situations

7.8 Fifth Variant of Multihash Key as a Simple Key Escrow Method and System

The fifth variant in Figure 7.9 is another specific application of multihash key, where it acts as a simple key escrow method and system for supervisor-wise non-critical secrets.

<table>
<tr><th colspan="2">Simple Key Escrow Method and System Using Key Management of Multihash Key for Supervisor-wise Non-critical Secrets</th></tr>
<tr><td colspan="2">K_{GM} = grandmaster key, SID = staff identity number, Y = current year,
K_{SM} = staff master key, K_{SS} = staff slave key, CID = client identity number,
EID = event identity number, K_{CS} = client slave key,
Multihash = multihash key function as in Figures 7.2, 7.5-7.7.</td></tr>
<tr><td><b>Generation of Staff Slave Keys</b>
Supervisor holds key K_{GM}
$K_{SS} \leftarrow \text{Multihash} (K_{GM} \| \text{SID} \| \text{EID} \| \text{Y})$
Both supervisor and staff know key K_{SS}.</td><td><b>Generation of Client Slave Keys</b>
$K_{SM} \leftarrow K_{SS}$
$K_{CS} \leftarrow \text{Multihash} (K_{SM} \| \text{CID} \| \text{EID} \| \text{Y})$
Both staff and client know key K_{CS}.</td></tr>
<tr><td colspan="2"><b>Key Escrow</b>
Slave keys and master keys at a lower key management levels are known to people holding master keys and grandmaster keys, respectively, at a higher management level.</td></tr>
</table>

Figure 7.9 Fifth variant of multihash key method and system for the specific application to act as a simple key escrow method and system for supervisor-wise non-critical secrets

Key management of multihash key is applied here. Slave keys and master keys at a lower key management levels are known to people holding master keys and grandmaster keys, respectively, at a higher management level. For the generation of staff slave keys, a supervisor holding grandmaster key K_{GM} uses the staff identity number SID, event identity number EID, and current year Y, to generate staff slave keys K_{SS} from multihash key for different applications, where $K_{SS} \leftarrow \text{Multihash} (K_{GM} \| \text{SID} \| \text{EID} \| \text{Y})$.

A staff stores all one's staff slave keys into one's password vault. For the generation of client slave keys, a staff slave key becomes a staff master key K_{SM}.

K_{SM} is used together with client identity number CID, event identity number EID, and current year Y to generate client slave keys from multihash key again for different applications, where $K_{CS} \leftarrow$ Multihash (K_{SM} || CID || EID || Y). A client stores all one's client slave keys into one's password vault. In this way, the higher management people have escrowed the slave keys at the lower levels. This approach can be used for supervisor-wise non-critical secrets but confidential to the external parties.

7.9 Discussions on Variants of Multihash Key

Variants 1, 2, and 3 optionally require the passcode to work automatically or are upgraded to become a big memorizable secret created as in Figure 3.1. After the passcode has been replaced by a big memorizable secret with at least 128 bits, the sequence ID Q can be optionally used to make the generated slave keys unique.

Yet in the current Internet banking, a random number in an SMS (Short Message Service) through mobile phone network, or a one-time-password token (OTP token), like RSA SecurID token, is used as a second authentication factor.

Meanwhile, variant 4 alternatively uses downcounting or upcounting of hash iteration number to generate various slave keys from a master key to function as the second authentication factor. Lastly, variant 5 is designed for the key management of supervisor-wise non-critical secret in an organization like government, company, university, and school, to function as a simple key escrow method and system.

CHAPTER 8 MULTIHASH SIGNATURE

Since the introduction of classical digital signature scheme in 1976, many variants are created, such as blind signature, multisignature, group-oriented signature, threshold signature, etc. These variants perform certain specific functions of digital signature. Here, a new variant called multihash signature (aka object-designated signature) is proposed, where a single message can have multiple digital signatures from a single asymmetric key pair. This is done so by having a few rounds of hashing to produce multiple hashes, which are then signed by a private key to generate multiple unique digital signatures. All of these digital signatures can be verified using the same public key. This variant has some new message management functions, which are to trace a file downloaded from different mirror sites, to referee an advertiser broadcasting the news of a sponsor, and to monitor the leaking source that publicly discloses a classified digital file such as will, contract, form, etc.

8.1 Introduction

Digital signature scheme (Simmons, 1984; NIST, 2000, 2006c; Lyons-Burke, 2000; Atreya, Hammond, Paine, Starrett & Wu, 2002) was firstly conjectured by Whitfield Diffie and Martin Hellman (1976). Then Ronald Lorin Rivest, Adi Shamir and Leonard Max Adleman (1978, 1983) realized the first digital signature scheme using RSA algorithm, which is a type of integer factorization cryptography (IFC). Subsequently, this classical digital signature scheme has a lot of variants carrying out certain specific functions.

These variants are blind signature (Chaum, 1982, 1988), multisignature (Itakura & Nakamura, 1983; Boyd, 1989; Harn & Kresler, 1989), group-oriented signature (Desmedt, 1987), undeniable signature (Chaum & van Antwerpen, 1989), threshold signature (Desmedt & Frankel, 1989), fail-stop signature (Pfitzmann & Waidner, 1990; Pfitzmann, 1996), group signature (Chaum & van Heyst, 1991), proxy signature (Mambo, Usuda & Okamoto, 1996), signcryption (Zheng, 1997), and forward-secure signature (Bellare & Miner, 1999).

Cryptographic hash function is used in digital signature scheme to process a specific digital message with variable length, and produce a unique fixed-length bit stream called hash (Dang, 2007a, 2007b; Wikipedia Contributors, 2008o, 2008w) to represent the message. Since hash is like a summary of message, it is also known as message digest and digital fingerprint. Hash function is a one-way function with the properties of preimage resistance and collision resistance. MD5 (Rivest, 1992; Wikipedia Contributors, 2008r), SHA-1 (Eastlake & Jones, 2001; NIST, 2002b; Eastlake & Hansen, 2006; Wikipedia Contributors, 2008w) and RIPEMD-160 are the most popular hash functions as of 2005 (Wikipedia Contributors, 2008o). They are built based on the architecture of MD4. MD5 has a digest size of 128 bits and is no longer sufficient for the minimum requirement of 160 bits by 2010. SHA-1 has been compromised (Wang, Yin & Yu, 2005) and this hastens NIST to propose the transition to SHA-2. SHA-2 consists of four variants: SHA-224, SHA-256, SHA-384 and SHA-512 (NIST, 2002b; Lilly, 2004; Eastlake & Hansen, 2006).

One of the many applications of secret is to assign a particular message with particular object like meaning, function, or recipient. For instance (Cox, Miller, Bloom & Fridrich, 2006), to prevent and trace the public disclosure of government documents by the press, Margaret Thatcher, who was British former Prime Minister in the 1980s, inserted certain unique number of white spaces (aka blanks) as secret in documents distributed to different cabinet ministers, so as to identify the document recipients who have disclosed the documents to the press. This is a type of covert text watermarking with recipient-designated message. Recipients of cabinet ministers here are designated objects for the message of distributed government documents.

Likewise, the secret of blanks can be used to represent other objects like specific meaning and function. Anonymity and non-repudiation are two of its not yet well-established requirements. Comparing with watermarking, digital signature has stronger security strength in terms of randomness, integrity, and robustness.

Nevertheless, so far there is no object-designated message using digital signature scheme. Hence, there exists a need to create object-designated signature scheme with optional properties of anonymity and non-repudiation called "multihash signature". There are a few applications of hash function. Here, multiple hashes are processed to be generated from a single message. This is possible by using different rounds of hash iteration. These unique hashes are then signed by a private key to create multiple digital signatures that can be verified using the same public key. This new variant of classical digital signature scheme has a few new functions in term of better message management and control.

8.2 Classical Digital Signature Scheme

In a classical digital signature scheme, a message is hashed only once. Then, a sender signs or encrypts this generated hash using his private key to produce a digital signature. The message and digital signature are sent to one or more receivers. To verify a digital signature, a receiver hashes the message to produce hash H_1 and decrypts the digital signature using the sender's public key to get H_2. If H_1 is the same with H_2, then the digital signature is verified; or else, it is rejected. The pseudocode of this classical scheme is shown in Figure 8.1.

```
0.0 Initialization
    0.1 Sender has private key $K_{Pte}$ and public key $K_{Pub}$.
    0.2 There may be one or more receivers.
1.0 Signing a message
    1.1 Sender hashes a message using hash function to get a hash H.
    1.2 Sender signs or encrypts the hash H using $K_{Pte}$ to get a digital signature S.
    1.3 Sender sends the message and digital signature S to one or more receivers.
2.0 Verifying a message
    2.1 A receiver hashes the received message using hash function to get a hash $H_1$.
    2.2 Receiver decrypts the digital signature S using $K_{Pub}$ to get a hash $H_2$.
    2.3 Compare $H_1$ and $H_2$
        2.3.1 If $H_1$ is the same with $H_2$, the digital signature S is verified;
        2.3.2 else if $H_1$ is not the same with $H_2$, the digital signature S is rejected.
```

Figure 8.1 Pseudocode of the classical digital signature scheme

8.3 Applications of Hashing

Hash function has a few applications. It is used to hash a password or key to form a hashed password to be stored in a computer for future authentication purpose.

In another password authentication scheme, hashing is used in password-authenticated key exchange (PAKE) like SPEKE and SRP-6. Hashing is also used to process a digital file to get a fixed-length bit stream uniquely representing this digital file. It has the cryptographic properties of integrity and confidentiality in this case. Any change to the message can change the hash value. Disclosing a message digest does not reveals the contents of a message, but proves that an owner has this message at an earlier time.

The message digest encrypted by a sender's private key produces a digital signature that can be verified using the sender's public key. This is the classical digital signature scheme. Further application of this classical scheme gives birth to digital timestamping scheme (Haber & Stornetta, 1990, 1991). On the other hand, hashing can be used for pseudorandom number generation as well.

There are some special applications of hashing. Key strengthening, which is also called key stretching, is one of them (Kelsey, Schneier, Hall & Wagner, 1997). A message like password or key is hashed many rounds. It is used to make a weak key stronger by setting the key processing time at a maximum of one second. Some variants of this key strengthening allow a master key to be concatenated with website name to produce multiple site keys (aka slave keys) for different offline and online accounts. There may be one or two levels of key strengthening. Some of the examples are LPWA (Lucent Personal Web Assistant), HP Site Password, CPG (Compass Password Generator), Password Multiplier, SPP (Single Password Protocol), PwdHash, Passpet, and Multihash Key (Lee & Ewe, 2007a).

For our proposed digital signature scheme, the similar concept used in multihash key is applied again. Instead of generating multiple hashes from a master key, this time multiple hashes are created from a single message. This is the main reason why this signature variant is called multihash signature.

8.4 Proposed Method: Single Message with Multiple Digital Signatures from Single Asymmetric Key Pair

In multihash signature scheme, majority of the steps in classical digital signature scheme are maintained. The main difference is that a message is hashed using different number of hash iteration to produce multiple hashes. These hashes or message digests are unique among themselves. When these message digests are encrypted or signed using a sender's private key, different digital signatures are created, but all the produced digital signatures can be verified using the sender's public key. Figure 8.2 illustrates the pseudocode of this new signature variant.

```
0.0 Initialization
    0.1 Sender has private key $K_{Pte}$ and public key $K_{Pub}$.
    0.2 There may be one or more receivers with a maximum.
    0.3 Sender keeps a table matching the numbers of hash iteration N to every
        receiver.
1.0 Signing a message
    1.1 Sender hashes a message using hash function for N rounds to get a hash $H_N$.
    1.2 Sender signs or encrypts the hash $H_N$ using $K_{Pte}$ to get a digital signature $S_N$.
    1.3 Sender sends the message and digital signature $S_N$ to receiver $R_N$.
2.0 Verifying a message
    2.1 Receiver $R_N$ hashes the received message for N rounds to get a hash $H_{N1}$.
    2.2 Receiver decrypts the digital signature $S_N$ using $K_{Pub}$ to get a hash $H_{N2}$.
    2.3 Compare $H_{N1}$ and $H_{N2}$
        2.3.1 If $H_{N1}$ is the same with $H_{N2}$, the digital signature $S_N$ is verified;
        2.3.2 else if $H_{N1}$ is not the same with $H_{N2}$, the digital signature $S_N$ is rejected.
```

Figure 8.2 Pseudocode of the proposed multihash signature scheme

The sender sends the message and one of the digital signatures to every receiver. Each receiver shall receive the same message but different digital signature. Every digital signature has a different number of hash iterations. This hash iteration number is concatenated to the digital signature file to ease the receiver in producing a message digest for verification purpose. Every receiver has a different hash iteration number, and they are indexed using this number by the sender in order to identify them. The sender keeps a table that matches the hash iteration numbers to the identities of receivers.

From the experiments on number of hash iterations with maximum one-second message processing time, the following results for two sets of computer systems are obtained. For the first computer system, it is a desktop PC with Pentium

II 266MHz 192MB RAM running on Windows XP Professional Edition. The upper bound for one-second hash iteration is 8200. It means 8192 (= 2^{13}) receivers can be supported using a 13-bit string to be concatenated to the digital signature.

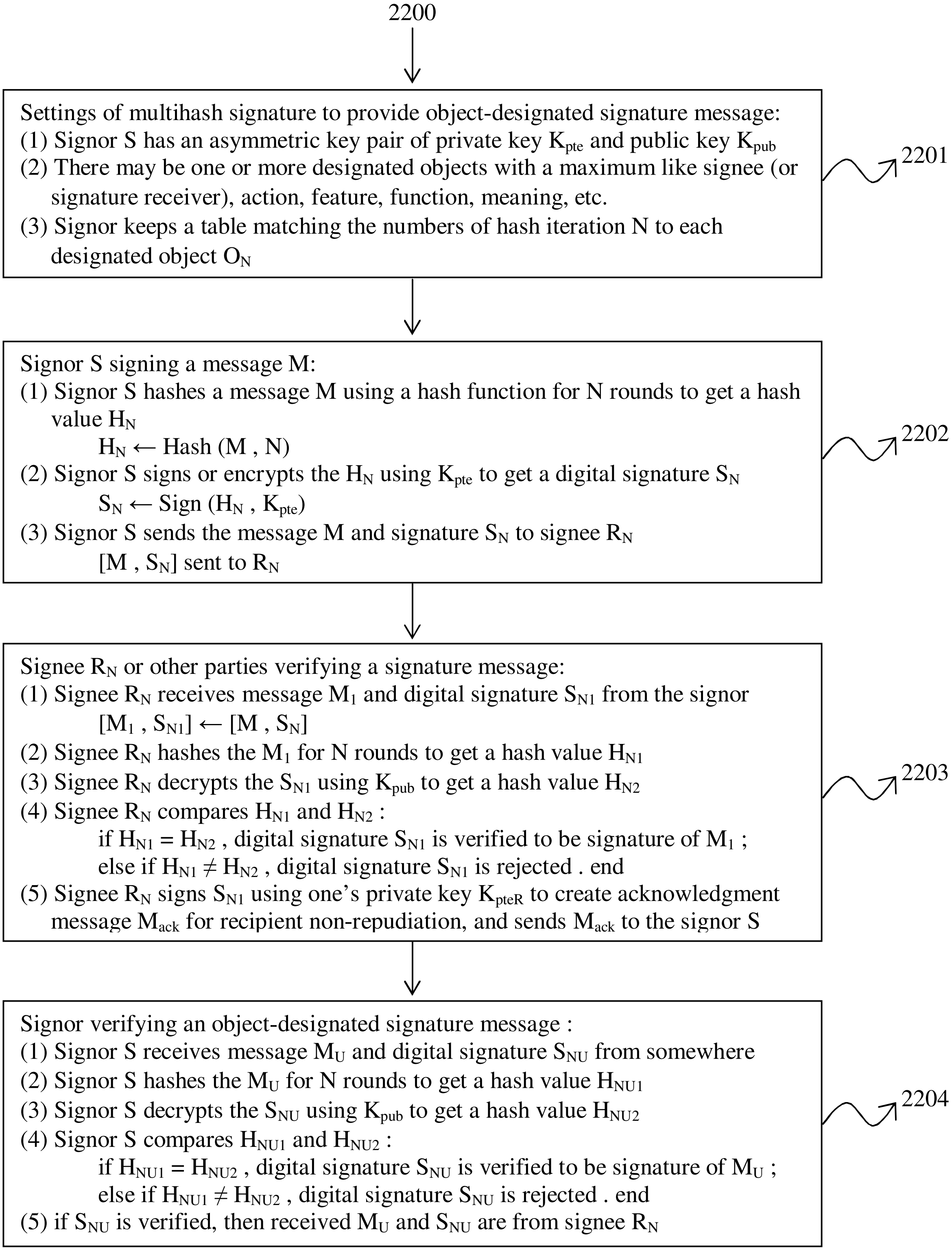
For the second computer system, it is a laptop PC with Centrino Duo 1.66GHz 1.5GB RAM running on Windows XP Home Edition. Its upper bound for one-second hash iteration is 93700. In other words, 65536 (= 2^{16}) receivers can be supported using a 16-bit string. Yet for the third computer system, it is a desktop PC with Pentium D 2.80GHz 512MB RAM running on Windows XP Professional Edition SP3. Its upper bound for one-second hash iteration is 122,500. In other words, 65536 (= 2^{16}) receivers can be supported using a 16-bit string.

Better computer systems with better performance can support larger number of receivers. However, to have one-second message processing for all the computer systems, the oldest computer system that is still in the market has to be the main parameter in setting the maximum number of receivers.

In short, multihash signature method and system can provide object-designated signature message with specific meaning, function, or recipient as illustrated in Figure 8.3. A message is hashed iteratively for variable rounds by a signor, and later signed using signor's asymmetric private key to generate a new type of digital signature. This new digital signature only differs from the conventional digital signature in the aspect that it carries the information of hash iteration number as well. In other words, a message can have multiple digital signatures from an asymmetric key pair, and each hash iteration number can be designated for any object, action, feature, function, meaning, recipient, etc., as a representation. Here, the signor keeps a table matching the hash iteration number and its represented object.

8.5 Comparisons for Advantages

Table 8.1 shows the main differences between the classical and multihash signature schemes. The proposed variant has the same security strength as classical scheme. The key difference is a message is hashed for one round in classical digital signature scheme; whereas a message is hashed for variable round in multihash-based digital signature scheme. This allows our signature variant to have variable number of digital signatures per message, where classical scheme has one digital signature per message.



Settings of multihash signature to provide object-designated signature message:
(1) Signor S has an asymmetric key pair of private key K_{pte} and public key K_{pub}
(2) There may be one or more designated objects with a maximum like signee (or signature receiver), action, feature, function, meaning, etc.
(3) Signor keeps a table matching the numbers of hash iteration N to each designated object O_N

Signor S signing a message M:
(1) Signor S hashes a message M using a hash function for N rounds to get a hash value H_N
$H_N \leftarrow$ Hash (M , N)
(2) Signor S signs or encrypts the H_N using K_{pte} to get a digital signature S_N
$S_N \leftarrow$ Sign (H_N , K_{pte})
(3) Signor S sends the message M and signature S_N to signee R_N
[M , S_N] sent to R_N

Signee R_N or other parties verifying a signature message:
(1) Signee R_N receives message M_1 and digital signature S_{N1} from the signor
[M_1 , S_{N1}] $\leftarrow$ [M , S_N]
(2) Signee R_N hashes the M_1 for N rounds to get a hash value H_{N1}
(3) Signee R_N decrypts the S_{N1} using K_{pub} to get a hash value H_{N2}
(4) Signee R_N compares H_{N1} and H_{N2} :
if $H_{N1} = H_{N2}$, digital signature S_{N1} is verified to be signature of M_1 ;
else if $H_{N1} \neq H_{N2}$, digital signature S_{N1} is rejected . end
(5) Signee R_N signs S_{N1} using one's private key K_{pteR} to create acknowledgment message M_{ack} for recipient non-repudiation, and sends M_{ack} to the signor S

Signor verifying an object-designated signature message :
(1) Signor S receives message M_U and digital signature S_{NU} from somewhere
(2) Signor S hashes the M_U for N rounds to get a hash value H_{NU1}
(3) Signor S decrypts the S_{NU} using K_{pub} to get a hash value H_{NU2}
(4) Signor S compares H_{NU1} and H_{NU2} :
if $H_{NU1} = H_{NU2}$, digital signature S_{NU} is verified to be signature of M_U ;
else if $H_{NU1} \neq H_{NU2}$, digital signature S_{NU} is rejected . end
(5) if S_{NU} is verified, then received M_U and S_{NU} are from signee R_N

Figure 8.3 Multihash signature method and system to provide object-designated signature message

Table 8.1 Comparisons of classical and multihash-based digital signature schemes

Difference	Classical	Multihash Signature
Number of hash iterations	One	Variable
Number of digital signatures per message	One	Variable
To trace a file downloaded from different mirror sites	No	Yes
To referee an advertiser broadcasting the news of a sponsor	No	Yes
To monitor the leaking source that publicly discloses a classified digital file	No	Yes
Security compared with classical digital signature scheme	-	Same

These two main differences create three further differences that act as new functions to digital signature scheme in term of message management and control. Firstly, this proposed signature variant can trace a file downloaded from different mirror sites. For example, current executables in some websites are EXE files listed together with .SIG files. .SIG file is a digital signature file format that allows users who have downloaded the executable to verify the integrity of this executable. Normally, there is more than one link for download, and these links are sharing the same .EXE and .SIG. It would be better if we have a single .EXE with different .SIG files for different mirror sites. An administrator can create a table matching the hash indices with respective mirror sites in the process of creating the multiple digital signatures. This allows the administrator to trace the mirror site of an executable.

Secondly, this signature variant can referee an advertiser broadcasting the news of a sponsor. Let say someone acting as a sponsor wants to advertise news like job recruitment, where an advertiser acting as a broadcaster of this news can earn an allowance if an audience is successfully acquired to certain degree like interviewed or recruited. If this sponsor does not want the information of the advertiser to be displayed in the message file, then the sponsor can opt to have multiple digital signatures for a single advertisement message. The number of hash iterations in the digital signature file is matched to the identity of an advertiser. This matching table is kept by the sponsor. In short, a sponsor can trace the performance of each advertiser, who acts as a referee, by using an exactly unique message without having any reference number in the message, where any difference in the message file can cause ambiguity to the receivers like future employees and customers.

Thirdly, it can monitor the leaking source that publicly discloses a classified digital file such as will, contract, form, etc. For instance, in a will, the involved parties are will holder, lawyer(s), and witnesses. Every lawyer and every witness are issued a different set of digital signatures with different hash iteration number to identify them. The will holder keeps this matching table. Whenever there is a public leakage of the will before the death of the will holder or any allowed period, the will holder or any other party can identify who is the leaking source.

In short, advantages of multihash signature are designated recipient function to alternate with watermarking, object-designated meaning, referral function, anonymity support, avoidance of name clashing and renaming problems, stronger collision resistance than method using the hashing of the concatenation of message digest and object name like Equation (8.1), as well as recipient non-repudiation. The example of object-designated meaning is the cheque validity status including status like valid, invalid, paid, void, on hold, late processing, rejected, withdrawn, cancelled, etc. The examples of referral functions are to trace a file downloaded from different websites, to referee an advertiser broadcasting the news of a sponsor, and to monitor the leaking source that has publicly disclosed a classified digital file.

$$\text{Hash Value} = \text{Hash}(\text{Hash}(\text{Message}) \parallel \text{Object Name}) \qquad (8.1)$$

Here, multihash signature is used in some other inventions of this doctoral research project. One of them is called triple-watermark digital cheque and another is triple-watermark software licensing schemes in Chapter 10, together with MePKC, steganography, and watermarking. The security of multihash signature has the same strength with the conventional digital signature scheme. For higher security to trace the identity of an Internet user signing a message and one's Internet geographical region, a message is suggested to be hashed and concatenated with MAC address and/or IP address, and then undergoes an optional conventional digital signature or multihash signature as in Equation (8.2) where S = Signature.

$$S = \text{Multihash Signature (Hash (Message)} \parallel \text{MAC Address} \parallel \text{IP Address)} \quad (8.2)$$

8.6 Conclusion

In a nutshell, introducing variable round of hash iterations on a message can create new functions to digital signature scheme. Using multiple digital signatures from single message, which are sent to different receivers, can trace the file downloaded from some mirror sites, referee an advertiser broadcasting the advertisement of a sponsor, and monitor the leakage of any classified digital file. The digital signatures in this proposed signature variant called multihash signature are unique among themselves. However, they can be verified using single public key even though they are generated from single message.

CHAPTER 9 APPLICATIONS OF BIG SECRET & MePKC (PART 1)

9.1 Applications of Created Big Memorizable Secret(s)

For useful applications of the created big memorizable secret(s) and MePKC (Memorizable Public-Key Cryptography), they include:

(i) methods and systems to realize memorizable symmetric key the secret till resistance to quantum computer attack;

(ii) methods and systems to realize memorizable public-key cryptography (MePKC);

(iii) methods and systems to improve security strength of other cryptographic, information-hiding, and non-cryptographic applications of secret beyond 128 bits;

(iv) method and system to harden the identification of embedded data in steganography although stego-data has been detected;

(v) method and system to transfer fund electronically over a remote network using MePKC;

(vi) method and system to license software electronically over a remote network using MePKC;

(vii) methods and systems to authenticate human-computer and human-human communications at a local station or over a remote network using MePKC;

(viii) method and system to use digital certificate with more than one asymmetric key pair for different protection periods and password throttling;

(ix) method and system to use three-tier MePKC digital certificates for ladder authentication;

(x) method and system to store, manage, and download voice and video calls of mobile phone and wired phone at online distributed servers;

(xi) method and system of multipartite electronic commerce transactions; as well as

(xii) Method and system to boost up the trust level of MePKC digital certificate by using more than one certification authority (CA) and/or introducer of trust of web.

Applications (i-iii) are presented in this Chapter 9. Applications (iv-vi) are presented in Chapter 10. Applications (vii-ix) are explained in Chapter 11. For applications (x-xii), they are in Chapter 12.

To apply big memorizable secret(s) to the novel methods and systems using MePKC from (iv) to (xii), two more independent inventions are applied here to enhance the features of MePKC. These two inventions are multihash key and multihash signature (aka object-designated signature). Multihash key includes some methods and systems to generate multiple slave keys from a single master key as in Chapter 7. Meanwhile, multihash signature includes a method and system to generate object-designated signature message with specific feature, meaning, function, or recipient as in Chapter 8.

9.2 Memorizable Symmetric Key to Resist Quantum Computer Attack

Due to the successful cracking of 56-bit DES (Data Encryption Standard) in the 1990s, stronger symmetric ciphers with larger symmetric key sizes like 80-bit 2TDES, 112-bit 3TDES, as well as 128-, 192-, and 256-bit AES (developed from Rijndael cipher) are introduced to replace the DES.

Blaze, Diffie, Rivest, Schneier, Shimomura, Thompson and Wiener (1996) discussed the minimal key lengths for symmetric ciphers. The NIST (National Institute of Standards and Technology), USA, proposes different protection periods for security through years 2010, 2030, and beyond 2030, for 80, 112, and 128 bits, respectively (E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b). ECRYPT of

European Union (EU) proposes in its technical reports that 80-, 96-, 112-, 128-, and 256-bit security have protection periods of 4 years through year 2010, 10, 20, 30 years, and foreseeable future to be against quantum computer attack, respectively (Gehrmann & Näslund, 2005, 2006, 2007). Nevertheless, conventional methods and systems normally can only realize a key size of 128 bits or less.

Hence, the first application 9.1(i) of the present invention in applying the created big memorizable secret is to realize higher security levels of symmetric ciphers like AES-192 and AES-256. By using the methods and systems as in Figure 3.1 and Table 3.3, it can be observed that the current highest security level of symmetric cipher at 256 bits can be practically realized and achieved using big memorizable 256-bit secret. For another brief explanation, please visit Section 4.3.9.

9.3 Memorizable Public-Key Cryptography (MePKC)

9.3.1 Related Works: Storages of Private Key

For the current asymmetric key cryptosystem, a private key is normally encrypted using another symmetric key. The encrypted private key is stored in a local computing system or token; whereas the symmetric key is stored in the human brain. The present possible attacks for this method are guessing attack, dictionary attack, and pre-computation attack.

Another method is to split the private key into two or more portions (Ganesan, 1996b; Bishop, 2003, pp. 264-265). There are other literary works about split private key cryptography over here (Ganesan, 1996a, 1998a, 1998b, 1998c, 1999; Ganesan, Sandhu, Cottrell & Austin, 2006a, 2006b, 2006c; Ganesan, Sandhu, Cottrell, Schoppert & Bellare, 2006; Ganesan & Yacobi, 1996; Sandhu, deSa & Ganesan, 2003, 2005a, 2005b, 2005c, 2006a, 2006b, 2006c, 2006d, 2006e, 2006f; Sandhu, Schoppert, Ganesan, Bellare & deSa, 2006a, 2006b, 2006c, 2006d, 2006e, 2006f, 2007a, 2007b, 2007c, 2007d; Sandhu, Ganesan, Cottrell, Renshaw, Schoppert & Austin, 2007;). The first portion of the private key can be derived from a normal human-memorizable symmetric key. The other portions of the private key are stored as encrypted partial private key alike the normal encrypted private key. This method resists the pre-computation attack.

A third method is to store the encrypted private key in a server connected to a computer communication network (Baltzley, 2000, 2001a, 2001b). A user has the roaming capability where the encrypted private key can be downloaded from the server for decryption at anywhere. Proxy servers are needed for this method to avoid single point of failure. Its possible attacks are the same as encrypted private key stored in the local computing system.

9.3.2 The Proposed MePKC Appliacation 9.1(ii)

The second application 9.1(ii) of the present invention in applying the created big memorizable secret is to improve from the token-based public-key cryptography (PKC) to the realization of secret-based PKC using fully memorizable private key, which is named as MePKC (Memorizable Public-Key Cryptography) or MoPKC (Mobile Public-Key Cryptography) here. The main advantages of MePKC are full secret memorizability and mobility convenience. Yet another quite important advantage is that secret-based MePKC can resist some side-channel attacks vulnerable to token-based PKC, such as those attacks over the fully or partially encrypted private key. For illustration of MePKC, please refer to Figure 9.1. For another brief explanation, please visit Section 4.3.9.

The current lowest key size requirement of asymmetric private key is 160 bits operating in FFC and ECC. From Table 3.3 listing all the proposed novel methods and systems to create big memorizable secret, a 160-bit secret for 160-bit fully memorizable private key can be supported by self-created signature-like Han character for CLPW and CLPP, 2D key, multilingual key, and multi-tier geo-image key. This group of big memorizable secret creation method and system can easily support memorizable private key up to 256 bits at the symmetric bits of security strength of 128 bits and for a protection period of 30 years.

For higher security levels up to 512-bit secret used by 512-bit MePKC, multi-factor multimedia key using software token has to be adopted to halve the key size requirement towards a practical realization. Here, the mobility convenience is somehow sacrificed.

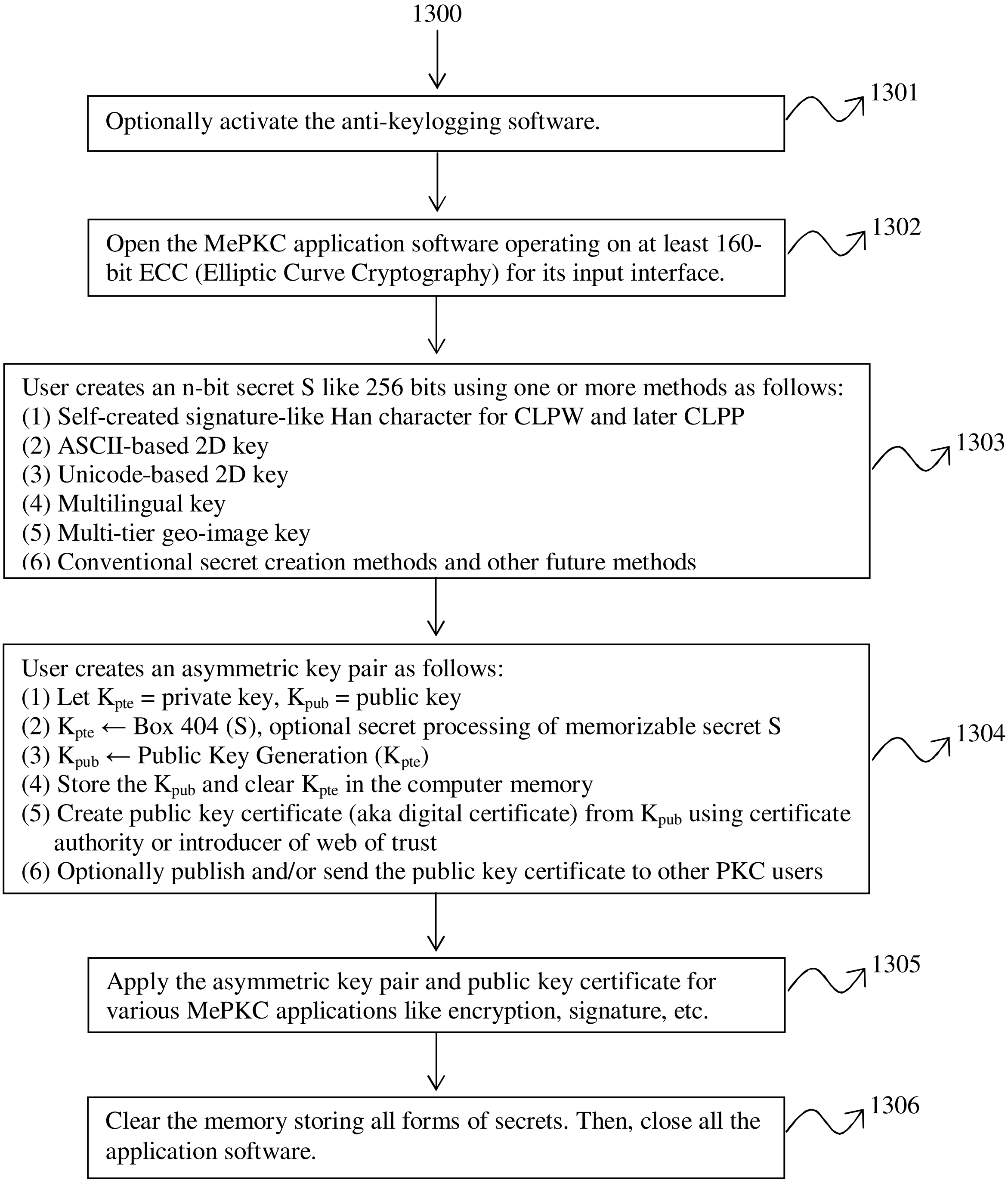

1300
Optionally activate the anti-keylogging software.
1301
Open the MePKC application software operating on at least 160-bit ECC (Elliptic Curve Cryptography) for its input interface.
1302
User creates an n-bit secret S like 256 bits using one or more methods as follows:
(1) Self-created signature-like Han character for CLPW and later CLPP
(2) ASCII-based 2D key
(3) Unicode-based 2D key
(4) Multilingual key
(5) Multi-tier geo-image key
(6) Conventional secret creation methods and other future methods
1303
User creates an asymmetric key pair as follows:
(1) Let K_{pte} = private key, K_{pub} = public key
(2) $K_{pte} \leftarrow$ Box 404 (S), optional secret processing of memorizable secret S
(3) $K_{pub} \leftarrow$ Public Key Generation (K_{pte})
(4) Store the K_{pub} and clear K_{pte} in the computer memory
(5) Create public key certificate (aka digital certificate) from K_{pub} using certificate authority or introducer of web of trust
(6) Optionally publish and/or send the public key certificate to other PKC users
1304
Apply the asymmetric key pair and public key certificate for various MePKC applications like encryption, signature, etc.
1305
Clear the memory storing all forms of secrets. Then, close all the application software.
1306

Figure 9.1 Operation of MePKC method and system

To generate this software token, firstly a big multimedia data file like random or non-random bit stream, text, image, audio, animation, or video, is hashed by a 2n-bit hash function to produce 2n-bit hash value. The 2n-bit hash value is encrypted by using an n-bit symmetric key and n-bit AES to further produce a software token.

Then, the multimedia data file is destroyed or hide at a safe location like safety box, and the software token is either stored in a local storage device like USB flash drive or in a remote server accessible through roaming network.

A user remembers only the n-bit secret of symmetric key. Whenever 2n-bit MePKC is needed for various applications, the software token is acquired and decrypted using the n-bit memorizable secret of symmetric key to obtain the 2n-bit hash value. This n-bit secret and 2n-bit hash value are then used to derive the 2n-bit MePKC private key.

The MePKC can be used for major PKC cryptographic applications like encryption and digital signature schemes. Other minor applied cryptographic schemes are key exchange, authentication, blind signature, multisignature, group-oriented signature, undeniable signature, threshold signature, fail-stop signature, group signature, proxy signature, signcryption, forward-secure signature, designated-verifier signature, public-key certificate (digital certificate), digital timestamping, copy protection, software licensing, digital cheque (aka electronic cheque), electronic cash, electronic voting, BAP (Byzantine Agreement Protocol), electronic commerce, MAC (Message Authentication Code), key escrow, online verification of credit card, multihash signature, etc.

The blind signature scheme includes its further applications for electronic cash (aka e-cash, electronic money, e-money, electronic currency, e-currency, digital cash, digital money, digital currency, or scrip), and electronic voting (aka e-voting, electronic election, e-election, electronic poll, e-poll, digital voting, digital election, or digital poll).

Advancement of computing technologies requests for longer key sizes for a fixed protection period. To freeze this unwanted request, key strengthening (aka key stretching) through many rounds of hash iteration, together with hash truncation and a hash function with longer hash value like 768, 1024 bits or more, can be used.

MePKC is extended to a novel claimed invention here called multihash signature scheme, and novel innovations of some cryptographic schemes like digital cheque, software licensing, human-computer and human-human authentication via a computer communications network, as well as MePKC digital certificate with multiple public keys for password throttling and ladder authentication.

These MePKC applications are best to be implemented using the ECC (Silverman, 1986; Blake, Seroussi & Smart, 1999, 2005; Hankerson, Menezes & Vanstone, 2004, 2005; Zhu & Zhang, 2006). This is because ECC needs a minimum private key size of 160 bits and it has been long time tested for its security strength. Alternatively, depending on further research and evaluation, shorter private key size at equivalent or better bits of security strength can be achieved by using hyperelliptic curve cryptography (HECC) (Pelzl, Wollinger & Paar, 2004; Cohen & Frey, 2006; Wang & Pei, 2006) and possibly other cryptosystems like torus-based cryptography (TBC) (Rubin & Silverberg, 2003).

For HECC, the genera 2 and 3 have so far been tested to have shorter key size requirement than ECC by twice and thrice. Between them, genus-2 HECC has a higher security without the demand to have a correction factor for its key size. In other words, the correction factor of HECC of genus 2 is 1. As information, genus-3 and genus-4 HECC have a correction factor of 1.05 and 1.286 times of its field, respectively, for the key size to get a larger group order at equivalent bits of security strength. For more information, please refer to an article entitled "High Performance Arithmetic for Special Hyperelliptic Curve Cryptosystems of Genus Two" by Jan Pelzl, Thomas Wollinger, and Christof Paar (2004).

9.4 Other Cryptographic, Information-Hiding, and Non-Cryptographic Applications of Secret beyond 128 bits

The third application 9.1(iii) of the present invention in applying the created big memorizable secret is various other cryptographic, information-hiding, and non-cryoptographic applications needing a big memorizable secret(s).

The other cryptographic applications include various PAKE (Password-Authenitcated Key Exchange) like SPEKE (Simple Password Exponential Key Exchange) (Jablon, 2006) and SRP-6 (Secure Remote Password Protocol version 6) (Wu, 2003).

Meanwhile, information-hiding applications (Petitcolas, Anderson & Kuhn, 1999; Moulin & O'Sullivan, 2003) include stego-key in steganography (Simmons, 1984, 1998; Anderson & Petitcolas, 1998; Cachin, 1998; Mittelholzer, 1999; Fridrich & Goljan, 2004; Fridrich, Goljan & Soukal, 2004; Lu, 2005), secret key in symmetric watermarking, and private key in asymmetric watermarking (Swanson, Kobayashi & Tewfik, 1998; Low & Maxemchuk, 1998; Hartung & Kutter, 1999; Mittelholzer, 1999; Mohanty, 1999; Wolfgang, Podilchuk & Delp, 1999; Eggers, Su & Girod, 2000; Collberg & Thomborson, 2002; Hachez & Quisquater, 2002; Arnold, Schmucker & Wolthusen 2003; Furon & Duhamel, 2003; Barni & Bartolini, 2004; Furon, 2005; Cayre Fontaine & Furon, 2005a, 2005b, 2005c; Lu, 2005; Cox, Doërr & Furon, 2006; Furht & Kirovski, 2006a, 2006b).

Lastly, non-cryptographic applications include seed for PRNG (Pseudo-Random Number Generator) and CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator) (Eastlake, Crocker & Schiller, 1994; Rukhin, Soto, Nechvatal, Smid, Barker, Leigh, Levenson, Vangel, Banks, Heckert, Dray & Vo, 2001; Le Quere, 2004; Keller, 2005; Barker & Kelsey, 2007; Campbell & Easter, 2007b) like the Blum-Blum-Shub (BBS) CSPRNG (Mollin, 2007a, p. 508, 2007b).

CHAPTER 10 APPLICATIONS OF BIG SECRET & MePKC (PART 2)

10.1 Identification Hardening of Embedded Data in Steganography

10.1.1 Related Works

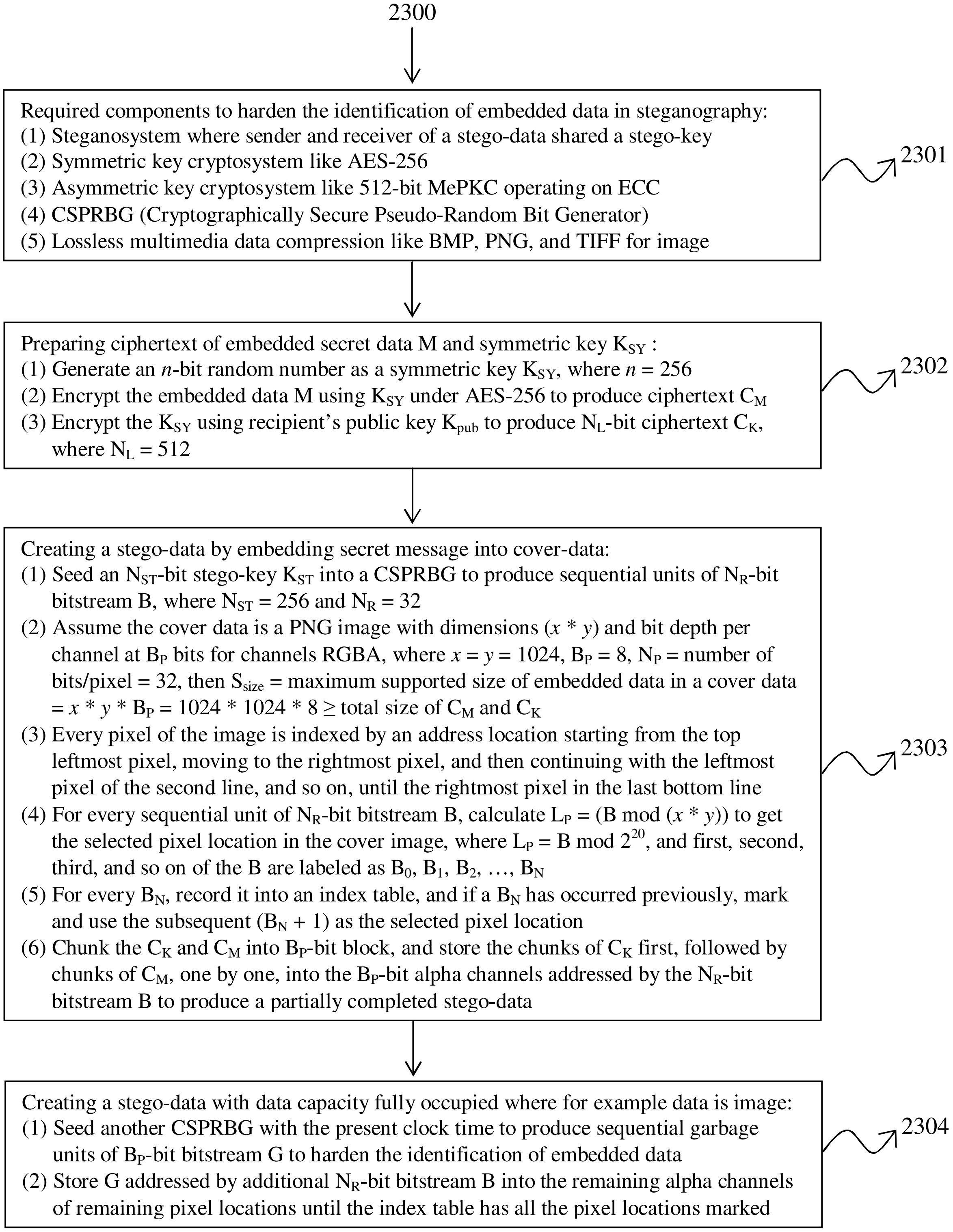
Steganography is a branch of information hiding. Secret message acts as embedded data into a cover data under the control of a stego-key to form a stego-data. Stego-data in its forms of storage and transmission through an insecure channel shall be like a normal data without triggering the suspicion of a person sensing the stego-data. To retrieve the secret message, the stego-data is processed using the stego-key to get back the embedded data.

In the current prior art, reliable detection of stego-image can be done successfully as shown by Fridrich and Goljan (2004). Yet the stego-key searching can also be done within promising time for a short stego-key.

This is reported by Fridrich, Goljan, and Soukal (2004) in “Searching for the Stego-Key” that as long as the embedded message is not occupying 100% of image capacity, then stego-key searching is independent of encryption key and takes about 12 hours to crack a 30-bit stego-key. Hence, there exists a need to have a big and yet memorizable stego-key, and to somehow fully occupy the data capacity for higher complexity to resist the cracking of steganographic system.

10.1.2 The Proposed MePKC Application 9.1(iv)

The fourth application 9.1(iv) of the present invention in applying the created big memorizable secret is to boost up the key size of stego-key to be more than 128 bits. Based on extrapolation of an article by Fridrich, Goljan, and Soukal (2004), for an 80-bit stego-key, it has a protection period of about 5 years or usable by year 2010 alike the 80-bit symmetric key. It is the contribution of the present embodiment to harden the identification of embedded data in steganography even after the stego-data has been detected as in Figures 10.1-10.2.



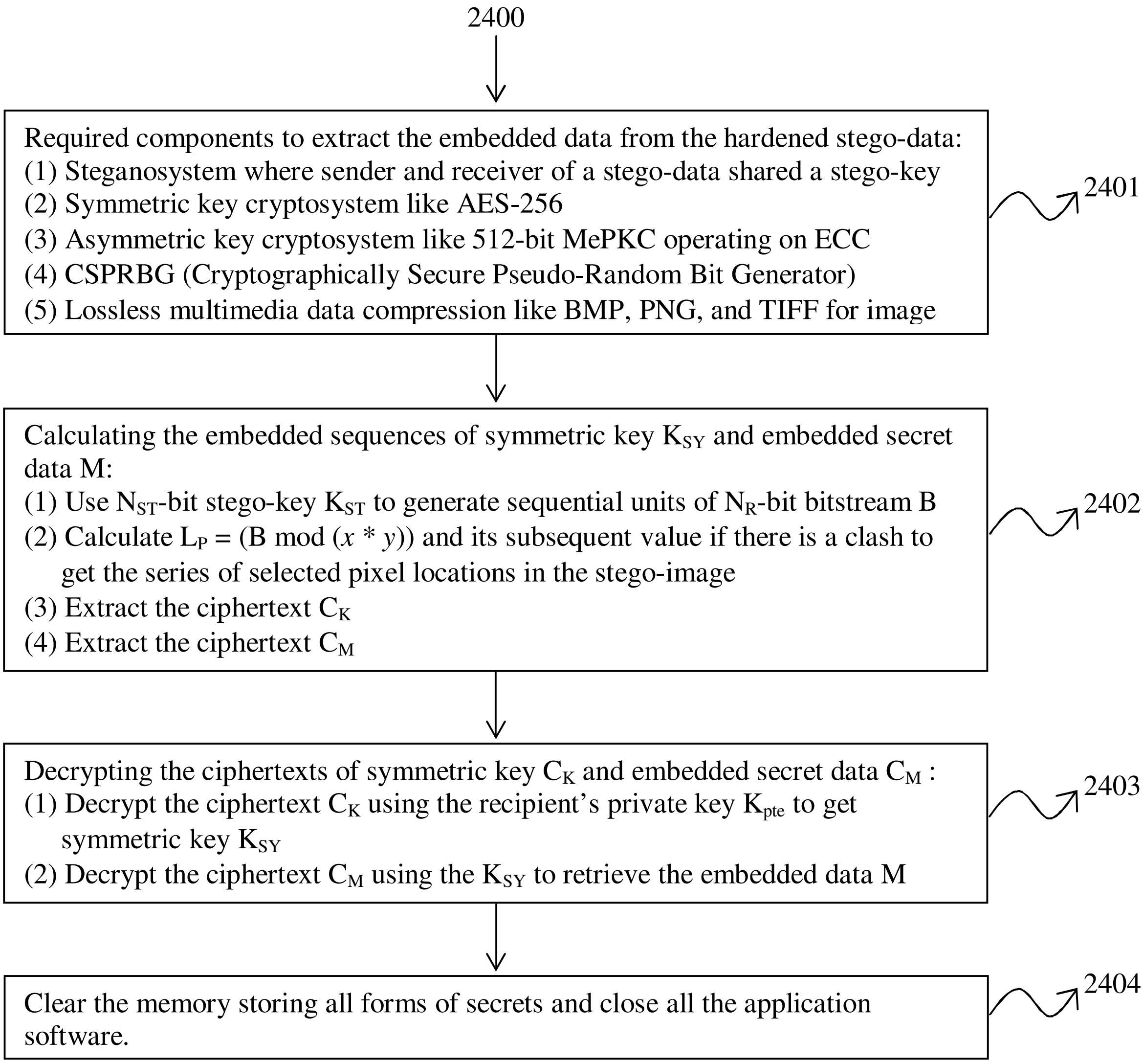
Required components to harden the identification of embedded data in steganography:
(1) Steganosystem where sender and receiver of a stego-data shared a stego-key
(2) Symmetric key cryptosystem like AES-256
(3) Asymmetric key cryptosystem like 512-bit MePKC operating on ECC
(4) CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator)
(5) Lossless multimedia data compression like BMP, PNG, and TIFF for image

Preparing ciphertext of embedded secret data M and symmetric key K_{SY} :
(1) Generate an n-bit random number as a symmetric key K_{SY}, where $n = 256$
(2) Encrypt the embedded data M using K_{SY} under AES-256 to produce ciphertext C_M
(3) Encrypt the K_{SY} using recipient's public key K_{pub} to produce N_L-bit ciphertext C_K, where $N_L = 512$

Creating a stego-data by embedding secret message into cover-data:
(1) Seed an N_{ST}-bit stego-key K_{ST} into a CSPRBG to produce sequential units of N_R-bit bitstream B, where $N_{ST} = 256$ and $N_R = 32$
(2) Assume the cover data is a PNG image with dimensions $(x * y)$ and bit depth per channel at B_P bits for channels RGBA, where $x = y = 1024$, $B_P = 8$, N_P = number of bits/pixel = 32, then S_{size} = maximum supported size of embedded data in a cover data $= x * y * B_P = 1024 * 1024 * 8 \geq$ total size of C_M and C_K
(3) Every pixel of the image is indexed by an address location starting from the top leftmost pixel, moving to the rightmost pixel, and then continuing with the leftmost pixel of the second line, and so on, until the rightmost pixel in the last bottom line
(4) For every sequential unit of N_R-bit bitstream B, calculate $L_P = (B \bmod (x * y))$ to get the selected pixel location in the cover image, where $L_P = B \bmod 2^{20}$, and first, second, third, and so on of the B are labeled as $B_0, B_1, B_2, \ldots, B_N$
(5) For every B_N, record it into an index table, and if a B_N has occurred previously, mark and use the subsequent $(B_N + 1)$ as the selected pixel location
(6) Chunk the C_K and C_M into B_P-bit block, and store the chunks of C_K first, followed by chunks of C_M, one by one, into the B_P-bit alpha channels addressed by the N_R-bit bitstream B to produce a partially completed stego-data

Creating a stego-data with data capacity fully occupied where for example data is image:
(1) Seed another CSPRBG with the present clock time to produce sequential garbage units of B_P-bit bitstream G to harden the identification of embedded data
(2) Store G addressed by additional N_R-bit bitstream B into the remaining alpha channels of remaining pixel locations until the index table has all the pixel locations marked

Figure 10.1 Data embedding process into a cover data for method and system to harden the identification of an embedded data in steganography although stego-data has been detected



Required components to extract the embedded data from the hardened stego-data:
(1) Steganosystem where sender and receiver of a stego-data shared a stego-key
(2) Symmetric key cryptosystem like AES-256
(3) Asymmetric key cryptosystem like 512-bit MePKC operating on ECC
(4) CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator)
(5) Lossless multimedia data compression like BMP, PNG, and TIFF for image

Calculating the embedded sequences of symmetric key K_{SY} and embedded secret data M:
(1) Use N_{ST}-bit stego-key K_{ST} to generate sequential units of N_R-bit bitstream B
(2) Calculate L_P = (B mod ($x * y$)) and its subsequent value if there is a clash to get the series of selected pixel locations in the stego-image
(3) Extract the ciphertext C_K
(4) Extract the ciphertext C_M

Decrypting the ciphertexts of symmetric key C_K and embedded secret data C_M :
(1) Decrypt the ciphertext C_K using the recipient's private key K_{pte} to get symmetric key K_{SY}
(2) Decrypt the ciphertext C_M using the K_{SY} to retrieve the embedded data M

Clear the memory storing all forms of secrets and close all the application software.

Figure 10.2 Data extracting process of embedded data from a stego-data for method and system to harden the identification of an embedded data in steganography although stego-data has been detected

Firstly, a stego-key is shared between the sender and receiver using some key exchange protocol like PAKE and MePKC key exchange scheme. Then, a symmetric key is created from a CSPRBG and use it to encrypt an embedded secret data to produce ciphertext of embedded data C_M. The symmetric key is later encrypted by recipient's public key to produce ciphertext of symmetric key C_K. To identify the address locations to hide the C_M and C_K, another CSPRBG is seeded with the stego-key and used to produce a list of addresses. Every unique address is recorded in an index table. If a generated address clashes with an address in the index table, then its subsequent address not in the index table is used.

After the C_M and C_K are hidden into the cover data, then use a third CSPRBG to generate random garbage bit streams G and use them to fully occupy the remaining data capacity. Consequently, from the full occupation of data capacity, the complexity to search for a stego-key will be higher when even encryption key searching is needed for cracking. To paralyze the stego-data detection, a sender can often broadcast dummy stego-data with noises as the embedded data.

10.2 Electronic Fund Transfer Using MePKC

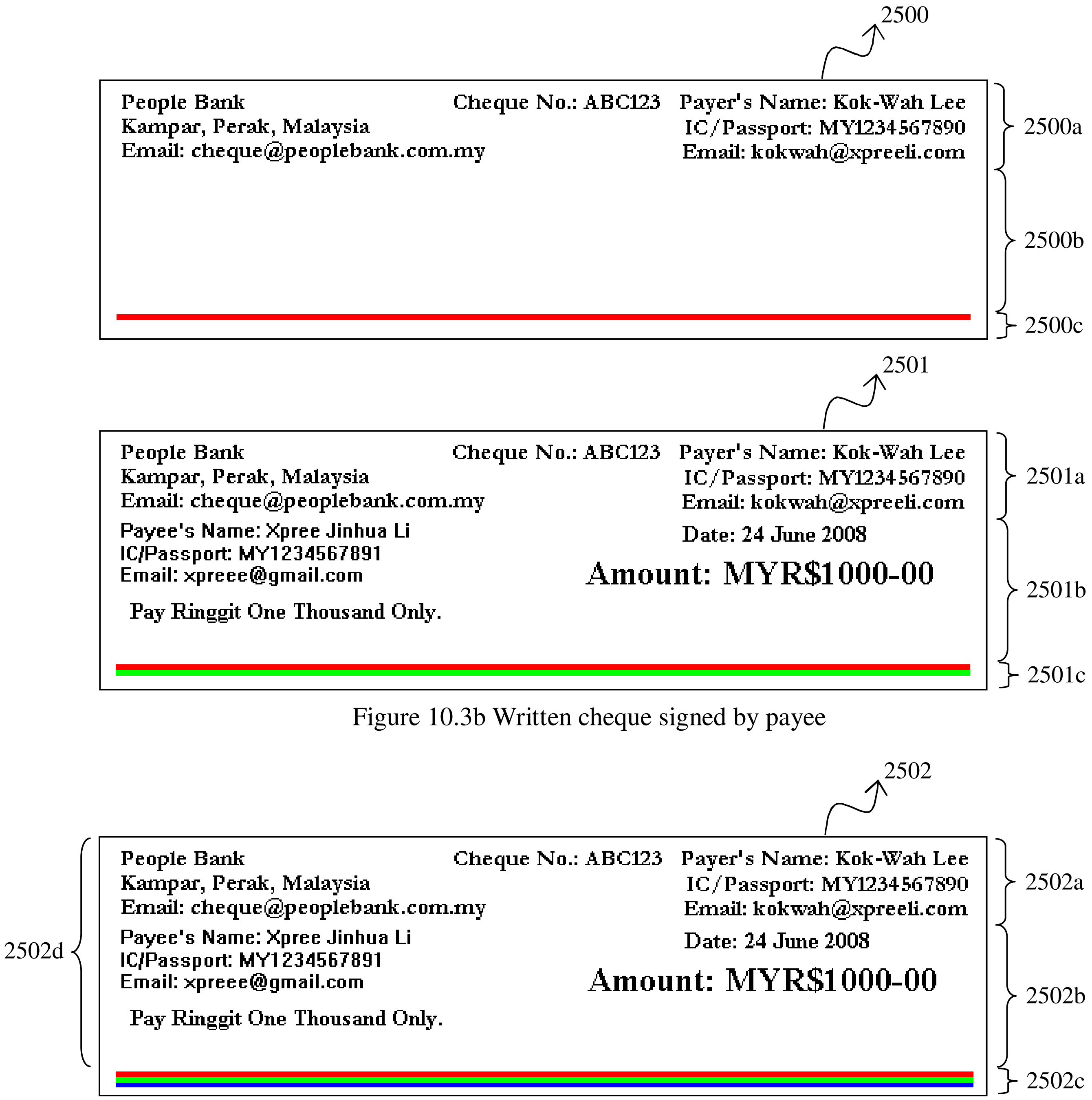
10.2.1 Related Works

Among the various applications of digital signature scheme, electronic cheque (aka digital cheque) is a special and important type of messages. Electronic cheque (Doggett, Jaffe & Anderson, 1997; Anderson, Jaffe, Hibbert, Virkki, Kravitz, Chang & Palmer, 2000, 2001) introduced another form of electronic fund transfer using conventional digital signature scheme. The popularity of these method and system are low due to the drawbacks of PKC (Public-Key Cryptography), i.e. low mobility of partially or fully encrypted private key, and management difficulty of certificate revocation list. Furthermore, the digital signature of Doggett's method carries only the information of electronic fund transfer from a payer to a payee via one or more banks.

In fact, a physical cheque has various processing states for accounting records like blank cheque, signed for payment, paid cheque, returned cheque by payee, withdrawn payment by payer, withdrawn payment by payer's bank, bounced cheque, advanced cheque, outdated cheque, fake cheque, etc.

Yet for electronic cheque, that transfer fund between accounts electronically and speedily throughout the world in the computer network, it shall have more optional security protection beyond the digital signature because money is a sensitive and critical object to be tracked for the convenient investigation of (organized) criminal activities (Lampe, No date; Glick, 1995; Livingston, 1996; "UNODC and

Organized Crime," No date; Layman & Potter, 1997; Maxim & Whitehead, 1998; Chen, 2004; Ruan & Wang, 2005; Siegel, 2005; Wang, 2007; "Identity-Related Crime," 2007; He, 2007; United Nations Office on Drugs and Crime (UNODC), 2004, 2008; Wikipedia Contributors, 2008s, 2008af) and civil cases.



People Bank
Kampar, Perak, Malaysia
Email: cheque@peoplebank.com.my

Cheque No.: ABC123

Payer's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

People Bank
Kampar, Perak, Malaysia
Email: cheque@peoplebank.com.my

Cheque No.: ABC123

Payer's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

Payee's Name: Xpree Jinhua Li
IC/Passport: MY1234567891
Email: xpreee@gmail.com

Date: 24 June 2008

Amount: MYR$1000-00

Pay Ringgit One Thousand Only.

Figure 10.3b Written cheque signed by payee

People Bank
Kampar, Perak, Malaysia
Email: cheque@peoplebank.com.my

Cheque No.: ABC123

Payer's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

Payee's Name: Xpree Jinhua Li
IC/Passport: MY1234567891
Email: xpreee@gmail.com

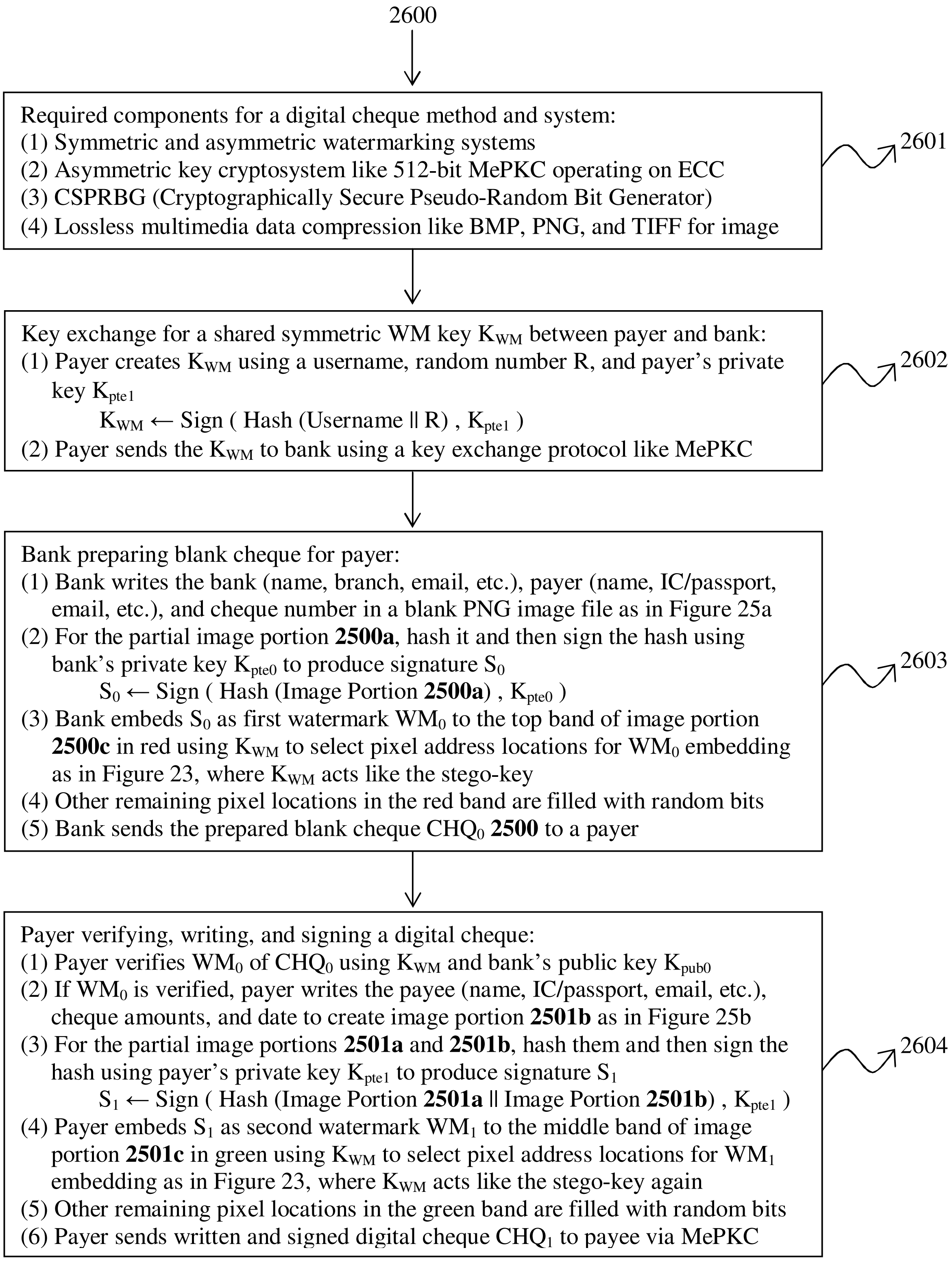
Date: 24 June 2008

Amount: MYR$1000-00

Pay Ringgit One Thousand Only.

Figure 10.3c Processed payee's cheque by bank

Figure 10.3 Samples of digital cheque in triple-watermark digital cheque scheme



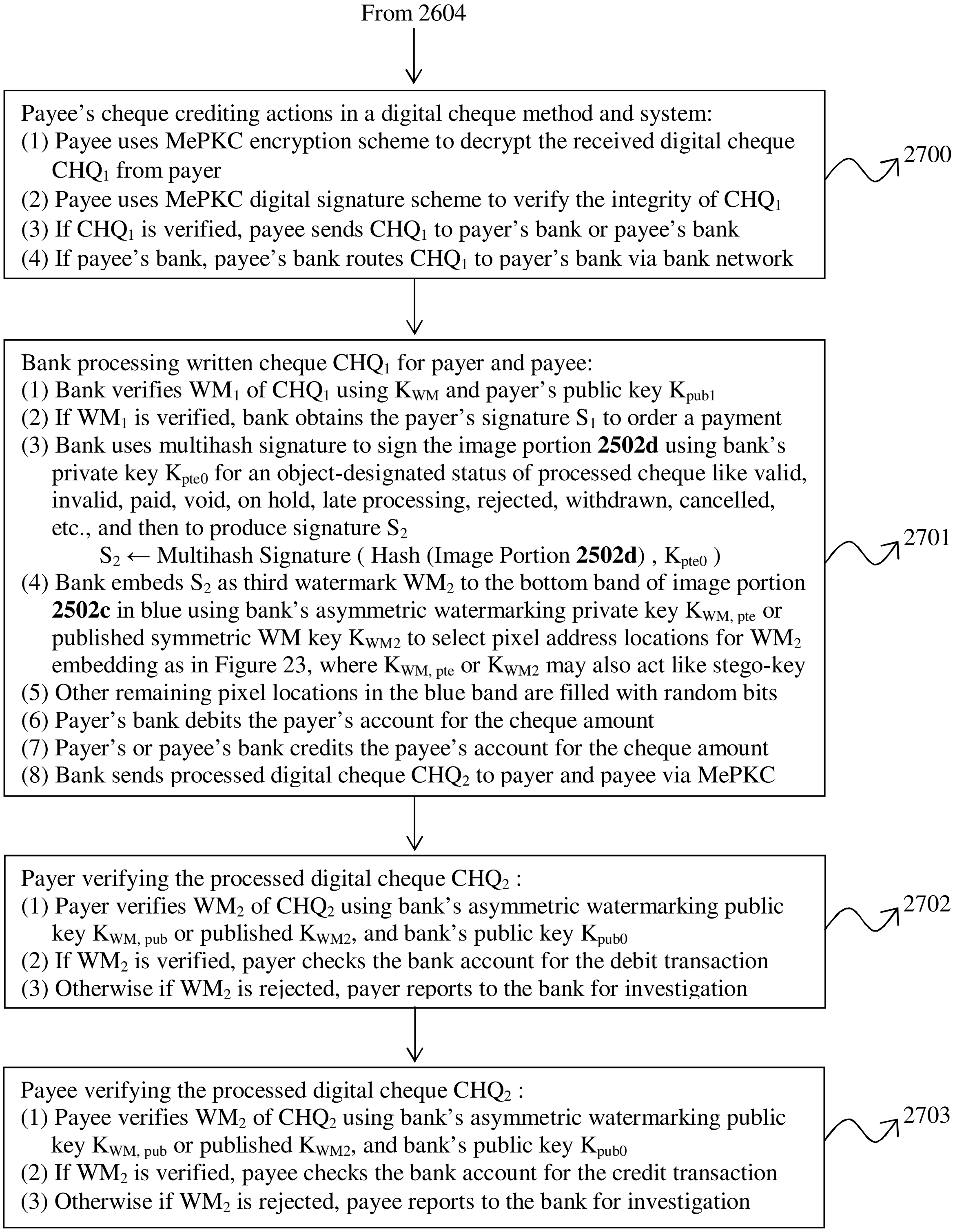
Required components for a digital cheque method and system:
(1) Symmetric and asymmetric watermarking systems
(2) Asymmetric key cryptosystem like 512-bit MePKC operating on ECC
(3) CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator)
(4) Lossless multimedia data compression like BMP, PNG, and TIFF for image

Key exchange for a shared symmetric WM key K_{WM} between payer and bank:
(1) Payer creates K_{WM} using a username, random number R, and payer's private key K_{pte1}
$K_{WM} \leftarrow \text{Sign}(\text{Hash}(\text{Username} \parallel R), K_{pte1})$
(2) Payer sends the K_{WM} to bank using a key exchange protocol like MePKC

Bank preparing blank cheque for payer:
(1) Bank writes the bank (name, branch, email, etc.), payer (name, IC/passport, email, etc.), and cheque number in a blank PNG image file as in Figure 25a
(2) For the partial image portion **2500a**, hash it and then sign the hash using bank's private key K_{pte0} to produce signature S_0
$S_0 \leftarrow \text{Sign}(\text{Hash}(\text{Image Portion } \mathbf{2500a}), K_{pte0})$
(3) Bank embeds S_0 as first watermark WM_0 to the top band of image portion **2500c** in red using K_{WM} to select pixel address locations for WM_0 embedding as in Figure 23, where K_{WM} acts like the stego-key
(4) Other remaining pixel locations in the red band are filled with random bits
(5) Bank sends the prepared blank cheque CHQ_0 **2500** to a payer

Payer verifying, writing, and signing a digital cheque:
(1) Payer verifies WM_0 of CHQ_0 using K_{WM} and bank's public key K_{pub0}
(2) If WM_0 is verified, payer writes the payee (name, IC/passport, email, etc.), cheque amounts, and date to create image portion **2501b** as in Figure 25b
(3) For the partial image portions **2501a** and **2501b**, hash them and then sign the hash using payer's private key K_{pte1} to produce signature S_1
$S_1 \leftarrow \text{Sign}(\text{Hash}(\text{Image Portion } \mathbf{2501a} \parallel \text{Image Portion } \mathbf{2501b}), K_{pte1})$
(4) Payer embeds S_1 as second watermark WM_1 to the middle band of image portion **2501c** in green using K_{WM} to select pixel address locations for WM_1 embedding as in Figure 23, where K_{WM} acts like the stego-key again
(5) Other remaining pixel locations in the green band are filled with random bits
(6) Payer sends written and signed digital cheque CHQ_1 to payee via MePKC

Figure 10.4 Creation of blank cheque by a bank and written cheque by a payer in the triple-watermark digital cheque method and system



Payee’s cheque crediting actions in a digital cheque method and system:
(1) Payee uses MePKC encryption scheme to decrypt the received digital cheque CHQ_1 from payer
(2) Payee uses MePKC digital signature scheme to verify the integrity of CHQ_1
(3) If CHQ_1 is verified, payee sends CHQ_1 to payer’s bank or payee’s bank
(4) If payee’s bank, payee’s bank routes CHQ_1 to payer’s bank via bank network

Bank processing written cheque CHQ_1 for payer and payee:
(1) Bank verifies WM_1 of CHQ_1 using K_{WM} and payer’s public key K_{pub1}
(2) If WM_1 is verified, bank obtains the payer’s signature S_1 to order a payment
(3) Bank uses multihash signature to sign the image portion **2502d** using bank’s private key K_{pte0} for an object-designated status of processed cheque like valid, invalid, paid, void, on hold, late processing, rejected, withdrawn, cancelled, etc., and then to produce signature S_2
$S_2 \leftarrow$ Multihash Signature (Hash (Image Portion **2502d**) , K_{pte0})
(4) Bank embeds S_2 as third watermark WM_2 to the bottom band of image portion **2502c** in blue using bank’s asymmetric watermarking private key $K_{WM,\,pte}$ or published symmetric WM key K_{WM2} to select pixel address locations for WM_2 embedding as in Figure 23, where $K_{WM,\,pte}$ or K_{WM2} may also act like stego-key
(5) Other remaining pixel locations in the blue band are filled with random bits
(6) Payer’s bank debits the payer’s account for the cheque amount
(7) Payer’s or payee’s bank credits the payee’s account for the cheque amount
(8) Bank sends processed digital cheque CHQ_2 to payer and payee via MePKC

Payer verifying the processed digital cheque CHQ_2 :
(1) Payer verifies WM_2 of CHQ_2 using bank’s asymmetric watermarking public key $K_{WM,\,pub}$ or published K_{WM2}, and bank’s public key K_{pub0}
(2) If WM_2 is verified, payer checks the bank account for the debit transaction
(3) Otherwise if WM_2 is rejected, payer reports to the bank for investigation

Payee verifying the processed digital cheque CHQ_2 :
(1) Payee verifies WM_2 of CHQ_2 using bank’s asymmetric watermarking public key $K_{WM,\,pub}$ or published K_{WM2}, and bank’s public key K_{pub0}
(2) If WM_2 is verified, payee checks the bank account for the credit transaction
(3) Otherwise if WM_2 is rejected, payee reports to the bank for investigation

Figure 10.5 Cheque crediting process by a payee in the triple-watermark digital cheque method and system

Hence, there exists a need to boost the PKC popularity, to add more embedded information, and to increase the security strength of electronic cheque, by applying fully memorizable private key, object-oriented signature scheme (aka multihash signature scheme), and optional fragile watermarking scheme, respectively.

10.2.2 The Proposed MePKC Application 9.1(v)

The fifth application 9.1(v) of the present invention in applying the created big memorizable secret is a method and system to transfer fund electronically over a remote network using MePKC, CSPRBG, lossless data compression, as well as information-hiding techniques like steganography and fragile watermarking, as in Figures 10.3-10.5 and called triple-watermark digital cheque method and system. Stronger security and prettier aesthetics are needed for digital cheque that is faster, more efficient, and more environment-friendly than paper cheque and electronic textual cheque using PKC merely.

There are three watermarks in the digital cheque. The first watermark marks the information of payer’s bank, payer, and cheque account signed by a payer’s bank. The second watermark marks the information of payee and cheque amount signed by a payer. The third watermark marks the cheque status after processed by the payer’s bank like valid, invalid, paid, void, on hold, late processing, rejected, withdrawn, cancelled, etc.

To save the image size, lossless image compression file format like PNG (Portable Network Graphics) and TIFF (Tagged Image File Format) shall be used besides BMP (Bitmap file format). Moreover, the digital cheque can also be in the data type of text. Also, this method and system can be modified and applied in other fields like software licensing.

10.3 Electronic Software Licensing Using MePKC

10.3.1 Related Works

Yet in another application of PKC (Public-Key Cryptography), software licensing is part of software copy protection besides code obfuscation against reverse engineering, watermarking against software piracy, and tamper-proofing against tampering (Collberg & Thomborson, 2002). In the current prior art, software licensing scheme uses fully or partially encrypted private key of PKC. Token containing the encrypted private key is subject to loss and damage; whereas server containing the encrypted private key is subject to virtual hacking and subsequently guessing attack, dictionary attack, and pre-computation attack.

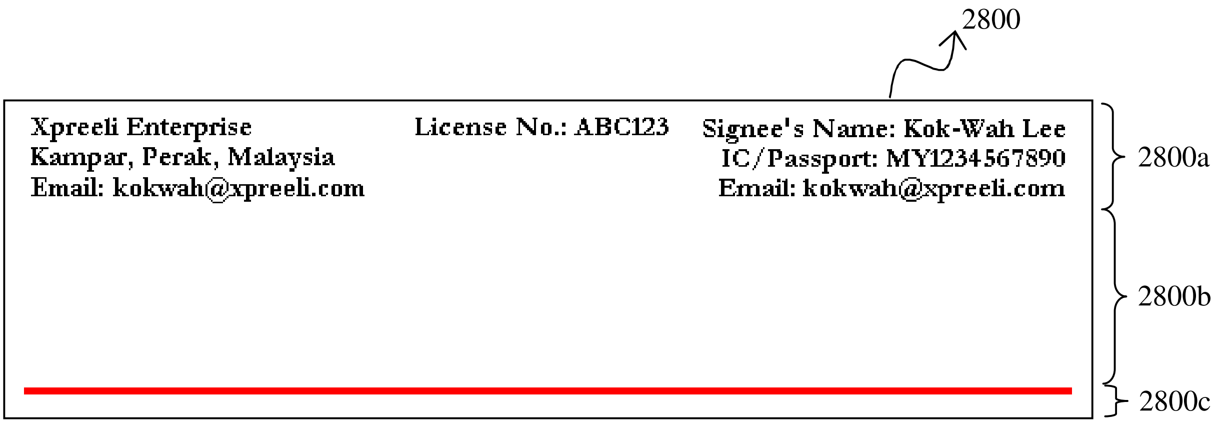
For computer software, its representative monetary value is its software product ID key rather than the duplicable electronic executable and storage device like floppy disk, CD-ROM, DVD, BD, HD DVD, etc., that stores the executable. Hence, there exists a need for current software licensing scheme to apply the fully memorizable private key for higher security and mobility, as well as to add more information using multihash signature scheme, and to have extra optional security protection to the software product ID key by using the fragile watermarking scheme.

10.3.2 The Proposed MePKC Appliacation 9.1(vi)

The sixth application 9.1(vi) of the present invention in applying the created big memorizable secret is a method and system to license software (Manoharan & Wu, 2007) electronically over a remote network using MePKC, CSPRBG, lossless data compression, as well as information-hiding techniques like steganography and fragile watermarking, as in Figures 10.6-10.8 and called triple-watermark digital software license method and system. Ethics, self-discipline, and education are mostly needed to fight against the software piracy (Limayem, Khalifa & Chin, 2004).

There are three watermarks in the digital software license. The first watermark marks the information of software licensing vendor, reseller (or sales agent), and reseller's account signed by a vendor. The second watermark marks the information of licensee and license selling price signed by a reseller. The third watermark marks the software license status after processed by the vendor like granted, upgraded, resold, void, withdrawn, evaluation, transferred, etc.

To save the image size, lossless image compression file format like PNG (Portable Network Graphics) and TIFF (Tagged Image File Format) shall be used besides BMP (Bitmap file format). Moreover, the digital software license can also be text data type. Also, this method and system can be modified and applied in other fields like digital cheque.

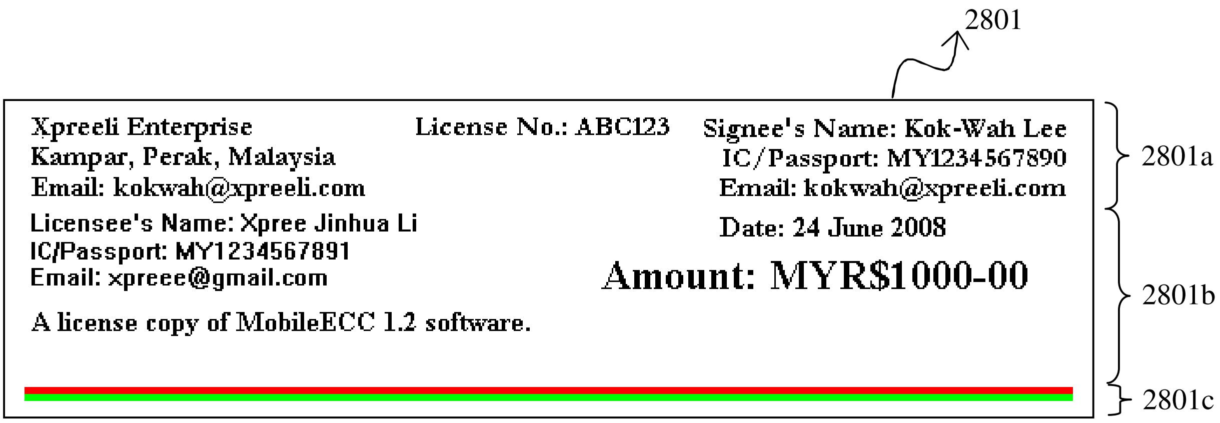


Xpreeli Enterprise
Kampar, Perak, Malaysia
Email: kokwah@xpreeli.com

License No.: ABC123

Signee's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

Figure 10.6a Blank software license issued by software vendor to reseller



Xpreeli Enterprise
Kampar, Perak, Malaysia
Email: kokwah@xpreeli.com

License No.: ABC123

Signee's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

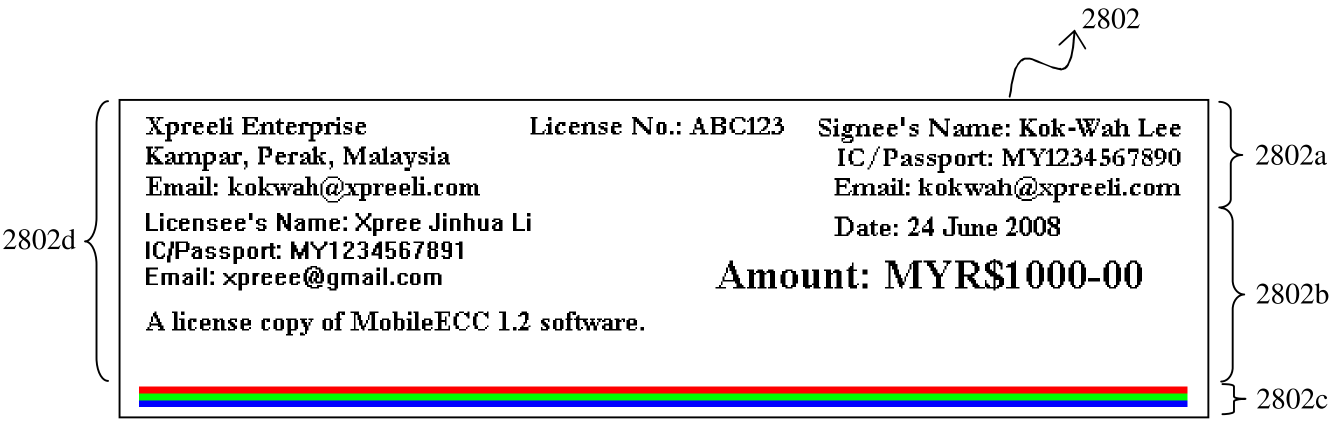
Licensee's Name: Xpree Jinhua Li
IC/Passport: MY1234567891
Email: xpreee@gmail.com

Date: 24 June 2008

Amount: MYR$1000-00

A license copy of MobileECC 1.2 software.

Figure 10.6b Written software license signed by reseller (or sales agent)



Xpreeli Enterprise
Kampar, Perak, Malaysia
Email: kokwah@xpreeli.com

License No.: ABC123

Signee's Name: Kok-Wah Lee
IC/Passport: MY1234567890
Email: kokwah@xpreeli.com

Licensee's Name: Xpree Jinhua Li
IC/Passport: MY1234567891
Email: xpreee@gmail.com

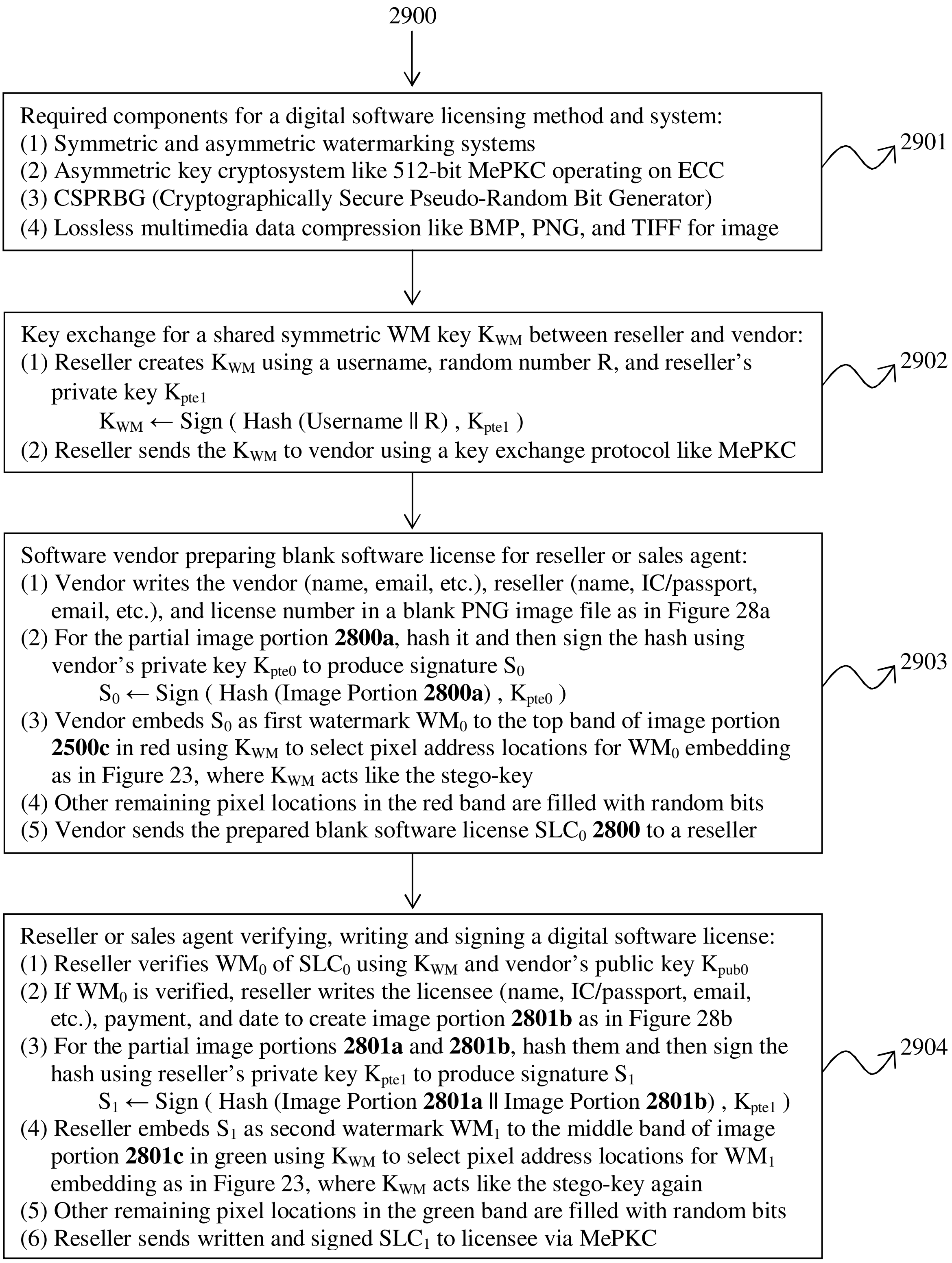
Date: 24 June 2008

Amount: MYR$1000-00

A license copy of MobileECC 1.2 software.

Figure 10.6c Processed software license by vendor

Figure 10.6 Samples of digital software license in triple-watermark digital software license scheme



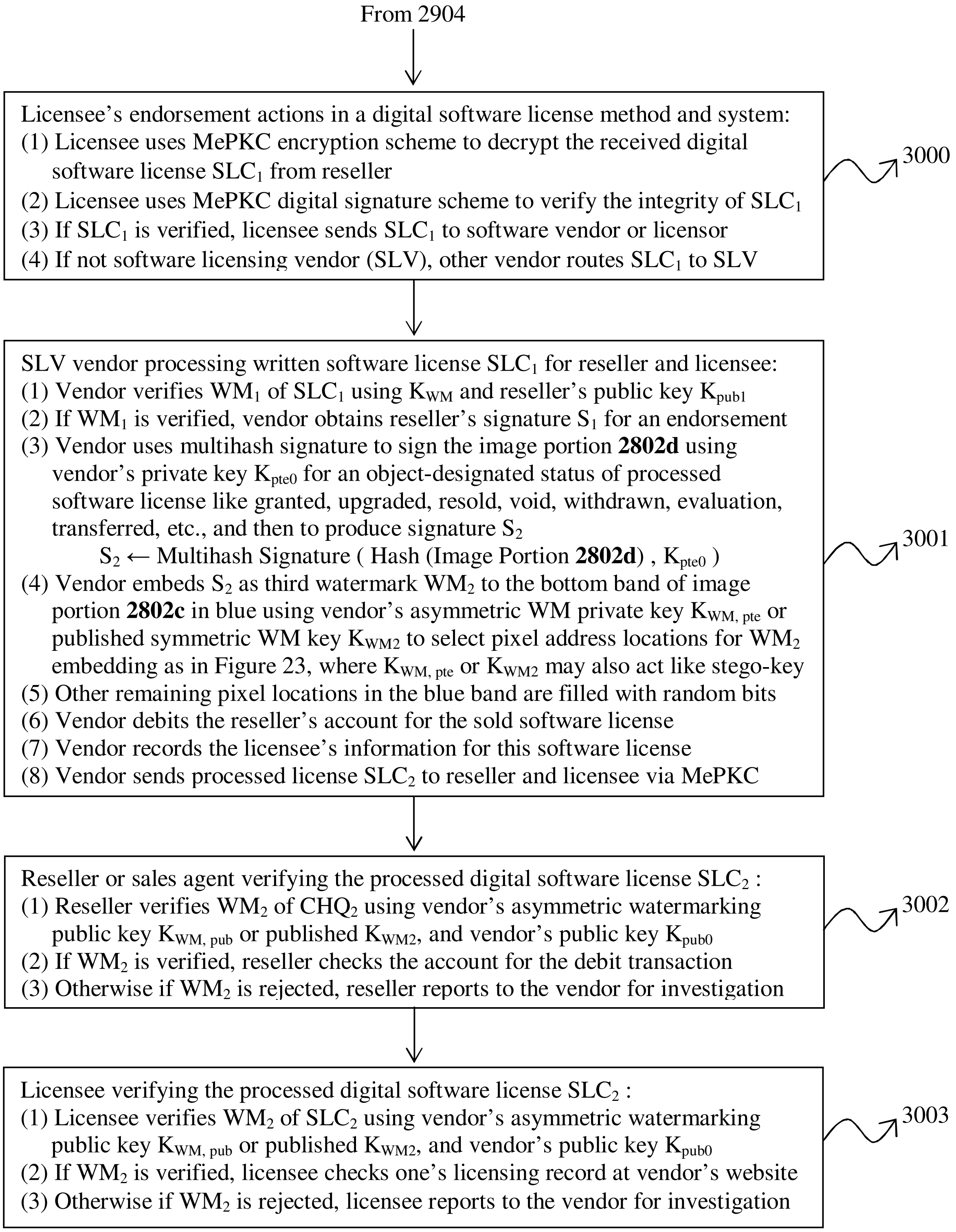
Required components for a digital software licensing method and system:
(1) Symmetric and asymmetric watermarking systems
(2) Asymmetric key cryptosystem like 512-bit MePKC operating on ECC
(3) CSPRBG (Cryptographically Secure Pseudo-Random Bit Generator)
(4) Lossless multimedia data compression like BMP, PNG, and TIFF for image

Key exchange for a shared symmetric WM key K_{WM} between reseller and vendor:
(1) Reseller creates K_{WM} using a username, random number R, and reseller's private key K_{pte1}
$K_{WM} \leftarrow$ Sign (Hash (Username || R) , K_{pte1})
(2) Reseller sends the K_{WM} to vendor using a key exchange protocol like MePKC

Software vendor preparing blank software license for reseller or sales agent:
(1) Vendor writes the vendor (name, email, etc.), reseller (name, IC/passport, email, etc.), and license number in a blank PNG image file as in Figure 28a
(2) For the partial image portion **2800a**, hash it and then sign the hash using vendor's private key K_{pte0} to produce signature S_0
$S_0 \leftarrow$ Sign (Hash (Image Portion **2800a**) , K_{pte0})
(3) Vendor embeds S_0 as first watermark WM_0 to the top band of image portion **2500c** in red using K_{WM} to select pixel address locations for WM_0 embedding as in Figure 23, where K_{WM} acts like the stego-key
(4) Other remaining pixel locations in the red band are filled with random bits
(5) Vendor sends the prepared blank software license SLC_0 **2800** to a reseller

Reseller or sales agent verifying, writing and signing a digital software license:
(1) Reseller verifies WM_0 of SLC_0 using K_{WM} and vendor's public key K_{pub0}
(2) If WM_0 is verified, reseller writes the licensee (name, IC/passport, email, etc.), payment, and date to create image portion **2801b** as in Figure 28b
(3) For the partial image portions **2801a** and **2801b**, hash them and then sign the hash using reseller's private key K_{pte1} to produce signature S_1
$S_1 \leftarrow$ Sign (Hash (Image Portion **2801a** || Image Portion **2801b**) , K_{pte1})
(4) Reseller embeds S_1 as second watermark WM_1 to the middle band of image portion **2801c** in green using K_{WM} to select pixel address locations for WM_1 embedding as in Figure 23, where K_{WM} acts like the stego-key again
(5) Other remaining pixel locations in the green band are filled with random bits
(6) Reseller sends written and signed SLC_1 to licensee via MePKC

Figure 10.7 Creation of blank software license by a vendor and written software license by a reseller in the triple-watermark digital software license method and system



Licensee's endorsement actions in a digital software license method and system:
(1) Licensee uses MePKC encryption scheme to decrypt the received digital software license SLC_1 from reseller
(2) Licensee uses MePKC digital signature scheme to verify the integrity of SLC_1
(3) If SLC_1 is verified, licensee sends SLC_1 to software vendor or licensor
(4) If not software licensing vendor (SLV), other vendor routes SLC_1 to SLV

SLV vendor processing written software license SLC_1 for reseller and licensee:
(1) Vendor verifies WM_1 of SLC_1 using K_{WM} and reseller's public key K_{pub1}
(2) If WM_1 is verified, vendor obtains reseller's signature S_1 for an endorsement
(3) Vendor uses multihash signature to sign the image portion **2802d** using vendor's private key K_{pte0} for an object-designated status of processed software license like granted, upgraded, resold, void, withdrawn, evaluation, transferred, etc., and then to produce signature S_2
$S_2 \leftarrow$ Multihash Signature (Hash (Image Portion **2802d**) , K_{pte0})
(4) Vendor embeds S_2 as third watermark WM_2 to the bottom band of image portion **2802c** in blue using vendor's asymmetric WM private key $K_{WM,\,pte}$ or published symmetric WM key K_{WM2} to select pixel address locations for WM_2 embedding as in Figure 23, where $K_{WM,\,pte}$ or K_{WM2} may also act like stego-key
(5) Other remaining pixel locations in the blue band are filled with random bits
(6) Vendor debits the reseller's account for the sold software license
(7) Vendor records the licensee's information for this software license
(8) Vendor sends processed license SLC_2 to reseller and licensee via MePKC

Reseller or sales agent verifying the processed digital software license SLC_2 :
(1) Reseller verifies WM_2 of CHQ_2 using vendor's asymmetric watermarking public key $K_{WM,\,pub}$ or published K_{WM2}, and vendor's public key K_{pub0}
(2) If WM_2 is verified, reseller checks the account for the debit transaction
(3) Otherwise if WM_2 is rejected, reseller reports to the vendor for investigation

Licensee verifying the processed digital software license SLC_2 :
(1) Licensee verifies WM_2 of SLC_2 using vendor's asymmetric watermarking public key $K_{WM,\,pub}$ or published K_{WM2}, and vendor's public key K_{pub0}
(2) If WM_2 is verified, licensee checks one's licensing record at vendor's website
(3) Otherwise if WM_2 is rejected, licensee reports to the vendor for investigation

Figure 10.8 Endorsement process of a software license by a licensee in the triple-watermark digital software license method and system

CHAPTER 11 APPLICATIONS OF BIG SECRET & MePKC (PART 3)

11.1 MePKC Human-Computer and Human-Human Authentication Schemes

11.1.1 Related Works: Computer Password Authentication Protocol

In this networked info-computer age, computer-computer mutual authentication uses asymmetric key cryptography, but human-computer and human-human mutual authentications till now still stick to symmetric key cryptography. In fact, the most frequently used application of secret is authentication access of a human to a computer for online account access. The online computer authentication methods (Beutelspacher, 1994) using password the secret include (i) simple transmission of key, (ii) transmission of encrypted key, (iii) transmission of key through encrypted channels, (iv) hash-based challenge-response method, (v) zero-knowledge password proof, and (vi) PAKE (Password-Authenticated Key Exchange). All of these six methods are based on a shared secret between a user and the server.

The first method using simple transmission of key in the clear channel is an insecure approach. The second method using transmission of encrypted key is in fact firstly proposed by H. Feistel (1974a, 1974b, 1974c) in his three patents, US Patents: US3798359 “Block Cipher Cryptographic System”, US3798360 “Step Code Ciphering System”, and US3798605 “Centralized Verification System”, filed on the same day on 30 June 1971. For the third method using transmission of key through encrypted channels, the encrypted channels are based on the protocols like SSL (Secure Sockets Layer) or TLS (Transport Layer Security). Later, hash function is created and subsequently the fourth method called hash-based challenge-response method using hashed password, where a server stores the hash value of a password. The second, third, and fourth methods remain as the current most popular online computer authentication methods till today.

For the fifth method called zero-knowledge password proof, it is more complicated where a secret owner can prove to a verifier its ownership of a secret without revealing the secret. The fifth method is somehow modified to become the sixth method called PAKE. Examples of PAKE include EKE (Encrypted Key Exchange), PAK (Password-Authenticated Key exchange), PPK (Password-Protected Key exchange), SPEKE (Simple Password Exponential Key Exchange) (Jablon, 2006), SRP-6 (Simple Remote Password Protocol version 6) (Wu, 2003), etc.

For a good computer password authentication protocol, there are three main issues to be fulfilled: Resistance to dictionary attack, (perfect) forward secrecy, and non-plaintext equivalence. Forward secrecy means resistance to compromise secret information if another part of the protocol is compromised. Perfect forward secrecy means the compromise of long-term key used to derive an agreed ephemeral key does not compromise the agreed keys from earlier runs. Non-plaintext equivalence means a data that cannot be used to gain the same access level of a key/password.

Computer password authentication protocols that can resist dictionary attack are EKE (Encrypted Key Exchange) family of protocols and a few public-key assisted protocols. Protocols that can fulfill the conditions of resistance to dictionary attack and prefect forward secrecy are the strongest members of EKE family of protocols like DH-EKE (Diffie-Hellman Encrypted Key Exchange) and SPEKE (Simple Password Exponential Key Exchange). SPEKE was firstly proposed by D. P. Jablon (2004) in US Patent: US7010692 “Cryptographic Methods for Remote Authentication”. For protocol that can fulfil all the three issues of resistance to dictionary attack, prefect forward secrecy, and non-plaintext equivalence, there is currently only one called SRP-6 (Simple Remote Password Protocol version 6). SRP was firstly proposed by T. J. Wu (1998) in US Patent: US6539479 “System and Method for Securely Logging onto a Remotely Located Computer”.

Nevertheless, the PAKE of SRP-6 still has a long-term shared secret and is not yet a fully asymmetric key cryptosystem. Hence, if the long-term shared secret is re-used, SRP-6 is subject to malicious server attack, where the faulty server having the username, salt, and verifier can pretend to be the another actual server using the same secret. Moreover, it is lacking of mutual authentication. As compared with the MePKC authentication methods and systems in the preferred embodiment of this thesis, SRP-6 also has more rounds of message exchange, more IP packets and longer processing time.

For authentication protocol operating on the platform of asymmetric key cryptosystem, split private key cryptosystem has a few protocols for these purposes. However, the private key of split private key cryptosystem is only partially memorizable and another portion of private key is stored in the authentication server. The weakness of split private key cryptosystem is a malicious authentication server can launch guessing attack and dictionary attack over the first portion of memorizable split private key. Hence, there exists a need to have a password authentication protocol for human-computer and human-human interfaces that operates on the asymmetric key cryptosystem using a fully memorizable private key for each user.

11.1.2 First Model without Perfect Forward Secrecy

Yet in the seventh application 9.1(vii) of the present invention in applying the created big memorizable secret, two MePKC human-computer and human-human authentication schemes between a human user and a local computer or remote server (or human user) over an insecure computer communication network are presented. Challenge-response authentication protocol is adopted for these authentication schemes without any shared secret and transmission of secret key over the insecure channel. The challenge has a nonce to resist replay attack. Nonce stands for "number used once" and may be a one-time random number, counter, or timestamp. Yet one of many advantages is no storage of encrypted password, hashed password, verifier, or shared secret in the local or remote computing system. Subsequently, this MePKC authentication scheme can also resist phishing attack and spoofing attack that try to steal user password.

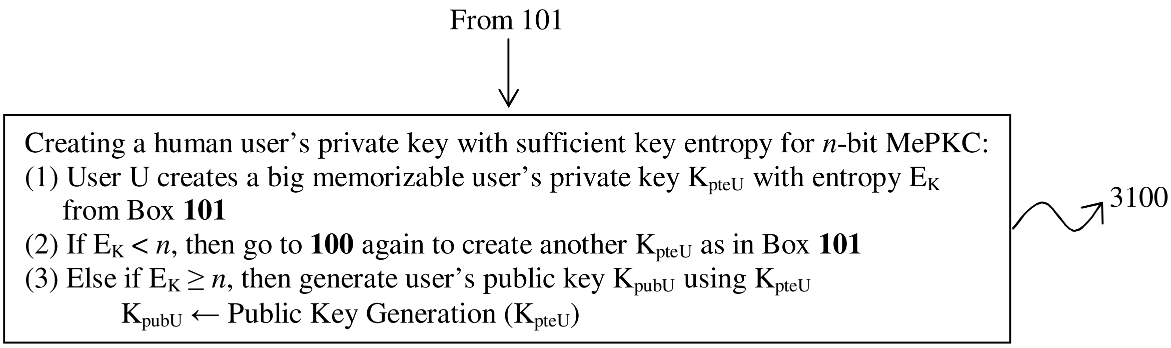
Since there is no storage of password, system and network administrators will no longer know the secret of any user's key. This allows a user to use the same asymmetric key pair for different offline/online accounts. By sharing the same asymmetric key pair among different accounts, the memorizability of a user is improved, and hence there is no more need to jot down various keys in the notebook. Since there is no encrypted password, hashed password, or verifier, the pre-computation attack can be avoided. Other attacks such as guessing attack, dictionary attack, and brute force attack will still be possible. However, guessing attack and dictionary attack can be avoided if the 2D key, multilingual key, multi-tier geo-image key, or multi-factor multimedia key is used properly as for the key style of ASCII art and Unicode art. If the same asymmetric key pair is used together with multihash key to create different slave keys for different online accounts, this allows pseudo-one-set password entry to multiple websites without having password domino cracking effect as in the symmetric key cryptosystems.

However, the disadvantage of MePKC authentication schemes is the slow processing speed of PKC. Hence, the size of challenge message has to be limited to only a few units of encryption block of PKC, like block size of 256 to 512 bits for 256- to 512-bit MePKC, respectively. A wonderful authentication scheme over a computer communication network shall have the features of non-plaintext equivalence, prefect forward secrecy, and resistance to dictionary attack.

For the first basic model of the MePKC authentication scheme as in Figures 11.1-11.2, it has the features of non-plaintext equivalence internally and resistance to dictionary attack externally by using secret creation method of 2D key, multilingual key, multi-tier geo-image key, or multi-factor multimedia key. The first basic model is still lacking of the feature of prefect forward secrecy, because the compromise of long-term private key used to derive an agreed ephemeral key does compromise the agreed keys from earlier runs.

11.1.3 Second Model with Perfect Forward Secrecy

To include the feature of prefect forward secrecy, the second model of MePKC authentication scheme as in Figures 11.3-11.5 is innovated. Now, a human user may use multihash key and has a long-term asymmetric key pair [K_{pteUL} , K_{pubUL}] and a one-time asymmetric key pair [K_{pteU} , K_{pubU}] acting as rolling key for each login or authentication access. Now, the compromise of long-term private key used to derive an agreed ephemeral key does not compromise the agreed keys from earlier runs. An added feature for this second model is the optional inclusion of a key exchange scheme to establish a shared key between the human user and remote server.



Creating a human user's private key with sufficient key entropy for n-bit MePKC:

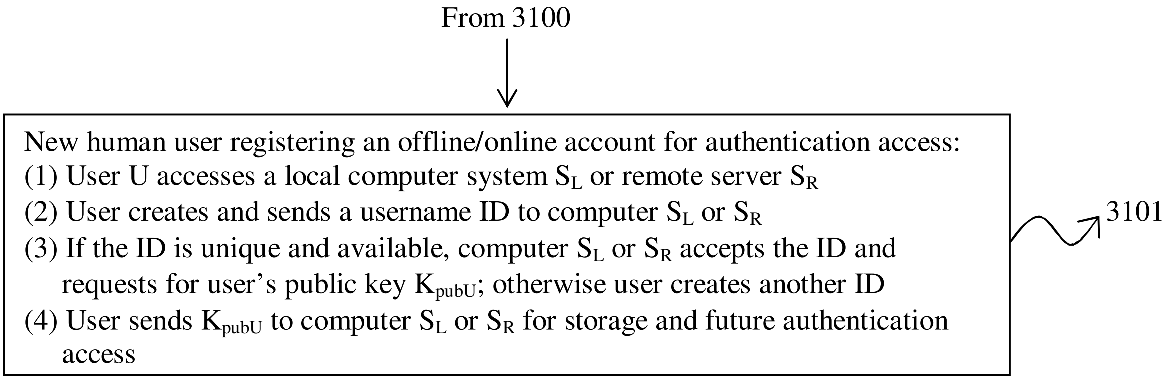
(1) User U creates a big memorizable user's private key K_{pteU} with entropy E_K from Box **101**

(2) If $E_K < n$, then go to **100** again to create another K_{pteU} as in Box **101**

(3) Else if $E_K \geq n$, then generate user's public key K_{pubU} using K_{pteU}

$K_{pubU} \leftarrow$ Public Key Generation (K_{pteU})

Figure 11.1a Creating a sufficiently big and yet memorizable user's private key



New human user registering an offline/online account for authentication access:

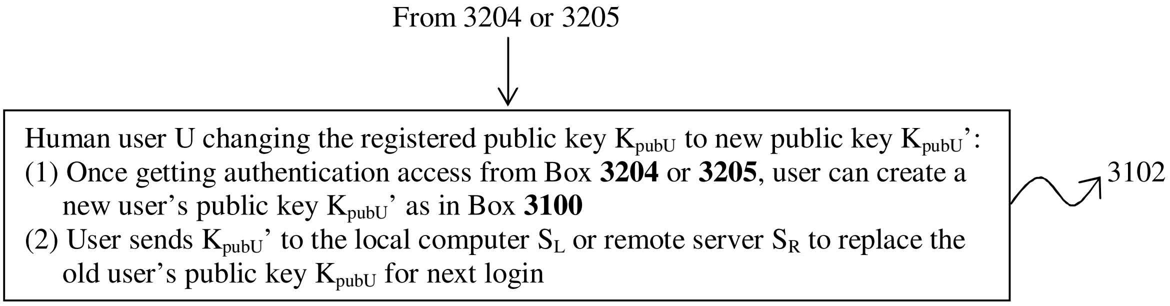
(1) User U accesses a local computer system S_L or remote server S_R

(2) User creates and sends a username ID to computer S_L or S_R

(3) If the ID is unique and available, computer S_L or S_R accepts the ID and requests for user's public key K_{pubU}; otherwise user creates another ID

(4) User sends K_{pubU} to computer S_L or S_R for storage and future authentication access

Figure 11.1b Account registration of a new user



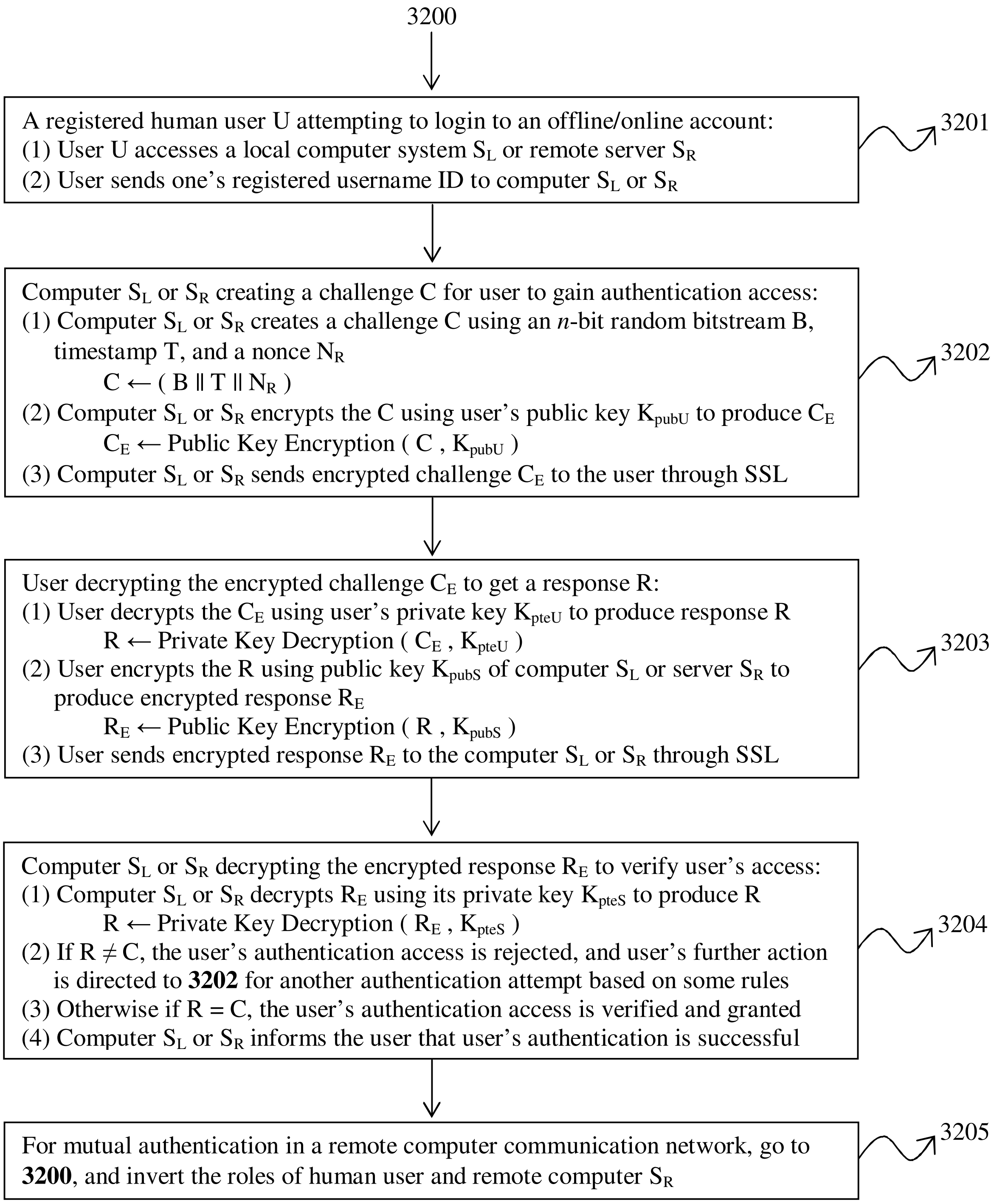
Human user U changing the registered public key K_{pubU} to new public key K_{pubU}':

(1) Once getting authentication access from Box **3204** or **3205**, user can create a new user's public key K_{pubU}' as in Box **3100**

(2) User sends K_{pubU}' to the local computer S_L or remote server S_R to replace the old user's public key K_{pubU} for next login

Figure 11.1c Replacing a user's public key by a user

Figure 11.1 Various not-so-frequent operations of the basic model of MePKC authentication schemes with feature of non-plaintext equivalence



A registered human user U attempting to login to an offline/online account:
(1) User U accesses a local computer system S_L or remote server S_R
(2) User sends one's registered username ID to computer S_L or S_R

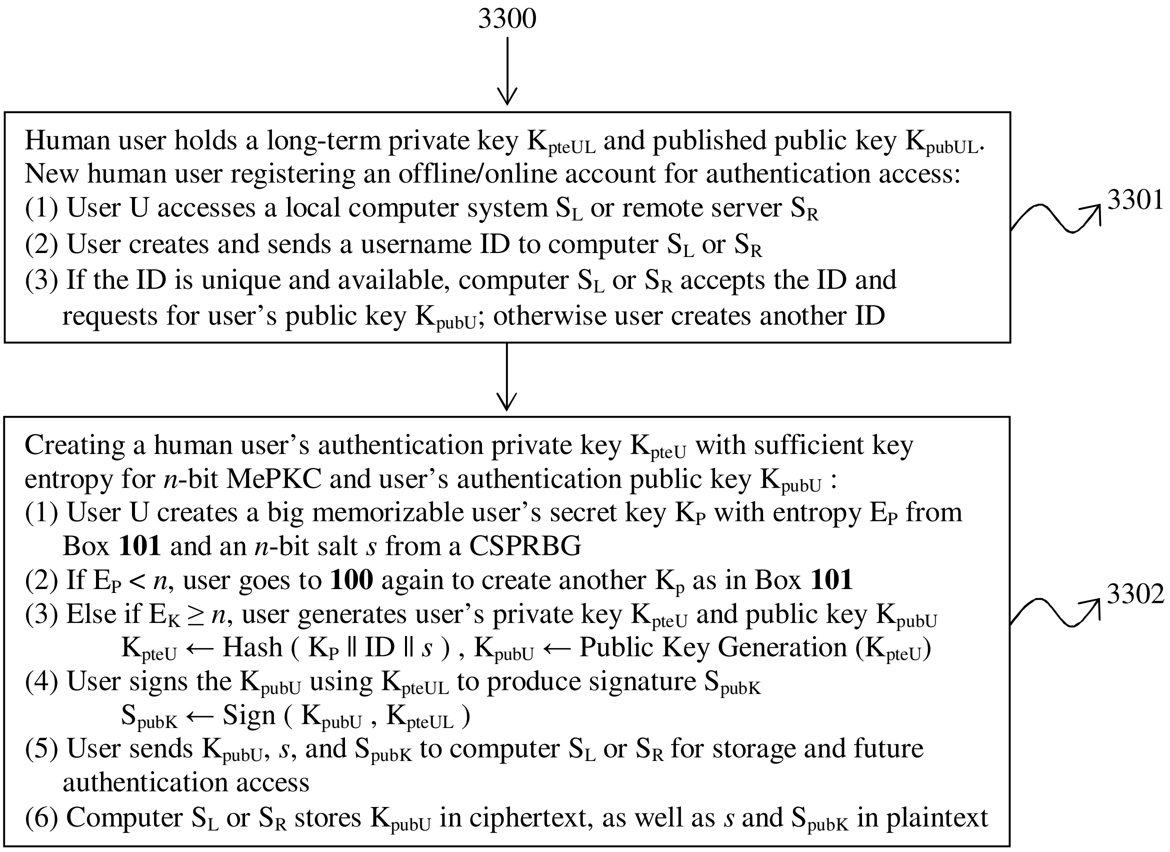
Computer S_L or S_R creating a challenge C for user to gain authentication access:
(1) Computer S_L or S_R creates a challenge C using an *n*-bit random bitstream B, timestamp T, and a nonce N_R
$C \leftarrow (B \parallel T \parallel N_R)$
(2) Computer S_L or S_R encrypts the C using user's public key K_{pubU} to produce C_E
$C_E \leftarrow$ Public Key Encryption (C , K_{pubU})
(3) Computer S_L or S_R sends encrypted challenge C_E to the user through SSL

User decrypting the encrypted challenge C_E to get a response R:
(1) User decrypts the C_E using user's private key K_{pteU} to produce response R
$R \leftarrow$ Private Key Decryption (C_E , K_{pteU})
(2) User encrypts the R using public key K_{pubS} of computer S_L or server S_R to produce encrypted response R_E
$R_E \leftarrow$ Public Key Encryption (R , K_{pubS})
(3) User sends encrypted response R_E to the computer S_L or S_R through SSL

Computer S_L or S_R decrypting the encrypted response R_E to verify user's access:
(1) Computer S_L or S_R decrypts R_E using its private key K_{pteS} to produce R
$R \leftarrow$ Private Key Decryption (R_E , K_{pteS})
(2) If $R \neq C$, the user's authentication access is rejected, and user's further action is directed to **3202** for another authentication attempt based on some rules
(3) Otherwise if $R = C$, the user's authentication access is verified and granted
(4) Computer S_L or S_R informs the user that user's authentication is successful

For mutual authentication in a remote computer communication network, go to **3200**, and invert the roles of human user and remote computer S_R

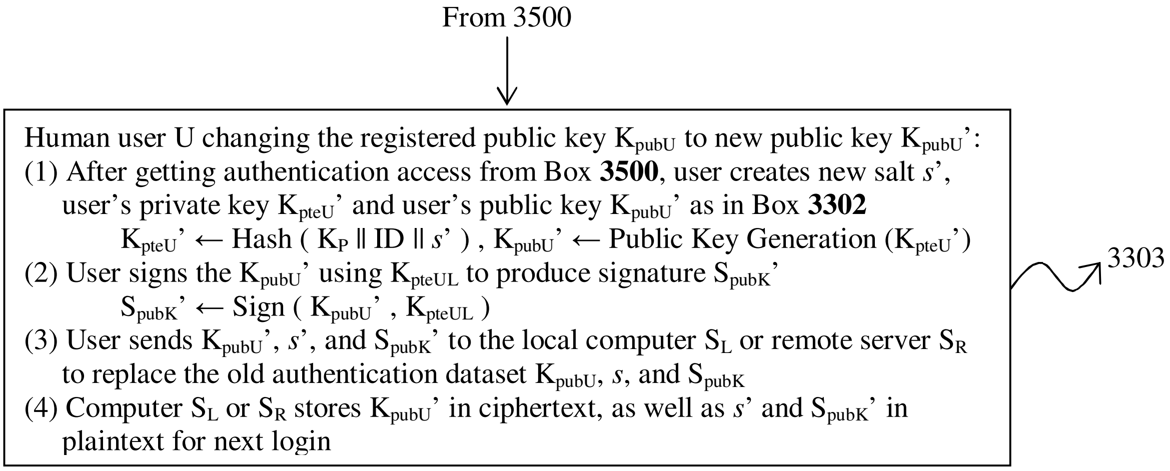
Figure 11.2 First basic model of MePKC authentication scheme between a human user and a computer with features of non-plaintext equivalence and optional mutual authentication



Human user holds a long-term private key K_{pteUL} and published public key K_{pubUL}.
New human user registering an offline/online account for authentication access:
(1) User U accesses a local computer system S_L or remote server S_R
(2) User creates and sends a username ID to computer S_L or S_R
(3) If the ID is unique and available, computer S_L or S_R accepts the ID and requests for user's public key K_{pubU}; otherwise user creates another ID

Creating a human user's authentication private key K_{pteU} with sufficient key entropy for *n*-bit MePKC and user's authentication public key K_{pubU} :
(1) User U creates a big memorizable user's secret key K_P with entropy E_P from Box **101** and an *n*-bit salt *s* from a CSPRBG
(2) If $E_P < n$, user goes to **100** again to create another K_p as in Box **101**
(3) Else if $E_K \geq n$, user generates user's private key K_{pteU} and public key K_{pubU}
$K_{pteU} \leftarrow \text{Hash}\ (\ K_P \,||\, \text{ID} \,||\, s\ )$, $K_{pubU} \leftarrow \text{Public Key Generation}\ (K_{pteU})$
(4) User signs the K_{pubU} using K_{pteUL} to produce signature S_{pubK}
$S_{pubK} \leftarrow \text{Sign}\ (\ K_{pubU}\ ,\ K_{pteUL}\ )$
(5) User sends K_{pubU}, *s*, and S_{pubK} to computer S_L or S_R for storage and future authentication access
(6) Computer S_L or S_R stores K_{pubU} in ciphertext, as well as *s* and S_{pubK} in plaintext

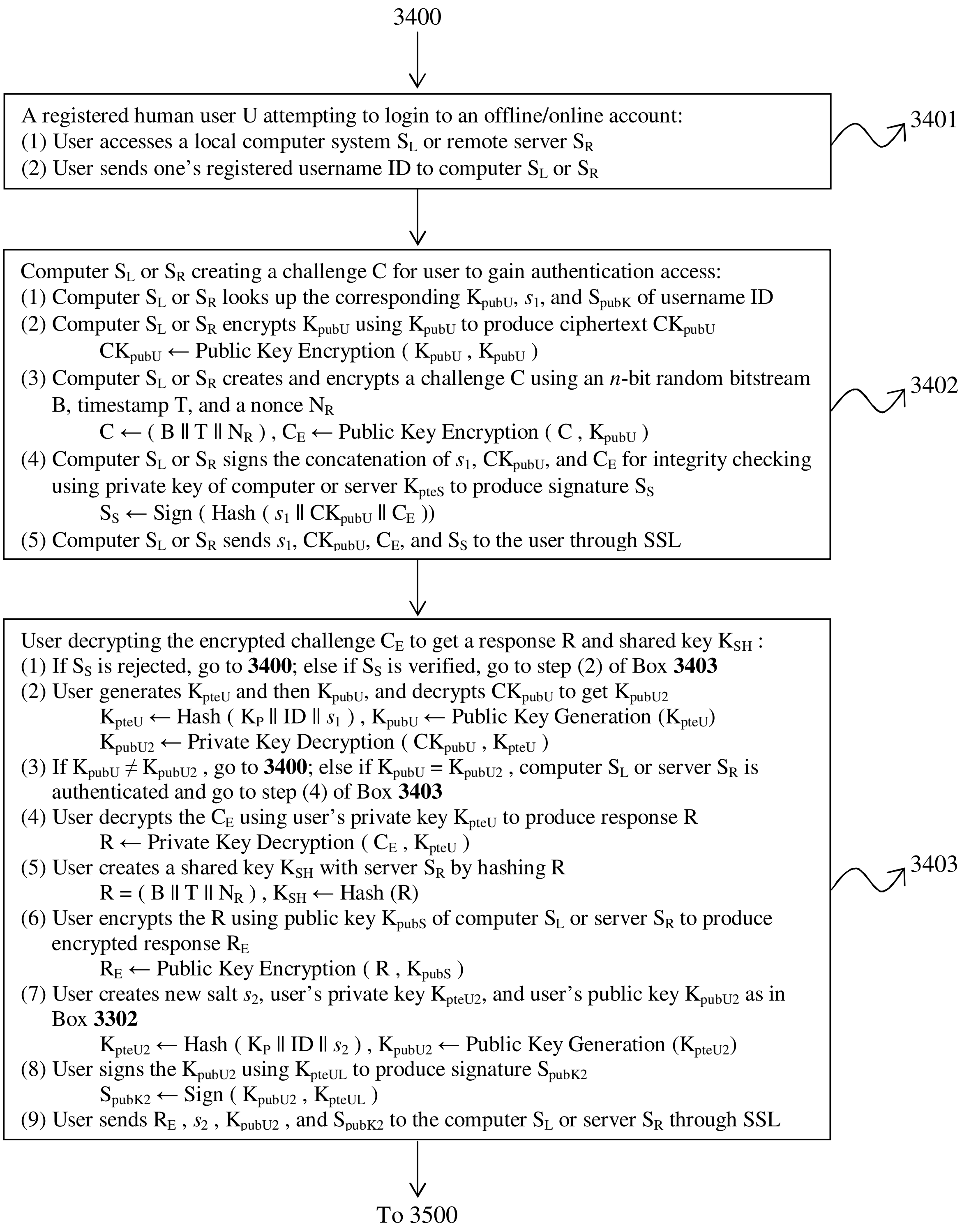
Figure 11.3a Account registration of a new user by creating a sufficiently big and yet memorizable user's private key



Human user U changing the registered public key K_{pubU} to new public key K_{pubU}':
(1) After getting authentication access from Box **3500**, user creates new salt *s*', user's private key K_{pteU}' and user's public key K_{pubU}' as in Box **3302**
$K_{pteU}' \leftarrow \text{Hash}\ (\ K_P \,||\, \text{ID} \,||\, s'\ )$, $K_{pubU}' \leftarrow \text{Public Key Generation}\ (K_{pteU}')$
(2) User signs the K_{pubU}' using K_{pteUL} to produce signature S_{pubK}'
$S_{pubK}' \leftarrow \text{Sign}\ (\ K_{pubU}'\ ,\ K_{pteUL}\ )$
(3) User sends K_{pubU}', *s*', and S_{pubK}' to the local computer S_L or remote server S_R to replace the old authentication dataset K_{pubU}, *s*, and S_{pubK}
(4) Computer S_L or S_R stores K_{pubU}' in ciphertext, as well as *s*' and S_{pubK}' in plaintext for next login

Figure 11.3b Replacing a user's authentication dataset like user's public key and salt by a user

Figure 11.3 Various not-so-frequent operations of the second model of MePKC authentication schemes with features of non-plaintext equivalence and perfect forward secrecy

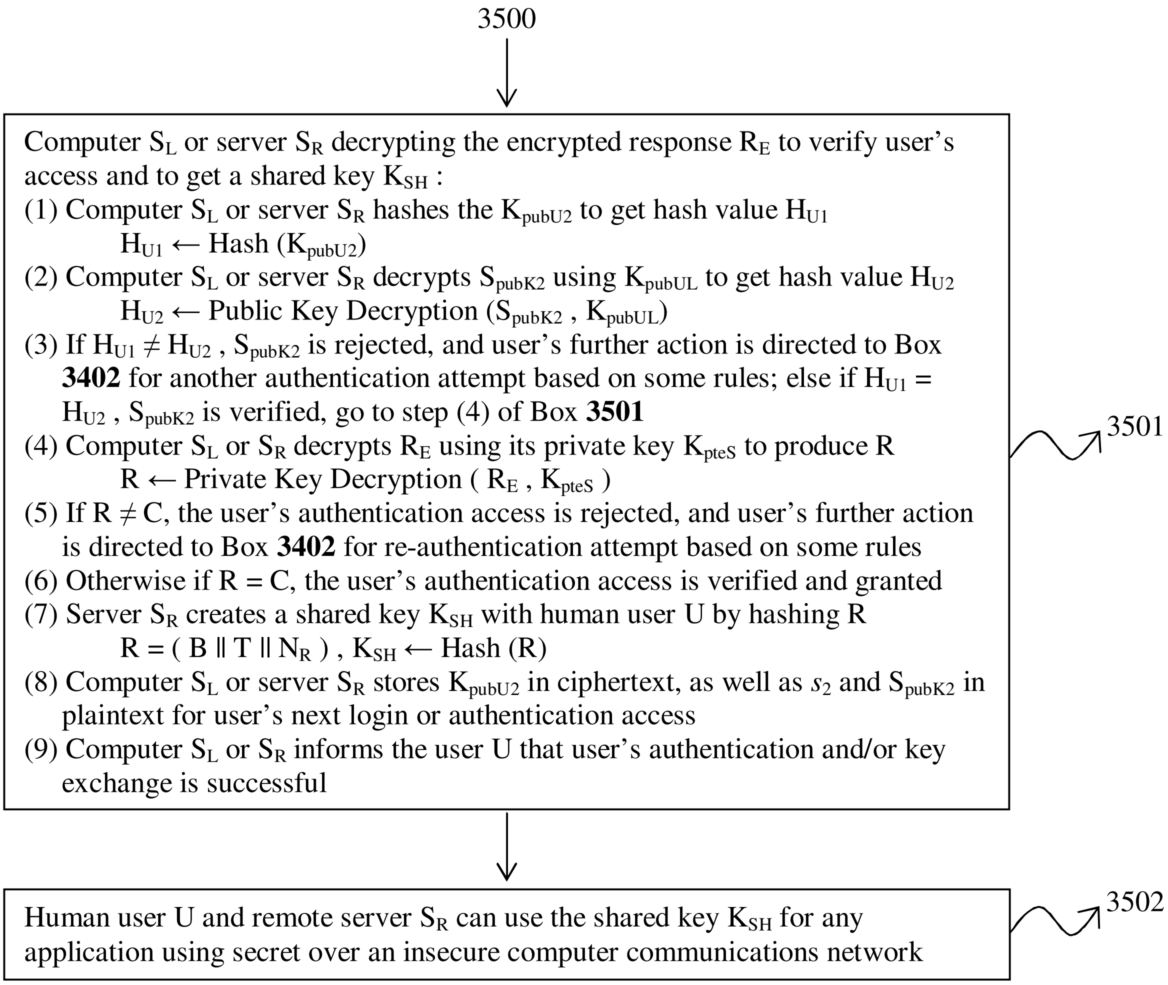


A registered human user U attempting to login to an offline/online account:
(1) User accesses a local computer system S_L or remote server S_R
(2) User sends one's registered username ID to computer S_L or S_R

Computer S_L or S_R creating a challenge C for user to gain authentication access:
(1) Computer S_L or S_R looks up the corresponding K_{pubU}, s_1, and S_{pubK} of username ID
(2) Computer S_L or S_R encrypts K_{pubU} using K_{pubU} to produce ciphertext CK_{pubU}
$CK_{pubU} \leftarrow$ Public Key Encryption (K_{pubU} , K_{pubU})
(3) Computer S_L or S_R creates and encrypts a challenge C using an n-bit random bitstream B, timestamp T, and a nonce N_R
$C \leftarrow (B \parallel T \parallel N_R)$, $C_E \leftarrow$ Public Key Encryption (C , K_{pubU})
(4) Computer S_L or S_R signs the concatenation of s_1, CK_{pubU}, and C_E for integrity checking using private key of computer or server K_{pteS} to produce signature S_S
$S_S \leftarrow$ Sign (Hash ($s_1 \parallel CK_{pubU} \parallel C_E$))
(5) Computer S_L or S_R sends s_1, CK_{pubU}, C_E, and S_S to the user through SSL

User decrypting the encrypted challenge C_E to get a response R and shared key K_{SH} :
(1) If S_S is rejected, go to **3400**; else if S_S is verified, go to step (2) of Box **3403**
(2) User generates K_{pteU} and then K_{pubU}, and decrypts CK_{pubU} to get K_{pubU2}
$K_{pteU} \leftarrow$ Hash ($K_P \parallel ID \parallel s_1$) , $K_{pubU} \leftarrow$ Public Key Generation (K_{pteU})
$K_{pubU2} \leftarrow$ Private Key Decryption (CK_{pubU} , K_{pteU})
(3) If $K_{pubU} \neq K_{pubU2}$, go to **3400**; else if $K_{pubU} = K_{pubU2}$, computer S_L or server S_R is authenticated and go to step (4) of Box **3403**
(4) User decrypts the C_E using user's private key K_{pteU} to produce response R
$R \leftarrow$ Private Key Decryption (C_E , K_{pteU})
(5) User creates a shared key K_{SH} with server S_R by hashing R
$R = (B \parallel T \parallel N_R)$, $K_{SH} \leftarrow$ Hash (R)
(6) User encrypts the R using public key K_{pubS} of computer S_L or server S_R to produce encrypted response R_E
$R_E \leftarrow$ Public Key Encryption (R , K_{pubS})
(7) User creates new salt s_2, user's private key K_{pteU2}, and user's public key K_{pubU2} as in Box **3302**
$K_{pteU2} \leftarrow$ Hash ($K_P \parallel ID \parallel s_2$) , $K_{pubU2} \leftarrow$ Public Key Generation (K_{pteU2})
(8) User signs the K_{pubU2} using K_{pteUL} to produce signature S_{pubK2}
$S_{pubK2} \leftarrow$ Sign (K_{pubU2} , K_{pteUL})
(9) User sends R_E , s_2 , K_{pubU2} , and S_{pubK2} to the computer S_L or server S_R through SSL

Figure 11.4 Second model of MePKC authentication scheme between a human user and a computer with features of non-plaintext equivalence, perfect forward secrecy, and optional key exchange scheme (Part 1)



Computer S_L or server S_R decrypting the encrypted response R_E to verify user's access and to get a shared key K_{SH} :

(1) Computer S_L or server S_R hashes the K_{pubU2} to get hash value H_{U1}

$H_{U1} \leftarrow$ Hash (K_{pubU2})

(2) Computer S_L or server S_R decrypts S_{pubK2} using K_{pubUL} to get hash value H_{U2}

$H_{U2} \leftarrow$ Public Key Decryption (S_{pubK2} , K_{pubUL})

(3) If $H_{U1} \neq H_{U2}$, S_{pubK2} is rejected, and user's further action is directed to Box **3402** for another authentication attempt based on some rules; else if H_{U1} = H_{U2} , S_{pubK2} is verified, go to step (4) of Box **3501**

(4) Computer S_L or S_R decrypts R_E using its private key K_{pteS} to produce R

R $\leftarrow$ Private Key Decryption (R_E , K_{pteS})

(5) If R $\neq$ C, the user's authentication access is rejected, and user's further action is directed to Box **3402** for re-authentication attempt based on some rules

(6) Otherwise if R = C, the user's authentication access is verified and granted

(7) Server S_R creates a shared key K_{SH} with human user U by hashing R

$R = (B \parallel T \parallel N_R)$, $K_{SH} \leftarrow$ Hash (R)

(8) Computer S_L or server S_R stores K_{pubU2} in ciphertext, as well as s_2 and S_{pubK2} in plaintext for user's next login or authentication access

(9) Computer S_L or S_R informs the user U that user's authentication and/or key exchange is successful

Human user U and remote server S_R can use the shared key K_{SH} for any application using secret over an insecure computer communications network

Figure 11.5 Second model of MePKC authentication scheme between a human user and a computer with features of non-plaintext equivalence, perfect forward secrecy, and optional key exchange scheme (Part 2)

11.1.4 Re-Authentication Rules

Mutual human-computer authentication for both the first and second models is possible, and it is also extendable to mutual human-human authentication over a computer network. For failed authentication, there are some re-authentication rules for another login attempt and so on. These re-authentication rules include limited time, limited usage amount of a factor, limited number of allowable attempts per unit of time, CAPTCHA activation, secret question(s) and answer(s), as well as password throttling using time, bit length, and cryptosystem, etc.

11.2 MePKC Digital Certificate Having More Than One Asymmetric Key Pair for Different Protection Periods and Password Throttling

11.2.1 Related Works: Digital Certificate and Password Throttling

In using PKC (Public-Key Cryptography), a user needs to bind one's public key with one's identity. The file binding the user's identity and public key is called digital certificate (aka public-key certificate). Digital signature is used to bind the user's identity and public key by an introducer using web of trust or by a trusted third party (TTP) using certification authority (CA).

In the current prior art, there is only one public key per digital certificate. In PKC, different key sizes correspondent to different protection periods. A short key size like RSA-1024 will have to be changed or revoked frequently. Frequent certificate revocation may cause complicated management problems. Hence, a private key has to be steady throughout its validity period to avoid frequent certificate revocation. Successful cracking of encrypted private key, as well as forgetfulness of symmetric key encrypting the private key and partially memorizable private key tend to fail this purpose. Therefore, the ciphertext of the encrypted private key has to be hidden from the public domain.

For online account using split private key cryptosystem, attackers may launch online dictionary attack to the server. The method of locking an account after a pre-set number of unsuccessful login attempts is not practical because it is subject to denial-of-service attack. The follow-up services to re-activate the account through phone and face-to-face communications are tedious and costly. Consequently, split private key cryptosystem was improved by Sandhu, deSa, and Ganesan (2005a) to have the function of password throttling using the increasing complexity of time response and bit length for unsuccessful authentication. The time response will be slower or the bit length of the challenge will be longer whenever a previous login attempt is unsuccessful until a maximum pre-set value tolerable by a user. A slight modification is to measure based on limited number of login attempts per time unit.

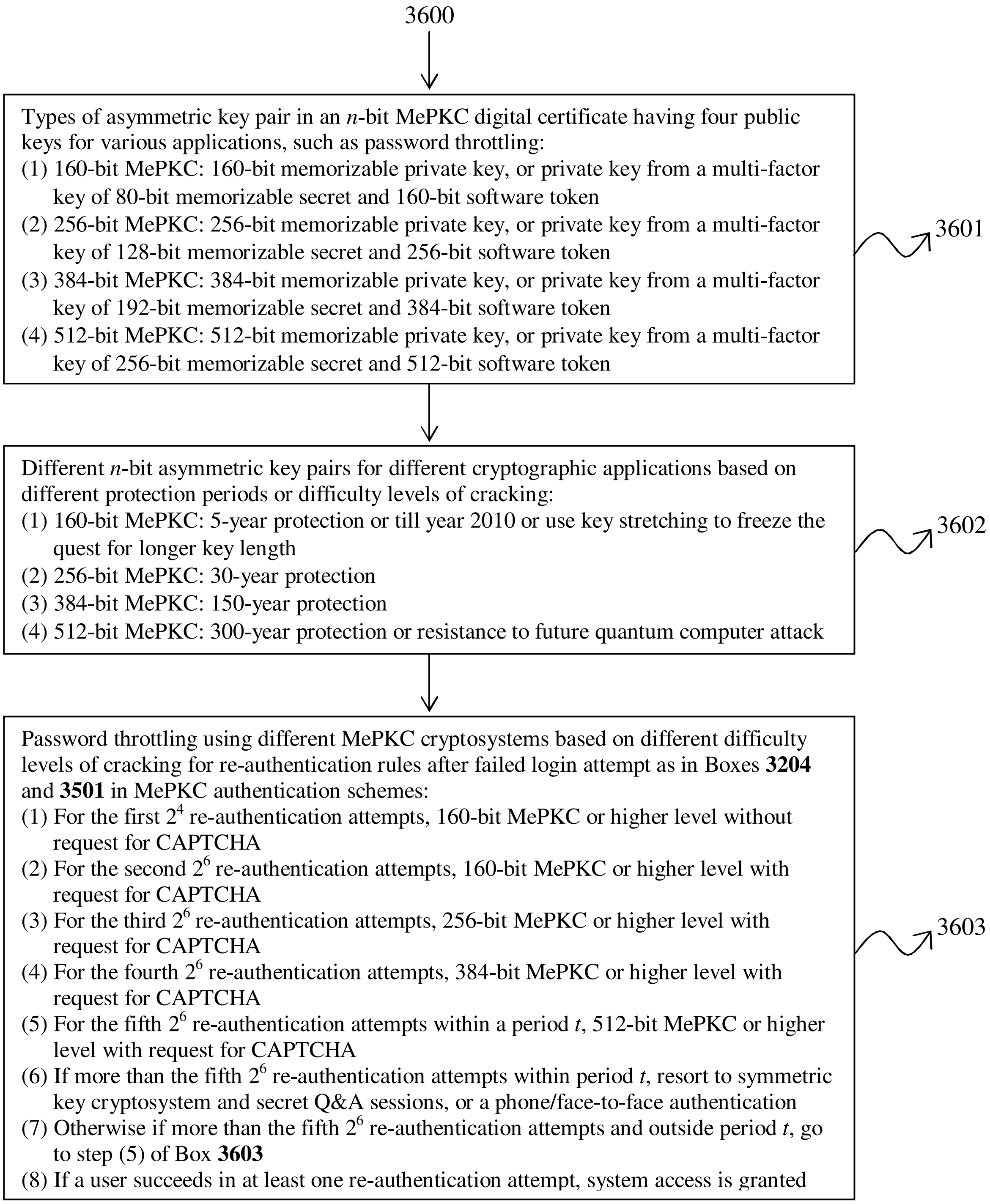
The disadvantage of this method is that a digital certificate with short asymmetric key pair like RSA-1024 will still have to be changed frequently. Another disadvantage is that there is a maximum of time response and processing time like one second that a user can tolerate. A delay of one second adds only by about 20 bits on the platform of contemporary computing technologies. Yet in some password generation systems, key strengthening (aka key stretching) is use to harden a password by hashing a password seed for many rounds of iteration for a pre-set time unit like one second to freeze the demand of better computing technologies for longer key length. It tells that password throttling (Sandhu, deSa & Ganesan, 2005a) using time response may be not tolerable if it is used together with key strengthening.

Hence, there exists a need to improve this method to have lower frequency of certificate revocation and yet fast time response. Moreover, there is a need to have bigger memorizable secret to resist online dictionary attack and malicious server attack over the split private key cryptosystem.

Another method to resist machinery online dictionary attack is to use CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) by asking a user to key in some data presented by a computer that cannot be interpreted by another remotely networked computer trying to attack the account. This method is quite effective but it cannot extend the validity of a digital certificate with short asymmetric key pair like RSA-1024 that is still changed or revoked frequently. Hence, need exists to extend the digital certificate validity to reduce the certificate revocation frequency via a better password throttling method.

11.2.2 MePKC Digital Certificate with Many Asymmetric Key Pairs

Yet in the eighth application 9.1(viii) of the present invention in applying the created big memorizable secret, the multihash key allows the usages of multiple secrets for various applications and this can realize the MePKC digital certificate having more than one asymmetric key pair. Due to technical security and legal factors, a pair of asymmetric key cannot be re-used for different cryptographic schemes like encryption, signature, and authentication. Hence, it is very common for a user to own more than one asymmetric key pair. Here, MePKC digital certificate with four public keys is illustrated in Figure 11.6 for one of its various functions according to private key sizes, protection periods, and difficulty levels of cracking.



Types of asymmetric key pair in an *n*-bit MePKC digital certificate having four public keys for various applications, such as password throttling:

(1) 160-bit MePKC: 160-bit memorizable private key, or private key from a multi-factor key of 80-bit memorizable secret and 160-bit software token
(2) 256-bit MePKC: 256-bit memorizable private key, or private key from a multi-factor key of 128-bit memorizable secret and 256-bit software token
(3) 384-bit MePKC: 384-bit memorizable private key, or private key from a multi-factor key of 192-bit memorizable secret and 384-bit software token
(4) 512-bit MePKC: 512-bit memorizable private key, or private key from a multi-factor key of 256-bit memorizable secret and 512-bit software token

Different *n*-bit asymmetric key pairs for different cryptographic applications based on different protection periods or difficulty levels of cracking:

(1) 160-bit MePKC: 5-year protection or till year 2010 or use key stretching to freeze the quest for longer key length
(2) 256-bit MePKC: 30-year protection
(3) 384-bit MePKC: 150-year protection
(4) 512-bit MePKC: 300-year protection or resistance to future quantum computer attack

Password throttling using different MePKC cryptosystems based on different difficulty levels of cracking for re-authentication rules after failed login attempt as in Boxes **3204** and **3501** in MePKC authentication schemes:

(1) For the first 2^4 re-authentication attempts, 160-bit MePKC or higher level without request for CAPTCHA
(2) For the second 2^6 re-authentication attempts, 160-bit MePKC or higher level with request for CAPTCHA
(3) For the third 2^6 re-authentication attempts, 256-bit MePKC or higher level with request for CAPTCHA
(4) For the fourth 2^6 re-authentication attempts, 384-bit MePKC or higher level with request for CAPTCHA
(5) For the fifth 2^6 re-authentication attempts within a period *t*, 512-bit MePKC or higher level with request for CAPTCHA
(6) If more than the fifth 2^6 re-authentication attempts within period *t*, resort to symmetric key cryptosystem and secret Q&A sessions, or a phone/face-to-face authentication
(7) Otherwise if more than the fifth 2^6 re-authentication attempts and outside period *t*, go to step (5) of Box **3603**
(8) If a user succeeds in at least one re-authentication attempt, system access is granted

Figure 11.6 MePKC digital certificate with four public keys for various applications, such as password throttling

The illustrated public key settings of a MePKC digital certificate are 160, 256, 384, and 512 bits, in which their private keys may be created from multi-factor multilingual key. For re-authentication rules after failed login attempts, password throttling based on cryptosystem is presented as one of its potential main functions. Other password throttling techniques use different periods of response time and lengths of challenge message. After series of password throttling, the authentication scheme may resort to symmetric key cryptosystem and secret Q&A (Questions and Answers) session for limited information access, or phone/face-to-face authentication to re-activate the account. Another potential function is to let the MePKC digital certificate to have at least a bait asymmetric key pair. This bait will detect if there is any criminal crony interested with any MePKC digital certificate.

11.3 Three-Tier MePKC Digital Certificates for Ladder Authentication

11.3.1 Related Works: Digital Certificate and Ladder Authentication

For Internet banking using password the secret for authentication access, usually more than one factor and one authentication process are needed for different services due to the sensitiveness and criticality of monetary matters. For instance, a first symmetric key through computer communications network is needed to login to an Internet banking account. A second random number the secret, that is sent from a bank server to a user's mobile phone through another communication channel, is needed to activate some financial services like fund transfer and utility bill payment, as well as non-financial services like changes of mailing address, email, and phone number. These different authentication processes for different sensitive services of an account is called ladder authentication (Sandhu, deSa & Ganesan, 2006e).

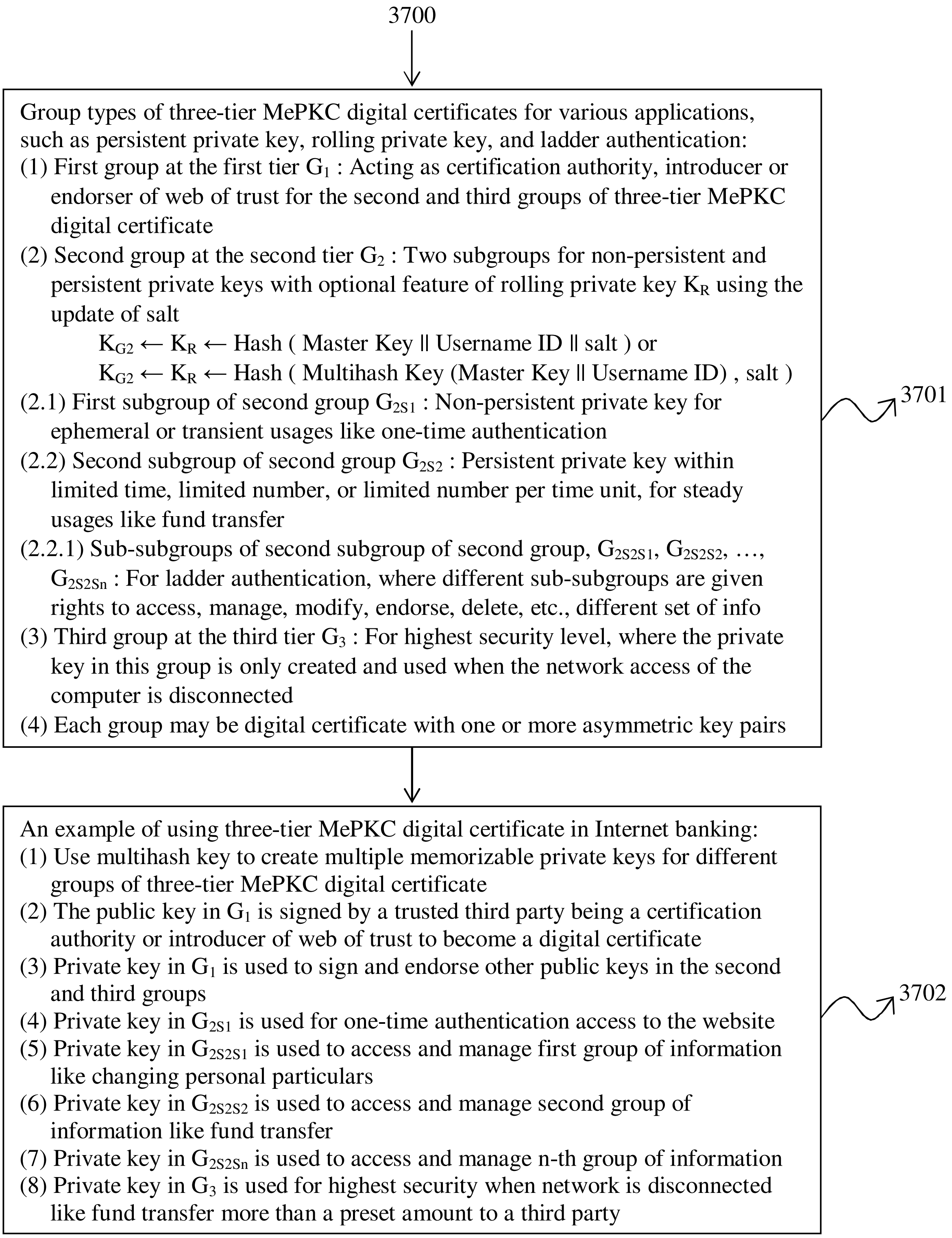
Although this method is effective, it limits to users' with mobile phone and the costs of SMS (Short Message Service) to deliver the random number can be quite a large amount when the Internet banking is prevalent. For example in Malaysia, there are a population of 27 million and an average household size of five members per family in 2007. Let each household have five types of utility bills per month: Water, electricity, one wired phone, and two wireless phones. Then, there are 27 million bills per month throughout Malaysia. If an SMS is charged one cent by the services provider of mobile phone, then it is MYR$3.24 million annually.

The operating costs become higher if a mobile phone is registered overseas. This is a norm of phenomenon for a Malaysian using Singapore Internet banking services, and vice versa. To solve this problem in Singapore, where lots of its residents are occasionally residing overseas, Singapore banks use the one-time-password token (OTP token) like RSA SecurID token. The seeded OTP token creates temporary password with a finite usable life such as thirty seconds. For every cycle of usable life, another temporary password is generated. An authentication server knows the seed and each usable temporary password as well as its usable life, based upon shared algorithms with the OTP token. An overseas user uses the temporary password from the OTP token to replace the random number of an SMS.

Nevertheless, the OTP token is subject to loss, damage, and mobility convenience. Bank will charge the users for replacement of an OTP token due to loss or damage. Currently in Singapore, the replacement cost is SGD$20 per unit of OTP token. Moreover, the temporary password of OTP token is displayed in plaintext mode. Anyone who gets the OTP token can subsequently obtain the temporary password. In a summary, in the current prior art, the ladder authentication methods using SMS of mobile phone and OTP token incur a high operating cost. Hence, there is a need to apply specific PKC digital certificate using fully memorizable private key to implement a cost-saving and yet securer ladder authentication system.

11.3.2 Three-Tier MePKC Digital Certificates for Various Applications

In the ninth application 9.1(ix) of the present invention in applying the created big memorizable secret, three-tier MePKC digital certificates can perform the functions of persistent private key, rolling private key, and ladder authentication as in Figure 11.7. The number of tier can also be other values depending on the design requirements. The first group at the first tier acts as the introducer or endorser for the other groups. The user information of the digital certificates in the second and third groups can be updated easily from time to time.



Group types of three-tier MePKC digital certificates for various applications, such as persistent private key, rolling private key, and ladder authentication:

(1) First group at the first tier G_1 : Acting as certification authority, introducer or endorser of web of trust for the second and third groups of three-tier MePKC digital certificate

(2) Second group at the second tier G_2 : Two subgroups for non-persistent and persistent private keys with optional feature of rolling private key K_R using the update of salt

$K_{G2} \leftarrow K_R \leftarrow$ Hash (Master Key || Username ID || salt) or

$K_{G2} \leftarrow K_R \leftarrow$ Hash (Multihash Key (Master Key || Username ID) , salt)

(2.1) First subgroup of second group G_{2S1} : Non-persistent private key for ephemeral or transient usages like one-time authentication

(2.2) Second subgroup of second group G_{2S2} : Persistent private key within limited time, limited number, or limited number per time unit, for steady usages like fund transfer

(2.2.1) Sub-subgroups of second subgroup of second group, G_{2S2S1}, G_{2S2S2}, …, G_{2S2Sn} : For ladder authentication, where different sub-subgroups are given rights to access, manage, modify, endorse, delete, etc., different set of info

(3) Third group at the third tier G_3 : For highest security level, where the private key in this group is only created and used when the network access of the computer is disconnected

(4) Each group may be digital certificate with one or more asymmetric key pairs

An example of using three-tier MePKC digital certificate in Internet banking:

(1) Use multihash key to create multiple memorizable private keys for different groups of three-tier MePKC digital certificate

(2) The public key in G_1 is signed by a trusted third party being a certification authority or introducer of web of trust to become a digital certificate

(3) Private key in G_1 is used to sign and endorse other public keys in the second and third groups

(4) Private key in G_{2S1} is used for one-time authentication access to the website

(5) Private key in G_{2S2S1} is used to access and manage first group of information like changing personal particulars

(6) Private key in G_{2S2S2} is used to access and manage second group of information like fund transfer

(7) Private key in G_{2S2Sn} is used to access and manage n-th group of information

(8) Private key in G_3 is used for highest security when network is disconnected like fund transfer more than a preset amount to a third party

Figure 11.7 Three-tier MePKC digital certificates for various applications, such as persistent private key, rolling private key, and ladder authentication

The second group has two subgroups with the optional feature of rolling private key, which means regular replacement of asymmetric key pair. Each rolling private key is updated when the salt value is updated according to one of the two equations, where the first equation is from the second model of the MePKC authentication scheme as in Figures 11.3-11.5, and the second equation applies the multihash key.

For the private key in the first subgroup of the second group, it is non-persistent in computer memory for ephemeral or transient usages like one-time authentication. For the private key in the second subgroup of the second group, it is persistent in computer memory within limited time, limited number, or limited number per time unit, for steady usages like changing personal particulars, fund transfer and bill payment.

The second subgroup of second group can be further divided into many sub-subgroups for ladder authentication to resist MITM (Man-In-The-Middle) attacks. The private key in the first, second, third, …, n-th sub-subgroups of the second subgroup of the second group may be used to independently access, manage, modify, endorse, delete, etc., first, second, third, …, n-th groups of information, respectively.

The first and second groups can function to alternate and complement the current prior art of authentication scheme in Internet banking, where first authentication using password, and second authentication using SMS random number or one-time-password token (OTP token). This SMS random number is called specifically as TAC (Transaction Authorisation Code or Transaction Authentication Code), TAP (Transaction Authorization Pin), Auth Code, and Authorization Code in Internet banking as a second layer of protection. The ladder authentication using different groups from different tiers of MePKC digital certificate can be applied to Internet banking, as well as online share trading.

For highest security, the private key of the third group is only used when the networked computer is offline or disconnected from the computer communications network like Internet and LAN. When anonymity feature is needed, then at least an additional set of MePKC digital certificate from the first, second, and/or third group is needed.

11.4 Anti-Phishing Using MePKC Authentication Schemes

According to APWG (Anti-Phishing Working Group) (2008), the numbers of crimeware-spreading URLs infecting PCs with password-stealing code rose 93% in 2008 Q1 to 6500 sites, nearly double the previous high of November, 2007, and an increase of 337 percent from the number detected end of 2007 Q1. Patel and Luo (2007) took a closer look at phishing, which is a serious crime for the Internet banking (Hilley, 2006; Mannan & van Oorschot, 2007).

Lately, Microsoft has declared war against the phishing (Hunter, 2006) by taking actions like recruiting the anti-phishing crusader (CastleCops, No date; McMillan, 2008a). There is also some key management tools to resist the phishing attack like Passpet (Yee & Sitaker, 2006).

Here, it is to note that when the seventh to ninth applications 9.1(vii)-9.1(ix) of the present invention in applying the created big memorizable secret are applied for authentication to access online accounts like Internet banking and online share trading, the phishing attack can be thwarted easily since there are no shared secret at all and no transmission of secret key between the human and computer over an insecure computer communications network (Ford, 1994).

CHAPTER 12 APPLICATIONS OF BIG SECRET & MePKC (PART 4)

12.1 Archiving the Voice/Video Calls of Wired/Wireless Phones

12.1.1 Related Works

Yet there is another important application of PKC (Public-Key Cryptography) using fully big memorizable secret. Here, the application of secret to mobile phone (aka wireless phone, cellular phone, cell phone, and hand phone) (Lee, 1995) is discussed. Since the invention of wireless telephone in the 1907 by Nathan B. Stubblefield (1908) in the US Patent: 887357 “Wireless Telephone”, filed on 5 April 1907, its number of functions keeps on increasing until now that even there is camera capturing real-time image and making video call a reality.

One of the many inventions is by Charles A. Gladden and Martin H. Parelman (1979) entitled “Rapidly Deployable Emergencey Communication System” and to introduce the concepts of frequency reuse and handoff. For mobile phone, it is possible to record SMS, voice mail, local image and video. A user needs a passcode (aka pin) the secret to access the voice mailbox. However, it is yet impossible to download voice mail from a website and record interactive voice and video calls. Moreover, the memory of mobile phone is limited due to its size and publicly affordable selling price.

Nevertheless, there are commercial activities, legal cases, personal matters, etc., that are constrained by physical distance and the most convenient communications channel is a phone connection. Here, normally a wired phone will be used together with a recorder to keep a copy of the conversation contents as electronic evidence. However, having every household to own a phone recorder is not cost-effective.

Hence, there exist needs to download voice mail from a website, as well as to record, encrypt, store, access, manage, copy, download, and decrypt the interactive voice and video calls from a website as electronic evidence. Distributed servers located in the CO (Central Office) (aka telephone exchange) of wired phone and

MTSO (Mobile Telephone Switching Office) (DeVaney, Harper & Short, 1987) of wireless phone shall be fully utilized for recording storage of voice and video calls. Computer password authentication protocol using symmetric key cryptosystem, PKC, or MePKC shall be used to access, manage, and download the recorded voice mail, voice and video calls.

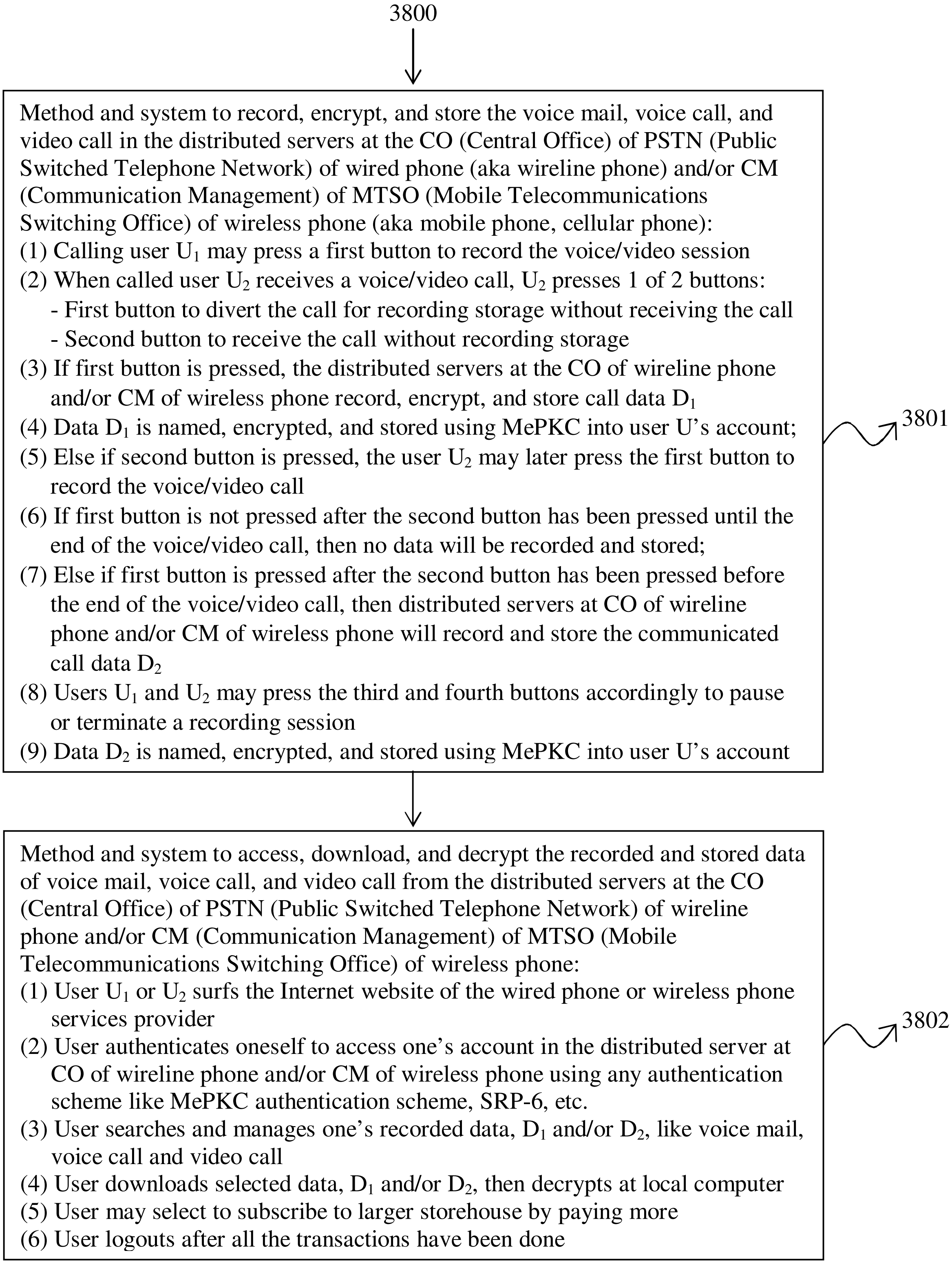
12.1.2 The Proposed Model

In the tenth application 9.1(x) of the present invention in applying the created big memorizable secret, MePKC authentication scheme is used to access a user online account storing the recorded data like voice mail, voice call, and video call of wired phone (aka wireline phone) and wireless phone (aka handphone, mobile phone, wireless phone, cellular phone, cell phone) as in Figure 12.1.

A user's handphone has two buttons to select the call modes. For calling user, if a first button is pressed, then a voice/video session will be recorded and stored at the distributed server. For called user, if the first button is pressed, the voice/video call will be diverted to recording mode directly without receiving the call. Otherwise if second button is pressed, the voice/video call of called user is received and there is interaction between the calling and called users.

After the second button has been pressed, if the first button of called user is not pressed until the end of a call, then no data will be recorded. Otherwise if the first button of called user is pressed after the second button has been pressed, then the following communicated data like voice, image, and video is recorded, encrypted, and stored. Yet calling and called users may press the third and fourth buttons accordingly to pause or terminate a recording session.

The distributed servers at the CO (Central Office) of PSTN (Public Switched Telephone Network) of wired phone and/or CM (Communication Management) of MTSO (Mobile Telecommunications Switching Office) of wireless phone records, encrypts using MePKC, and stores the communicated voice/video call between the calling and called parties. The voice/video data is named, encrypted using MePKC, and saved into the user account.



Method and system to record, encrypt, and store the voice mail, voice call, and video call in the distributed servers at the CO (Central Office) of PSTN (Public Switched Telephone Network) of wired phone (aka wireline phone) and/or CM (Communication Management) of MTSO (Mobile Telecommunications Switching Office) of wireless phone (aka mobile phone, cellular phone):

(1) Calling user U_1 may press a first button to record the voice/video session

(2) When called user U_2 receives a voice/video call, U_2 presses 1 of 2 buttons:
- First button to divert the call for recording storage without receiving the call
- Second button to receive the call without recording storage

(3) If first button is pressed, the distributed servers at the CO of wireline phone and/or CM of wireless phone record, encrypt, and store call data D_1

(4) Data D_1 is named, encrypted, and stored using MePKC into user U's account;

(5) Else if second button is pressed, the user U_2 may later press the first button to record the voice/video call

(6) If first button is not pressed after the second button has been pressed until the end of the voice/video call, then no data will be recorded and stored;

(7) Else if first button is pressed after the second button has been pressed before the end of the voice/video call, then distributed servers at CO of wireline phone and/or CM of wireless phone will record and store the communicated call data D_2

(8) Users U_1 and U_2 may press the third and fourth buttons accordingly to pause or terminate a recording session

(9) Data D_2 is named, encrypted, and stored using MePKC into user U's account

Method and system to access, download, and decrypt the recorded and stored data of voice mail, voice call, and video call from the distributed servers at the CO (Central Office) of PSTN (Public Switched Telephone Network) of wireline phone and/or CM (Communication Management) of MTSO (Mobile Telecommunications Switching Office) of wireless phone:

(1) User U_1 or U_2 surfs the Internet website of the wired phone or wireless phone services provider

(2) User authenticates oneself to access one's account in the distributed server at CO of wireline phone and/or CM of wireless phone using any authentication scheme like MePKC authentication scheme, SRP-6, etc.

(3) User searches and manages one's recorded data, D_1 and/or D_2, like voice mail, voice call and video call

(4) User downloads selected data, D_1 and/or D_2, then decrypts at local computer

(5) User may select to subscribe to larger storehouse by paying more

(6) User logouts after all the transactions have been done

Figure 12.1 Operations to record, store, access, manage, and download the voice mail, voice call, and video call in the distributed servers at the CO of PSTN of wireline phone and/or CM of MTSO of wireless phone

The user can then surf the website of the wired phone and wireless phone services provider to access one's account using MePKC authentication scheme or other methods. Upon gaining access to the user account, the user may be optionally required to gain a MePKC ladder authentication to further manage and download the recorded and stored voice mail, voice call, and video call.

After downloading the encrypted data to a local computer, the user can decrypt the data using MePKC schemes like hybrid encryption scheme of PKC and symmetric key cryptography, where a symmetric key used to encrypt the voice/video call is encrypted by a public key. Likewise, this method can be extended to other online electronic data storage using MePKC authentication scheme.

12.2 Multipartite Electronic Commerce Transactions Using MePKC

12.2.1 Related Works

And yet there is crucial cryptosystem using secret to be improved soonest possible. This cryptosystem is the current prevalent electronic commerce (aka e-commerce) transactions (Kini & Choobineh, 1998; Konrad, Fuchs & Barthel, 1999; Ahuja, 2000; H. M. Deitel, P. J. Deitel & Nieto, 2001; Ford & Baum, 2001; Kang, Park & Koo, 2003; Lee, 2006a). In the current prior art, the electronic commerce transactions operate in series of bipartite communication mode using credit card and password the secret.

Once a user has selected a list of products to be purchased online at a certain website (H. M. Deitel, P. J. Deitel & Nieto, 2000), normally a credit card, such like MasterCard or VISA, is then used to pay the bill, by sending the credit card number and an optional secure code behind the card to the online merchant. For more security, password the secret protecting the credit card may be requested by some merchants. Examples of the services providers of credit card password are PayPal, MasterCard SecureCode, and Verified by VISA. Graefe, Lashley, Guimaraes, Guodabia, Gupta, Henry, and Austin (2007) discussed on the credit card transaction security.

Besides merchant and credit card verifier for password, sometimes there exists online loyalty point website demanding for another password authentication. Hence, there are at least three rounds of bipartite communications for different stages of authentication. In fact, a comprehensive electronic commerce transaction involves many other entities such as merchant's bank, customer's bank, insurance company, various departments of local, state, and federal governments, transportation agent, storehouse agent, and so on. Each of this entity is now either usually paired with merchant or rarely customer to one round of bipartite communication to initiate and endorse a sub-process of an electronic commerce transaction.

Here, it can be observed that every individual round of bipartite communications using token of credit card number and/or secret of a symmetric key is not so secure and effective. It is in fact quite redundant and time-wasting. The nature of an electronic commerce transaction is in fact a multipartite communication.

In dealing with cryptography and multipartite communications, there is a branch of knowledge called BGP (Byzantine Generals Problem) (Lamport, Shostak & Pease, 1982; Fischer & Lynch, 1982; Lamport, 1983; Aguilera & Toueg, 1999; Pathak & Iftode, 2006). BGP involves a group of entities where loyal entities have to reach a common agreement called BA (Byzantine Agreement) (Pease, Shostak & Lamport, 1980; Yan & Chin, 1988; Meyer & Pradhan, 1991; Yan, Chin & Wang, 1992; Siu, Chin & Yang, 1998a, 1998b; Yan, Wang & Chin, 1999; Yan & Wang, 2005b; Fitzi & Hirt, 2006) at the end of a sufficient round of message exchanges, regardless of the malicious and arbitrary messages communicated by faulty entities.

The solution of BGP is known as BAP (Byzantine Agreement Protocol), in which BA can be successfully achieved based on the provided functions of PKC (Public-Key Cryptography) like access control, authentication, non-repudiation, and integrity. However, PKC popularity has to be boosted up by using fully big memorizable secret to realize the MePKC. Faulty node detection and identification are also needed (Wang & Yan, 2000; Yan & Wang, 2005a).

There are various types of available BAP. For the entities of electronic commerce, they can be basically partitioned into three groups: Essential, government, and non-essential groups. Here, there is a BAP also optimally divides a network of entities into three partitions. This specific BAP is called tripartite ANN based BAP (Tripartite Artificial Neural Network Based BAP) (aka Tripartite BAP-ANN or Tripartite BAP with ANN) (Lee & Ewe, 2003) and developed from ANN based BAP (Wang & Kao, 2001; Lee & Ewe, 2001, 2002, 2007b, 2007c; Lee, 2003).

The ANN here functions as a classifier and provides majority function over rows and columns of MEM (Message Exchange Matrix) formed from three message exchange rounds of Byzantine communications. For more details of ANN based BAP and tripartite ANN based BAP, please refer to a master's thesis published on 25 October 2002 at Multimedia University, Malaysia, entitled "Artificial Neural Network Based Byzantine Agreement Protocol" by Kok-Wah Lee @ Xpree Jinhua Li (2003).

For the functioning, implementation, and optimization of ANN (Artificial Neural Network), these literatures (Jacobs, 1988; Nguyen & Widrow, 1989, 1990; Cooke & Lebby, 1998; Haykin, 1999) can be referred.

Again to emphasize here, e-commerce transaction involves multipartite communications by nature and not many rounds of bipartite communications. The BGP can model this multipartite cryptography problem of electronic commerce. BAP is the solution of BGP, and hence multipartite communications of electronic commerce. Tripartite ANN based BAP is well-suited to a network of e-commerce entities divided into three groups.

Hence, there exists a need to realize e-commerce transaction based on multipartite communications of BGP and BAP using MePKC, wherein the main purposes are to speed up the processing time from many rounds of bipartite communications and to rely on stronger security protection than the current prior art using symmetric key cryptography.

12.2.2 Artificial Neural Network Based Byzantine Agreement Protocol (ANN Based BAP)

In the eleventh application 9.1(xi) of the present invention in applying the created big memorizable secret, MePKC cryptographic schemes like encryption and signature schemes are used in the method and system of multipartite electronic commerce (aka e-commerce) transactions using tripartite ANN based BAP (Artificial Neural Network Based Byzantine Agreement Protocol) (aka tripartite BAP-ANN (Tripartite BAP with ANN)) as in Figures 12.2-12.7(39-44) and article "Faulty Node Detection in the Tripartite ANN based BAP" by Kok-Wah Lee and Hong-Tat Ewe (2003). The MePKC provides the security like confidentiality, integrity, authentication, access control, and non-repudiation to the tripartite ANN based BAP. Other BAP can also be used for the multipartite e-commerce transactions.

Figure 12.2a shows the operating stages of a basic ANN based BAP. Figures 12.2b-12.2c show the FCN (Fully Connected network) model and ANN architecture for 4-node distributed network. The number of entities involved in the e-commerce ranges from 4 to more than 30.

The simplest network of an e-commerce model includes merchant, customer, bank, and a credit card company. For a big e-commerce model, it can be observed that the partitioning of the large network into a few groups for k-partite ANN based BAP is more efficient. This is because the bottleneck of processing time is the number of exchanged messages that needs to undergo the MePKC encryption, decryption, signing, and verifying processes. It is well-known that the operating time of PKC is so slow that it is 1000 times slower than the symmetric key cryptosystem.

12.2.3 Entity Partitioning in the Electronic Commerce Transactions

From Figures 12.3a-12.3b and 12.4b, it is known that tripartite partitioning is the optimal k-partite ANN based BAP. Figure 12.4a shows the way to partition a network into three partitions. Furthermore, from Figure 12.5, it is shown that the e-commerce entities can be basically divided into three groups: Essential group, government group, and non-essential group. For the first group, the entities of merchant and customer are critical and cannot be replaced; whereas other entities are non-critical and can be replaced. For the second group, all the entities are critical and cannot be replaced. For the third group, all the entities are non-critical and can be replaced. The source node now is the customer to confirm or cancel a buy order.

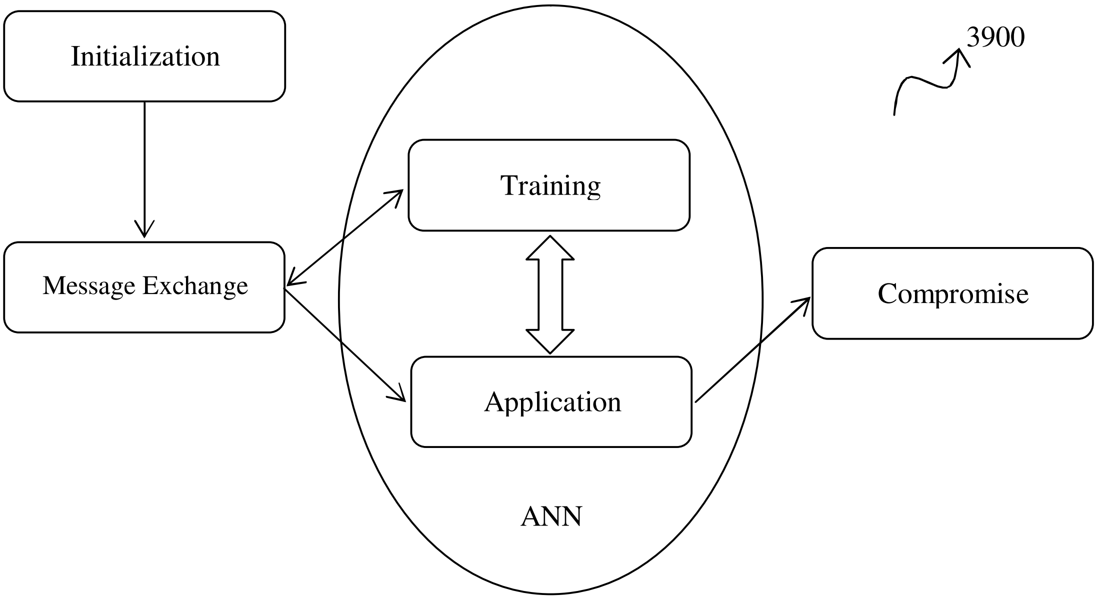
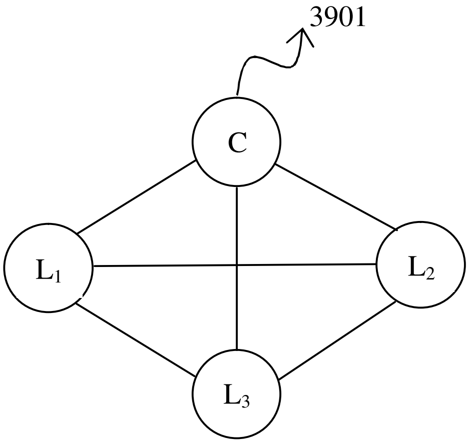


Figure 12.2a Block diagram of ANN based BAP



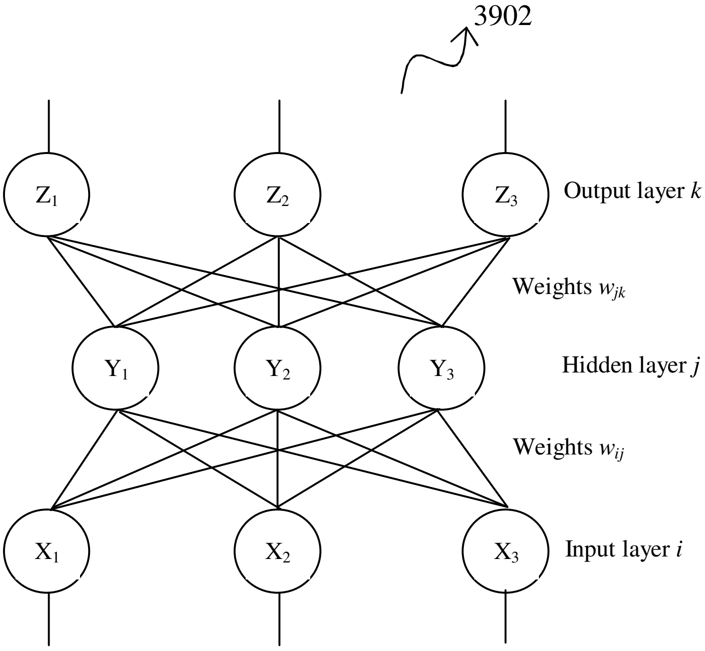


Figure 12.2b FCN model of 4-node distributed network

Figure 12.2c ANN model of 4-node distributed network

Figure 12.2 ANN based BAP and its smallest model of 4-node distributed network

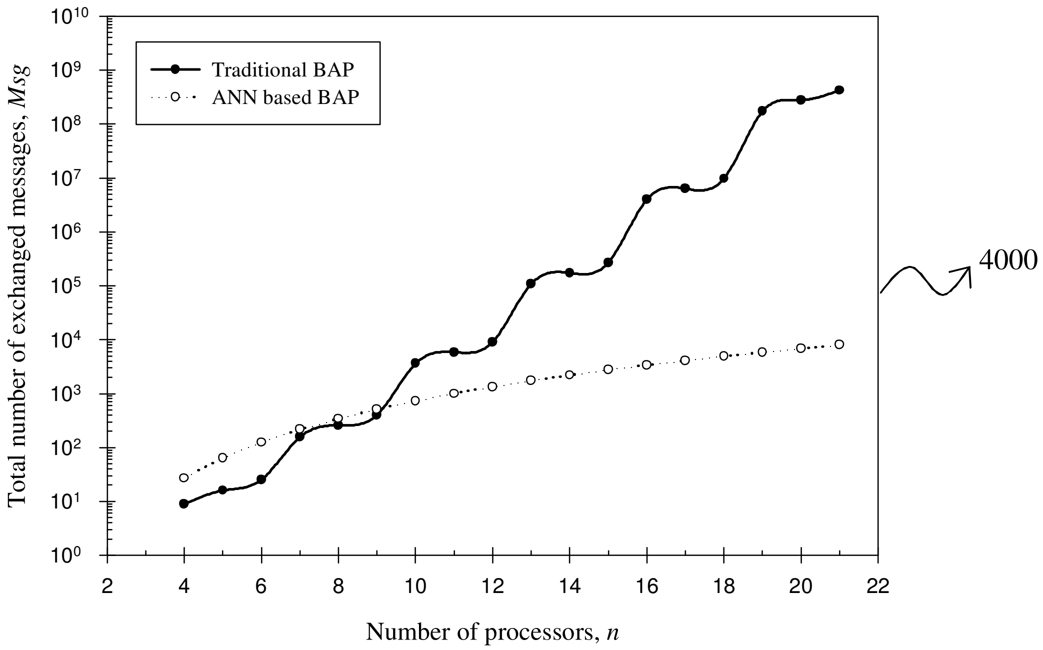


Figure 12.3a Traditional BAP and basic ANN based BAP

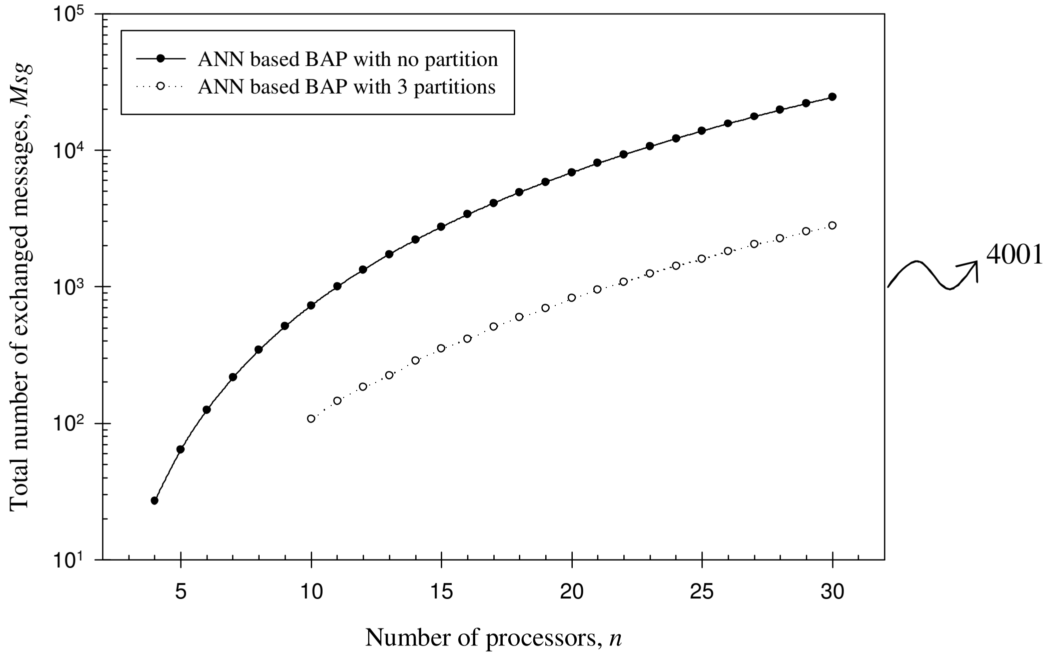


Figure 12.3b Basic ANN based BAP and tripartite ANN based BAP

Figure 12.3 Total number of exchanged messages for different types of BAP

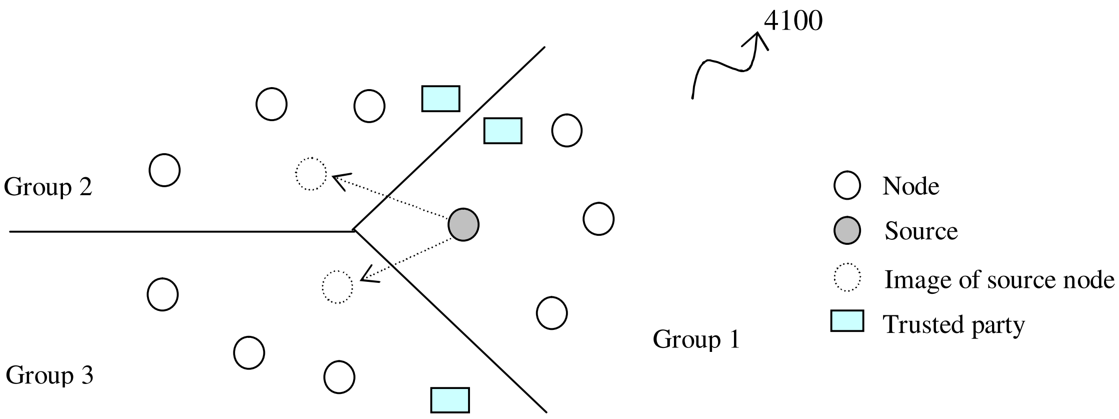


Figure 12.4a Partitioning of a 10-node distributed network into three groups

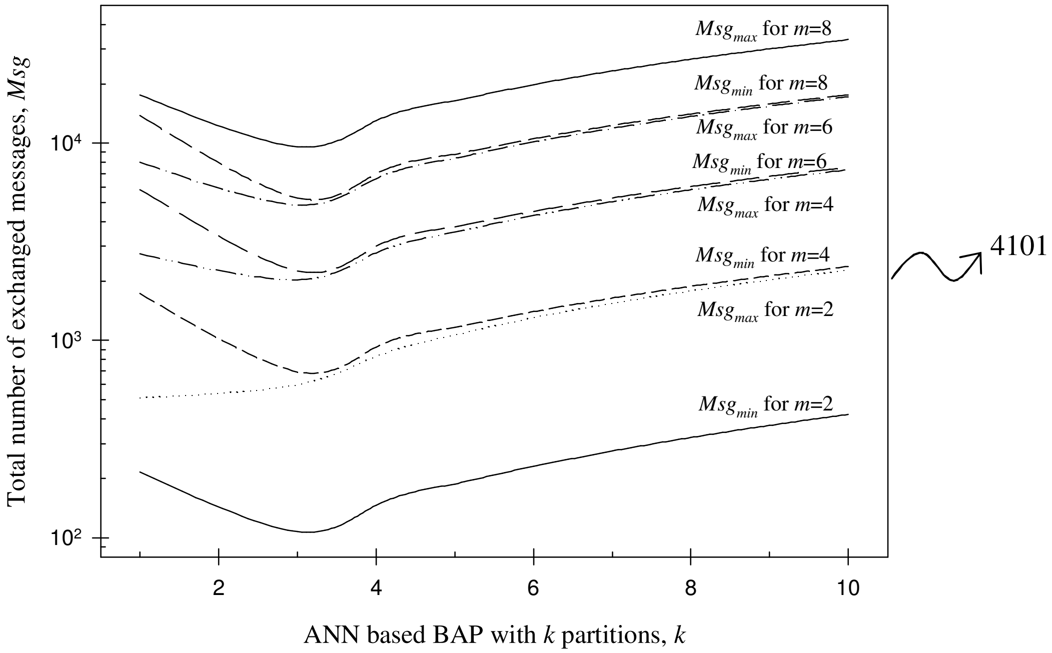
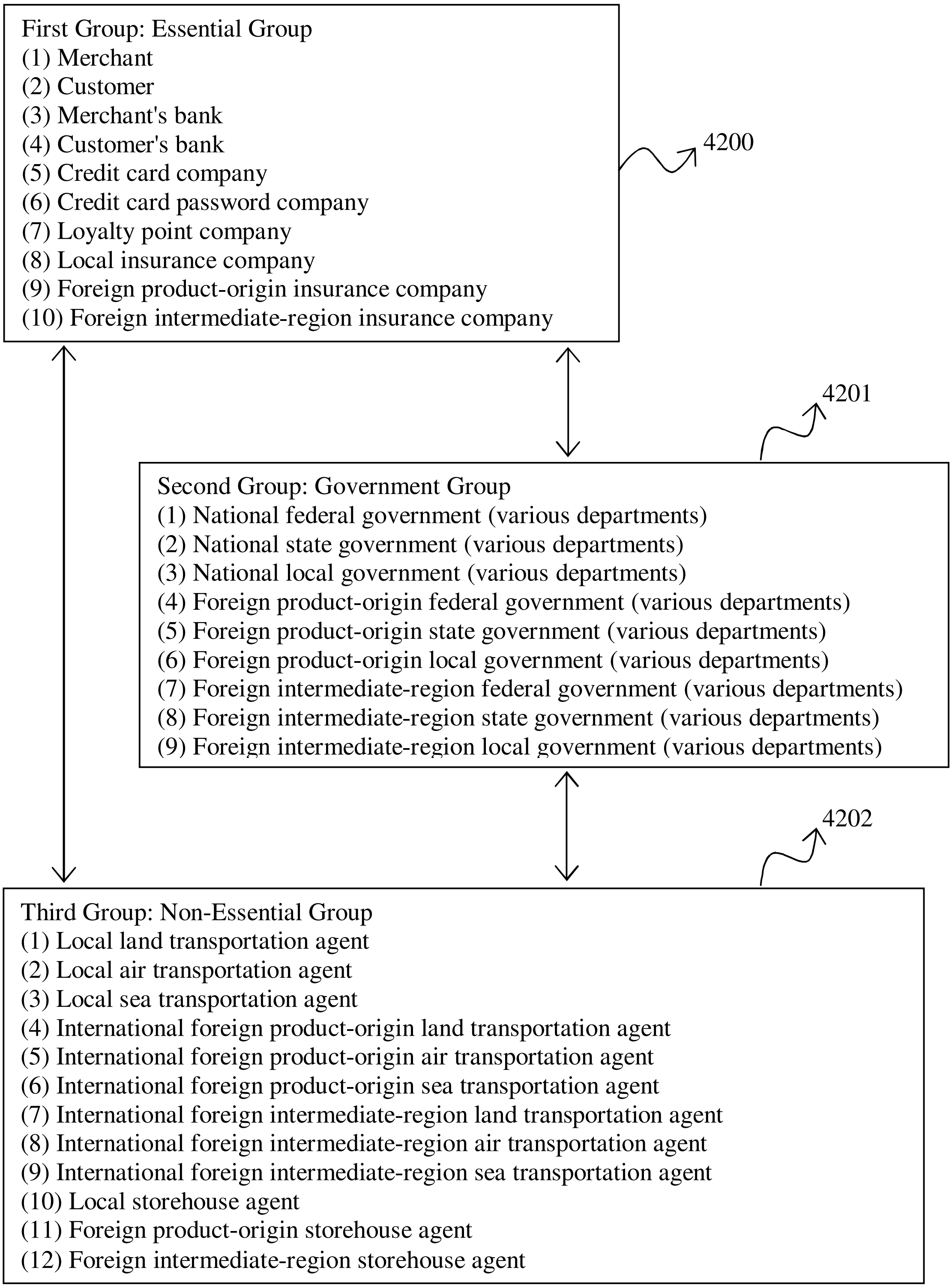


Figure 12.4b Optimal selection of network partitioning for tripartite ANN based BAP

Figure 12.4 Partitioning of a distributed network and its optimal partitioning selection

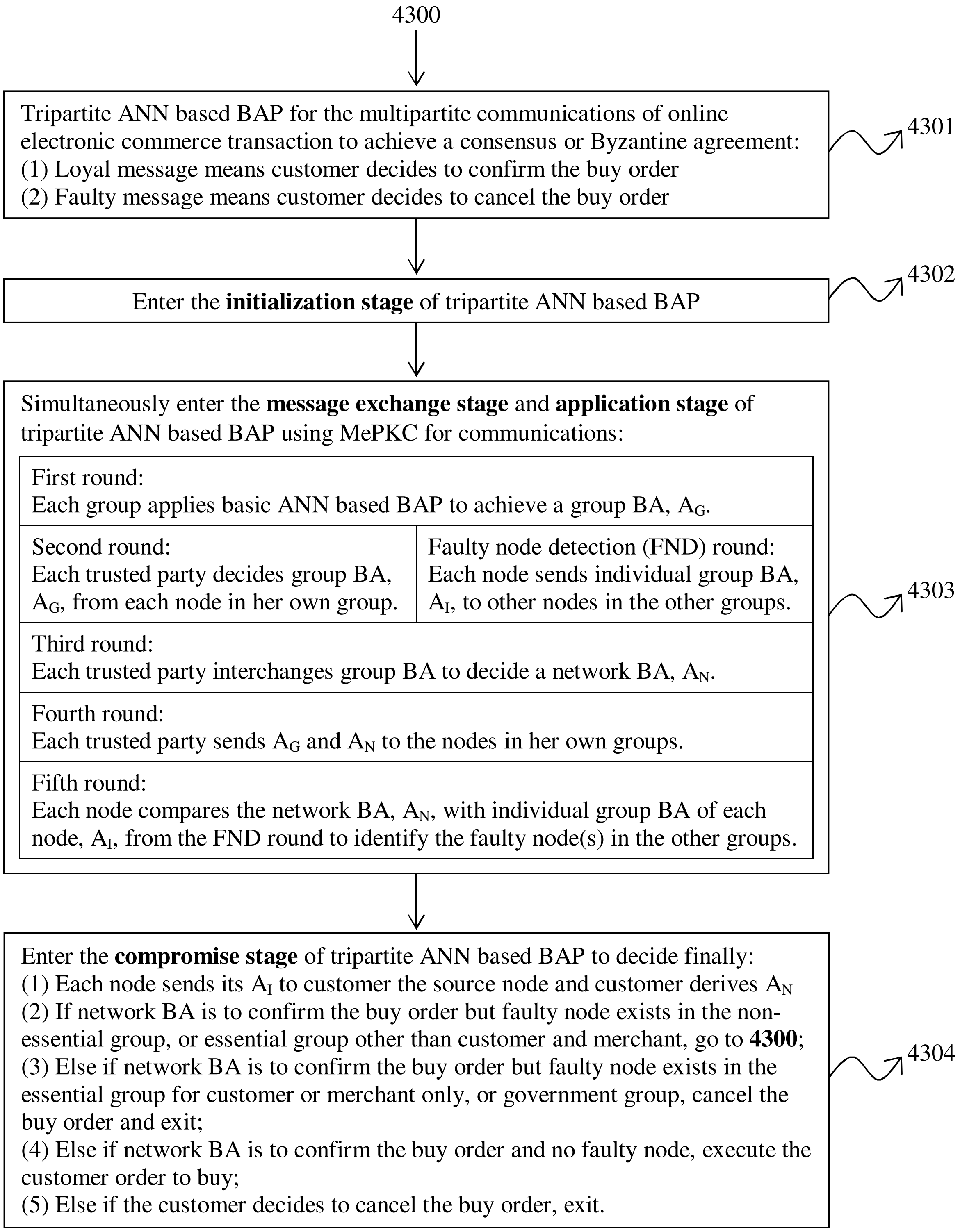


First Group: Essential Group
(1) Merchant
(2) Customer
(3) Merchant's bank
(4) Customer's bank
(5) Credit card company
(6) Credit card password company
(7) Loyalty point company
(8) Local insurance company
(9) Foreign product-origin insurance company
(10) Foreign intermediate-region insurance company

Second Group: Government Group
(1) National federal government (various departments)
(2) National state government (various departments)
(3) National local government (various departments)
(4) Foreign product-origin federal government (various departments)
(5) Foreign product-origin state government (various departments)
(6) Foreign product-origin local government (various departments)
(7) Foreign intermediate-region federal government (various departments)
(8) Foreign intermediate-region state government (various departments)
(9) Foreign intermediate-region local government (various departments)

Third Group: Non-Essential Group
(1) Local land transportation agent
(2) Local air transportation agent
(3) Local sea transportation agent
(4) International foreign product-origin land transportation agent
(5) International foreign product-origin air transportation agent
(6) International foreign product-origin sea transportation agent
(7) International foreign intermediate-region land transportation agent
(8) International foreign intermediate-region air transportation agent
(9) International foreign intermediate-region sea transportation agent
(10) Local storehouse agent
(11) Foreign product-origin storehouse agent
(12) Foreign intermediate-region storehouse agent

Figure 12.5 Partitioning of the entities involved in the electronic commerce transactions into three groups: Essential group, government group, and non-essential group



Tripartite ANN based BAP for the multipartite communications of online electronic commerce transaction to achieve a consensus or Byzantine agreement:
(1) Loyal message means customer decides to confirm the buy order
(2) Faulty message means customer decides to cancel the buy order

Enter the **initialization stage** of tripartite ANN based BAP

Simultaneously enter the **message exchange stage** and **application stage** of tripartite ANN based BAP using MePKC for communications:

First round:
Each group applies basic ANN based BAP to achieve a group BA, A_G.

Second round:
Each trusted party decides group BA, A_G, from each node in her own group.

Faulty node detection (FND) round:
Each node sends individual group BA, A_I, to other nodes in the other groups.

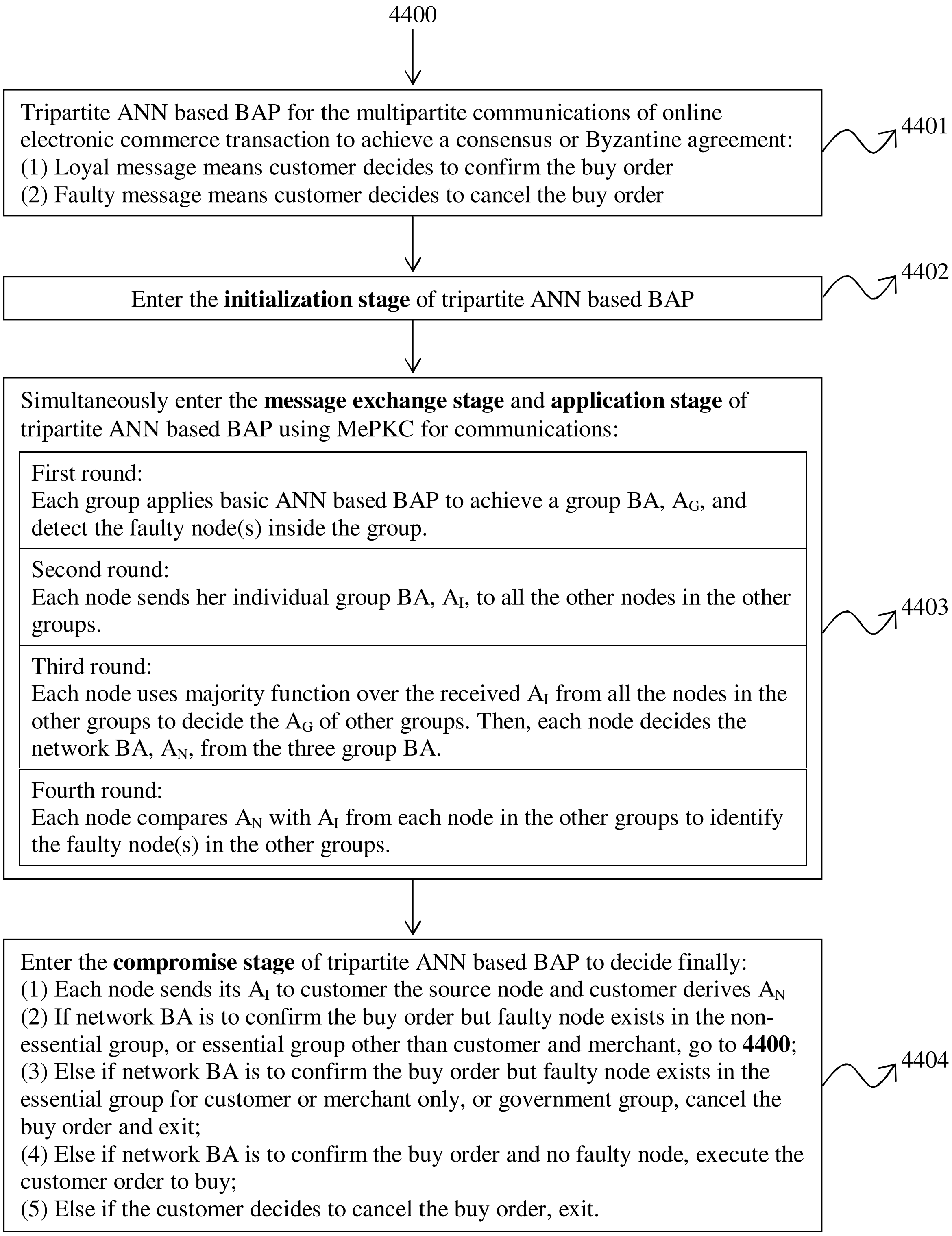
Third round:
Each trusted party interchanges group BA to decide a network BA, A_N.

Fourth round:
Each trusted party sends A_G and A_N to the nodes in her own groups.

Fifth round:
Each node compares the network BA, A_N, with individual group BA of each node, A_I, from the FND round to identify the faulty node(s) in the other groups.

Enter the **compromise stage** of tripartite ANN based BAP to decide finally:
(1) Each node sends its A_I to customer the source node and customer derives A_N
(2) If network BA is to confirm the buy order but faulty node exists in the non-essential group, or essential group other than customer and merchant, go to **4300**;
(3) Else if network BA is to confirm the buy order but faulty node exists in the essential group for customer or merchant only, or government group, cancel the buy order and exit;
(4) Else if network BA is to confirm the buy order and no faulty node, execute the customer order to buy;
(5) Else if the customer decides to cancel the buy order, exit.

Figure 12.6 Tripartite ANN based BAP with trusted party and faulty node detection for multipartite electronic commerce transaction using MePKC cryptographic schemes for communications



Tripartite ANN based BAP for the multipartite communications of online electronic commerce transaction to achieve a consensus or Byzantine agreement:
(1) Loyal message means customer decides to confirm the buy order
(2) Faulty message means customer decides to cancel the buy order

Enter the **initialization stage** of tripartite ANN based BAP

Simultaneously enter the **message exchange stage** and **application stage** of tripartite ANN based BAP using MePKC for communications:

First round:
Each group applies basic ANN based BAP to achieve a group BA, A_G, and detect the faulty node(s) inside the group.

Second round:
Each node sends her individual group BA, A_I, to all the other nodes in the other groups.

Third round:
Each node uses majority function over the received A_I from all the nodes in the other groups to decide the A_G of other groups. Then, each node decides the network BA, A_N, from the three group BA.

Fourth round:
Each node compares A_N with A_I from each node in the other groups to identify the faulty node(s) in the other groups.

Enter the **compromise stage** of tripartite ANN based BAP to decide finally:
(1) Each node sends its A_I to customer the source node and customer derives A_N
(2) If network BA is to confirm the buy order but faulty node exists in the non-essential group, or essential group other than customer and merchant, go to **4400**;
(3) Else if network BA is to confirm the buy order but faulty node exists in the essential group for customer or merchant only, or government group, cancel the buy order and exit;
(4) Else if network BA is to confirm the buy order and no faulty node, execute the customer order to buy;
(5) Else if the customer decides to cancel the buy order, exit.

Figure 12.7 Tripartite ANN based BAP without trusted party but still with faulty node detection for multipartite electronic commerce transaction using MePKC cryptographic schemes for communications

12.2.4 Tripartite ANN Based BAP with Trusted Party

Figure 12.6 shows a first implementation example of using BAP for the multipartite e-commerce transaction having customer as the only source node. Individual group BA, A_I, of each node equals to group BA, A_G, for loyal nodes but not faulty nodes. Yet in a second implementation, both customer and merchant can be source nodes for two independent Byzantine communications of e-commerce, where one is the customer confirming the money payment for the buy order, and another one is the merchant confirming the product/service delivery for the buy order. For a very brief introduction, please refer to Lee (2006a).

12.2.5 Tripartite ANN Based BAP without Trusted Party

And yet in another third implementation as in Figure 12.7, the trusted parties can be excluded if the individual group BA of each node is broadcasted to the nodes of other groups and used directly to derive the network BA.

12.3 Trust Boosting of MePKC Digital Certificate by Using More than One Certification Authority and/or Introducer of Trust of Web

12.3.1 Related Works: Risks of Public Key Infrastructure

The applications of PKI (Public Key Infrastructure) (Kuhn, Hu, Polk & Chang, 2001) in healthcare, finance, government, communications, etc., are presented by Kapil Raina (2003). Meanwhile, for the applications of PKI in the Internet protocols, one can refer to a book "Cryptography and Public Key Infrastructure on the Internet" by Klaus Schmeh (2001). For the details operations on how a user applies for a digital certificate through a CA (Certification Authority), one can refer to a book "PKI: Implementing and Managing E-Security" by Andrew Nash, William Duane, Celia Joseph, and Derek Brink (2001). It can be observed in the third book that in the current prior, the CA generates the asymmetric key pair for the user. This is not good because it may have malicious CA attack.

Yet Carl Ellison and Bruce Schneier (2000) discussed 10 PKI risks in their article "Ten Risks of PKI: What You're not Being Told about Public Key

Infrastructure". The first risk on "Who do we trust, and for what?" questions on how well the CA maintains its private keys well. The current digital certificate having only one digital signature to certify its authenticity is not having a strong enough trust. The successful cracking of a CA private key or existence of malicious CA remains as a PKI risk. Corell (2000) had some comments on these PKI risks that smartcard should be in favour.

The third risk on "How secure is the verifying computer?" questions on the possibility of attacker adding its own public key to the list of certificate verification. Again, the current digital certificate having only one digital signature to certify its authenticity is not having a strong enough trust. The sixth risk on "Is the user part of the security design?" questions on the degree of user involvement in the PKI. So far, the user role is not strong in keeping one's secret because the asymmetric key pair is still generated by the CA. A user holds only a symmetric key protecting the private key of the asymmetric key pair. Hence, there exists a need to innovate the PKI to allow user to create asymmetric key pair oneself and boost up the trust level of PKI.

The identity-related crime conspired by an organized crime group is getting serious in today electronically networked info-computer age. One may refer to UNODC (United Nations Office on Drugs and Crime) website to know more about this identity-related crime at [URL: http://www.unodc.org/unodc/en/organized-crime/index.html] (Identity Theft Resource Center (ITRC), No date; "Identity-Related Crime," 2007; Wikipedia Contributors, 2008bh).

Some human interaction models are needed to simulate the group efficiency of the organized crime group to fake the digital certificate. From the simulation, one can design PKI that can make the organized crime group (Lampe, No date; Glick, 1995; Livingston, 1996; "UNODC and Organized Crime," No date; Layman & Potter, 1997; Maxim & Whitehead, 1998; Chen, 2004; Ruan & Wang, 2005; Siegel, 2005; Wang, 2007; "Identity-Related Crime," 2007; He, 2007; United Nations Office on Drugs and Crime (UNODC), 2004, 2008; Wikipedia Contributors, 2008s, 2008af) to be inefficient and hence the PKI trust level can be increased.

Another approach is to enact warning punishment (Weiss & South, 1998; Wikipedia Contributors, 2007r). Bierly, Kolodinsky, and Charette (No date) discussed the relationships of creativity and ethical ideologies. Chatterjee (No date) analyzed the equity ownership and the directors' board attitude for responsibilities, obligations, and crimes. Khatri and Tsang (2003) surveyed and analyzed the antecedents and consequences of cronyism (Wikipedia Contributors, 2008q) in organizations, which helped to know the malice probability of one or more CAs.

Kaneyuki Kurokawa (1997) proposed some very interesting and good human interaction models in his paper entitled "Modeling Human Interactions". The studied models are committee meeting, labour division, exploratory group, and technology transfer. This article has somehow showed the coefficient of inefficiency of Parkinson's Law by Professor Cyril Northcote Parkinson (1958, 2002), in his book "Parkinson's Law: Or the Pursuit of Progress". The coefficient of inefficiency (Wikipedia Contributors, 2008i) ranges from 20 to 22 or more to trigger the phenomena that a human group starts to become inefficient. Hence, there exists a need to apply the results of these human interaction models over the organized crime group to fake digital certificate in order to boost up the trust level of the digital certificate.

There are other Kurokawa's articles on human interaction models (Kurokawa, 1988, 1990, 1991). The Parkinson's Law has also been studied by some other researchers (Aronson & Gerard, 1966; Aronson & Landy, 1967; Landy, McCue & Aronson, 1969; "The Proof of Parkinson," 1969; Wikipedia Contributors, 2008x). Klimek, Hanel, and Thurner (2008) analyzed the efficiency of a ministers' cabinet to show the Parkinson's Law.

12.3.2 Some Human Interaction Models

In the twelfth application 9.1(xii) of the present invention in applying the created big memorizable secret, method and system to boost up the trust level of MePKC digital certificate by using more than one certification authority (CA) and/or introducer of trust of web is designed. When one refers to the Figures 11.6-11.7 for the MePKC digital certificate, one will know that the private key and public key of a user's asymmetric key pair is generated by the user and not the CA. This step can avoid the malicious CA attack by giving the user to fully control one's private key secret, and hence alleviating the sixth risk of Carl Ellison and Bruce Schneier (2000) on “Is the user part of the security design?” questioning on the degree of user involvement in the PKI.

For the first group of the user’s asymmetric key pair of the three-tier MePKC digital certificate as in Figure 11.7, it acts as the introducer of trust of web to the other groups at tiers 2 and 3. For the certification of the first group instead, the current prior art uses a single digital signature from a CA or introducer of trust of web. However when the MePKC prevails, this prior art is not that appropriate in view of the high demand of trust for the first group of three-tier MePKC digital certificate. Innovated approach has to use to build up stronger trust by failing the organized crime to fake MePKC digital certificate.

The possibility that the asymmetric key can be generated by a user allows the user to bind one’s identity, public key, and other data, into a binding file oneself. A user can then request one or more CA and/or introducer of trust of web to sign, certify, and issue digital signature. Every pair of binding file and a CA/introducer’s digital signature acts as a MePKC digital signature. Due to the independent trust of each pair, other users only accept a binding file when all the pairs are verified. Whenever there is one pair fails to be verified, then the user’s binding file is rejected. Hence, the more pair is the MePKC digital certificate, the lower is the probability to successfully fake the user’s MePKC digital certificate, the harder is the organized crime group to be efficient, and the higher is the trust level of the user’s first group of MePKC digital certificate.

Coming to here, the Kaneyuki Kurokawa’s human interaction models are used to simulate the organized crime group to fake MePKC digital certificate. Organized crime group has at least three persons to conspire a crime. Figure 12.8 illustrates the group efficiency of committee meeting. Figure 12.9 illustrates the group efficiency of exploratory group. Figure 12.10 illustrates the success probability of technology transfer. The models in Figures 12.8-12.10 are all developed by Kurokawa and they are used in this article to derive Figures 12.11-12.13. Kurokawa’s model on committee meeting agrees with the coefficient of inefficiency of Parkinson’s Law ranging from 20 to 22 or more.

4500

Group Efficiency of Committee Meeting, GE_C = n * p^(n-1)
n = Network size of human group
p = Probability of the chemistry being good between the chairperson and a member

n \ p	0.500	0.700	0.850	0.900
0	0.000	0.000	0.000	0.000
1	1.000	1.000	1.000	1.000
2	1.000	1.400	1.700	1.800
3	0.750	1.470	2.168	2.430
4	0.500	1.372	2.457	2.916
5	0.313	1.201	2.610	3.281
6	0.188	1.008	2.662	3.543
7	0.109	0.824	2.640	3.720
8	0.063	0.659	2.565	3.826
9	0.035	0.519	2.452	3.874
10	0.020	0.404	2.316	3.874

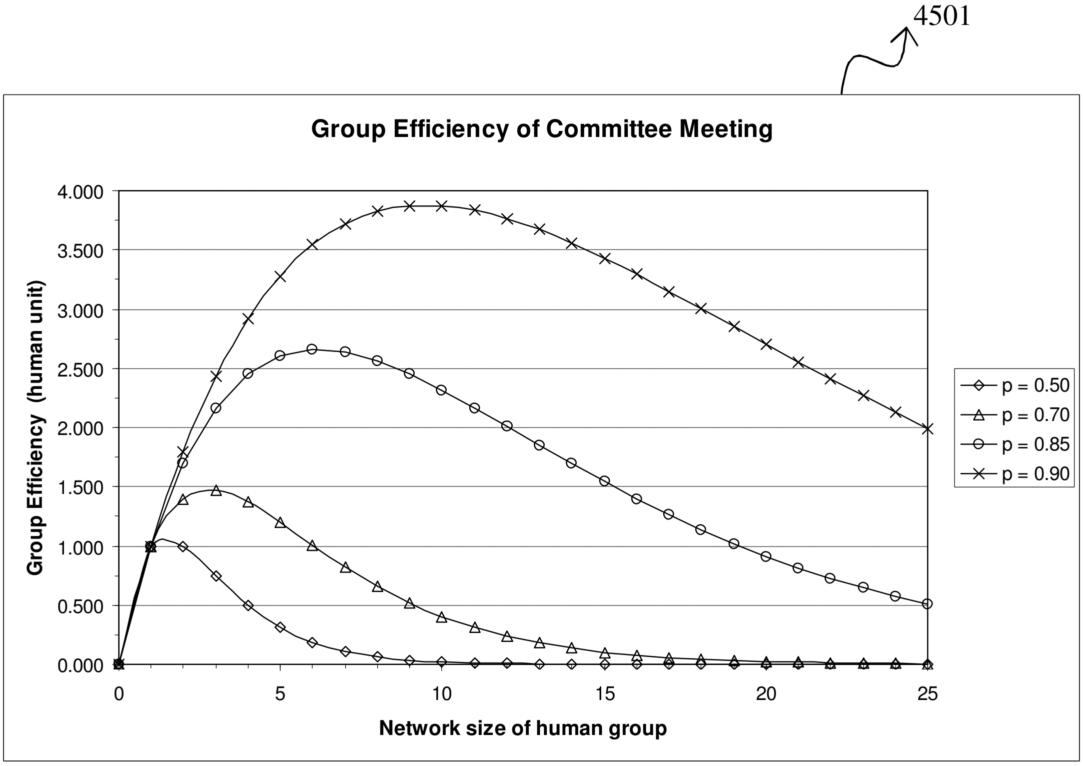


Figure 12.8 Group efficiency of a committee meeting according to the Kurokawa's human interaction model

4600

Group Efficiency of Exploratory Group, $GE_E = n * q^{(n*(n-1)/2)}$
n = Network size of human group
q = Probability of the chemistry being good between a pair of members

n \ q	0.500	0.700	0.850	0.900
0	0.000	0.000	0.000	0.000
1	1.000	1.000	1.000	1.000
2	1.000	1.400	1.700	1.800
3	0.375	1.029	1.842	2.187
4	0.063	0.471	1.509	2.126
5	0.005	0.141	0.984	1.743
6	0.000	0.028	0.524	1.235
7	0.000	0.004	0.231	0.766
8	0.000	0.000	0.084	0.419
9	0.000	0.000	0.026	0.203
10	0.000	0.000	0.007	0.087

4601

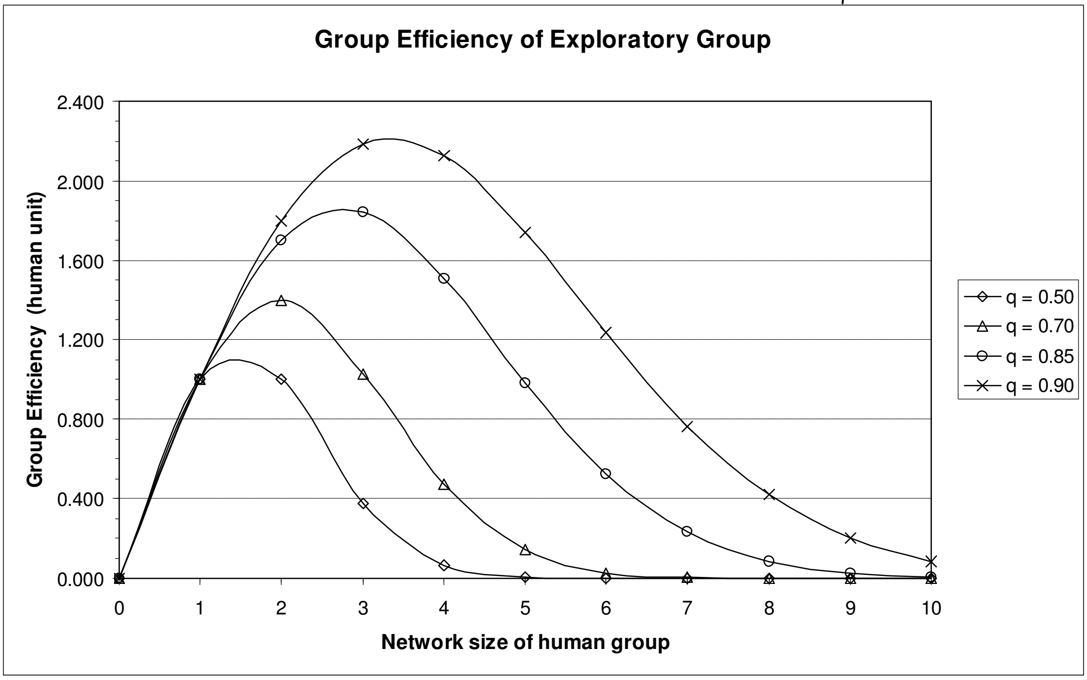


Figure 12.9 Group efficiency of an exploratory group according to the Kurokawa's human interaction model

4700

<table>
<tr><td colspan="5">Success Probability of Technology Transfer, SP_T = (p^(m-1+n)) * (q^n)
m = Number of ranks in the hierarchy, n = Number of receiving division, q = Probability of the chemistry being good between a pair of peer members, p = Probability of the chemistry being good between the chairperson and a member in a committee meeting</td></tr>
<tr><td>Curve</td><td>P</td><td>Q</td><td>R</td><td>S</td></tr>
<tr><td>n =</td><td>1</td><td>1</td><td>3</td><td>3</td></tr>
<tr><td>p =</td><td>0.750</td><td>0.850</td><td>0.750</td><td>0.850</td></tr>
<tr><td>m \ q</td><td>0.750</td><td>0.800</td><td>0.750</td><td>0.800</td></tr>
<tr><td>0</td><td></td><td></td><td></td><td></td></tr>
<tr><td>1</td><td>0.000</td><td>0.000</td><td></td><td></td></tr>
<tr><td>2</td><td>0.422</td><td>0.578</td><td></td><td></td></tr>
<tr><td>3</td><td>0.316</td><td>0.491</td><td>0.000</td><td>0.000</td></tr>
<tr><td>4</td><td>0.237</td><td>0.418</td><td>0.075</td><td>0.193</td></tr>
<tr><td>5</td><td>0.178</td><td>0.355</td><td>0.056</td><td>0.164</td></tr>
<tr><td>6</td><td>0.133</td><td>0.302</td><td>0.042</td><td>0.140</td></tr>
<tr><td>7</td><td>0.100</td><td>0.256</td><td>0.032</td><td>0.119</td></tr>
<tr><td>8</td><td>0.075</td><td>0.218</td><td>0.024</td><td>0.101</td></tr>
<tr><td>9</td><td>0.056</td><td>0.185</td><td>0.018</td><td>0.086</td></tr>
<tr><td>10</td><td>0.042</td><td>0.157</td><td>0.013</td><td>0.073</td></tr>
</table>

4700

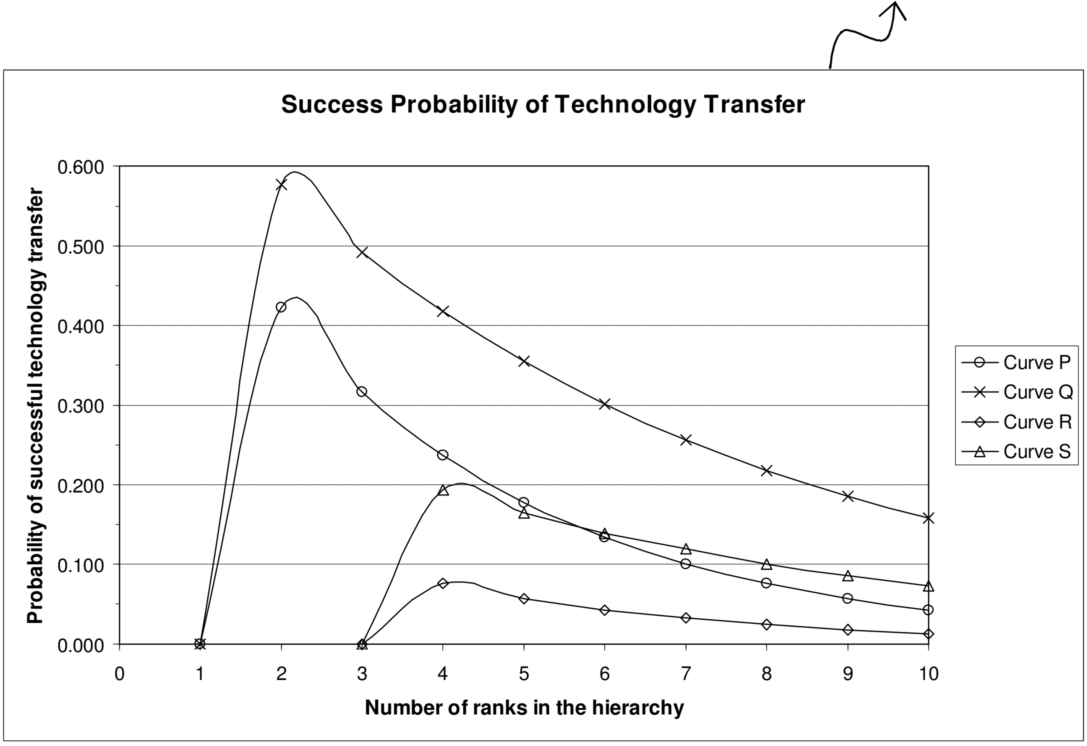


Figure 12.10 Success probability of technology transfer according to the Kurokawa's human interaction model

4800

Group Efficiency of Exploratory Group Formed from Leaders of Some Committee Meetings (without condition for common consensus), GE_{ECO}
for m = 0, GE = 0; for m = 1, GE = n*p^(n-1); for m > 1, GE = ((n*p^(n-1))*m) + (m*q^(m*(m-1)/2))
m = Network size of human group of exploratory leaders, n = Network size of every committee meeting, q = Probability of the chemistry being good between a pair of leader members, p = Probability of the chemistry being good between the chairperson and a member in a committee meeting

n =	1	2	3	4	5
p =	0.850	0.850	0.850	0.850	0.850
m \ q	0.800	0.800	0.800	0.800	0.800
0	0.000	0.000	0.000	0.000	0.000
1	1.000	1.700	2.168	2.457	2.610
2	3.600	5.000	5.935	6.513	6.820
3	4.536	6.636	8.039	8.906	9.366
4	5.049	7.849	9.719	10.875	11.489
5	5.537	9.037	11.374	12.819	13.587
6	6.211	10.411	13.216	14.950	15.871
7	7.065	11.965	15.237	17.260	18.335
8	8.015	13.615	17.355	19.667	20.896
9	9.003	15.303	19.510	22.111	23.493
10	10.000	17.000	21.675	24.565	26.101

4801

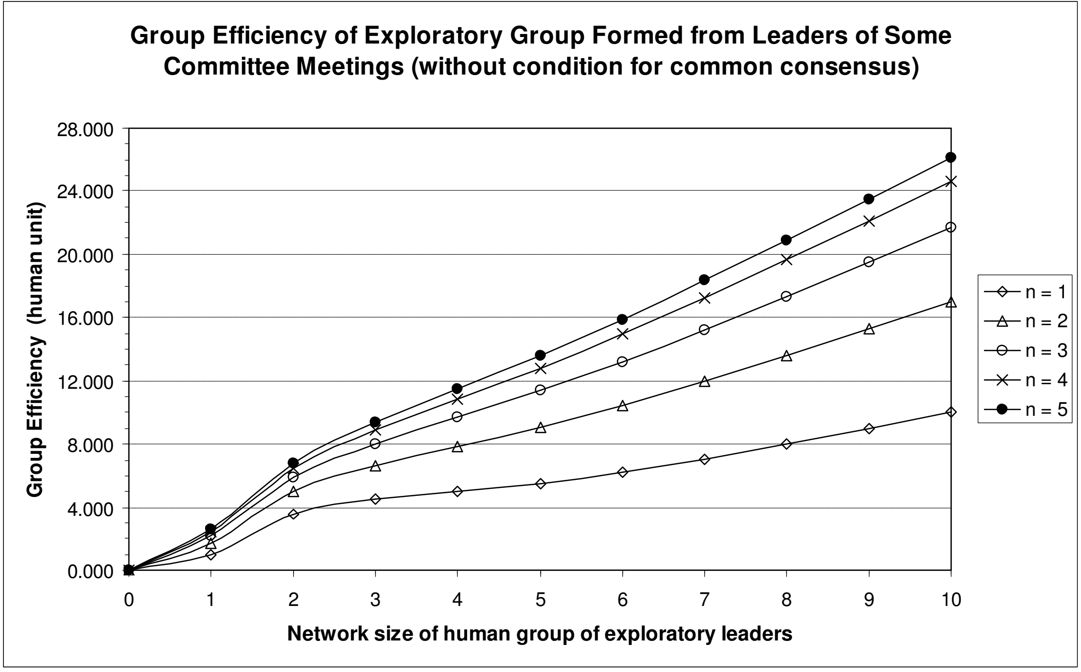


Figure 12.11 Group efficiency of an exploratory group formed from leaders of some committee meetings (without condition for common consensus) as modified and enhanced from the Kurokawa's human interaction models

4900

Group Efficiency of Exploratory Group Formed from Leaders of Some Committee Meetings (with condition for common consensus), GE_{ECW}
for m = 0, GE = 0; for m = 1, GE = (n*p^(n-1)) * (p^n);
for m > 1, GE = (((n * p^(n-1)) * m) + (m * q^(m*(m-1)/2))) * ((p*q)^m) * (p^((n-1)*m))
m = Network size of human group of exploratory leaders, n = Network size of every committee meeting, q = Probability of the chemistry being good between a pair of leader members, p = Probability of the chemistry being good between the chairperson & a member in a committee meeting					

n =	1	2	3	4	5
p =	0.850	0.850	0.850	0.850	0.850
m \ q	0.800	0.800	0.800	0.800	0.800
0	0.000	0.000	0.000	0.000	0.000
1	0.850	1.228	1.331	1.282	1.158
2	1.665	1.670	1.433	1.136	0.859
3	1.426	1.281	0.953	0.649	0.419
4	1.079	0.876	0.566	0.331	0.182
5	0.805	0.583	0.326	0.163	0.077
6	0.614	0.388	0.186	0.079	0.032
7	0.475	0.258	0.105	0.038	0.013
8	0.366	0.170	0.059	0.018	0.005
9	0.280	0.110	0.033	0.009	0.002
10	0.211	0.071	0.018	0.004	0.001

4901

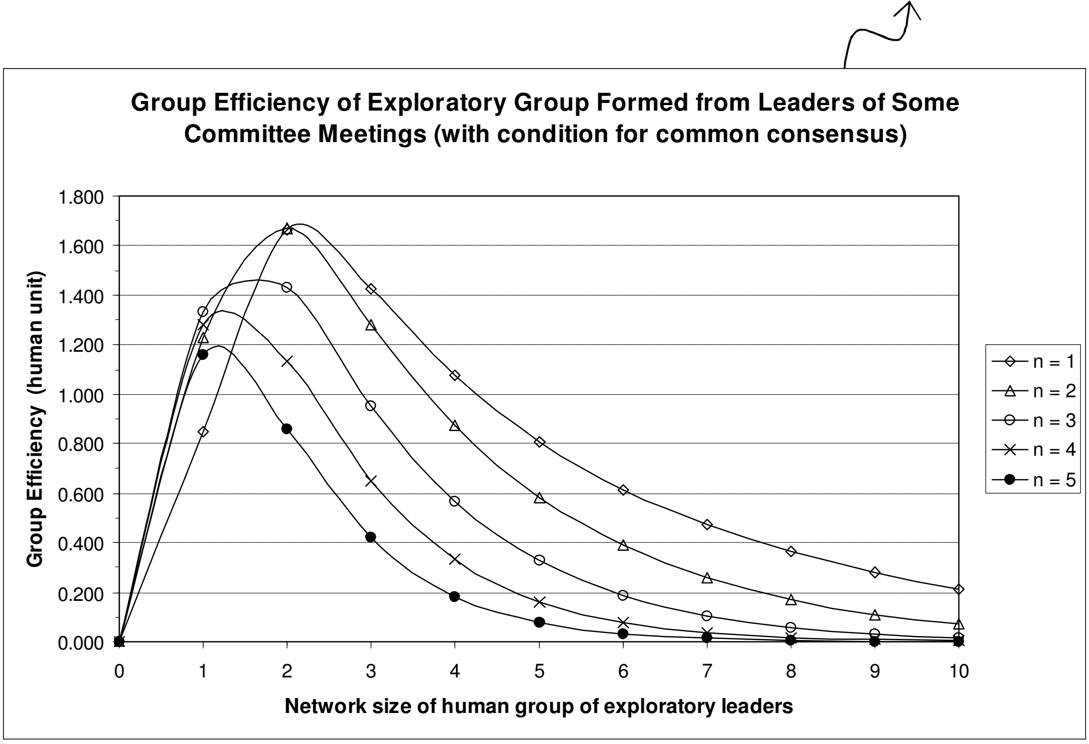


Figure 12.12 Group efficiency of an exploratory group formed from leaders of some committee meetings (with condition for common consensus) as modified and enhanced from the Kurokawa's human interaction models

5000

Success Probability of Exploratory Group Formed from Leaders of Some Committee Meetings (with condition for common consensus), SP_{ECW}
for m = 0, SP = 0; for m = 1, SP = p^n; for m > 1, SP = ((p*q)^m) * (p^((n-1)*m))
m = Network size of human group of exploratory leaders, n = Network size of every committee meeting, q = Probability of the chemistry being good between a pair of leader members, p = Probability of the chemistry being good between the chairperson & a member in a committee meeting

n =	1	2	3	4	5
p =	0.850	0.850	0.850	0.850	0.850
m \ q	0.800	0.800	0.800	0.800	0.800
0	0.000	0.000	0.000	0.000	0.000
1	0.850	0.723	0.614	0.522	0.444
2	0.462	0.334	0.241	0.174	0.126
3	0.314	0.193	0.119	0.073	0.045
4	0.214	0.112	0.058	0.030	0.016
5	0.145	0.065	0.029	0.013	0.006
6	0.099	0.037	0.014	0.005	0.002
7	0.067	0.022	0.007	0.002	0.001
8	0.046	0.012	0.003	0.001	0.000
9	0.031	0.007	0.002	0.000	0.000
10	0.021	0.004	0.001	0.000	0.000

5001

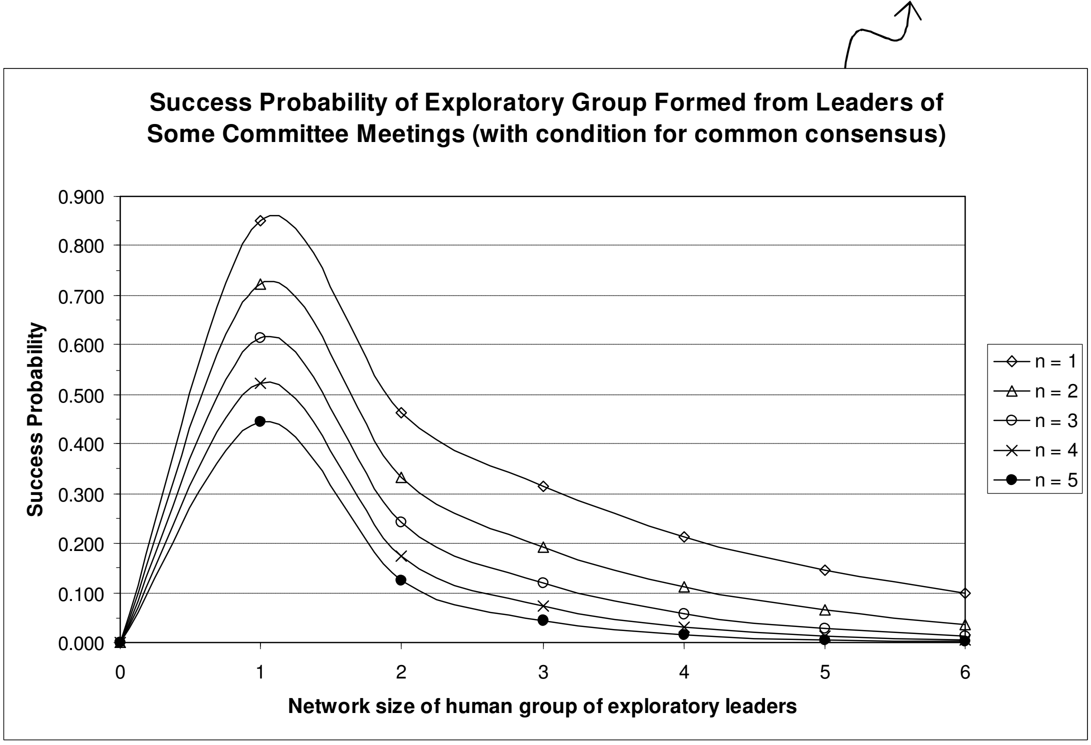


Figure 12.13 Success probability of an exploratory group formed from leaders of some committee meetings (with condition for common consensus)) as modified and enhanced from the Kurokawa's human interaction models

In other words, if an organized crime group similar to committee meeting has 20 to 22 persons or more, then it starts to be inefficient. If the organized crime group is similar to the exploratory group, then its inefficiency starts when the group has five or more members.

Nevertheless, for the personnel in the CA, the situation is similar to the committee meeting and getting 20 to 22 or more digital signatures from the CA personnel is not that practical. For the introducer of trust of web, the situation is similar to exploratory group. It is quite easy to get five of more digital signature to certify a user's binding file. However, the trust level of introducer is limited to how well the people know the introducer. It becomes quite impractical when other users are asked if they know all the five or more introducers certifying a user's binding file. Hence, other approach has to be implemented.

Up to here, the organized crime group, whether similar to committee meeting and/or exploratory group, becomes inefficient when the number of group members is more and hits a threshold. This is because criminals in an organized crime group are normally lacking of a high level of trust among themselves. They normally try their best to get rid of giving chances to other criminals to hold the evidence of their criminal activities. The more members in an organized crime group, the harder it is to be efficient. Furthermore, membership has to keep low to maintain a certain level of profit sharing as reflected by the Sayan Chatterjee's article (No date) "Does increased equity ownership lead to more strategically involved boards?".

A proof given to the Parkinson's Law is the time required to achieve a final agreement on the works to be done tends to be more when more people are involved and/or more time limit is given. This phenomenon is explained in articles Elliot Aronson and Eugene Gerard (1966), "Beyond Parkinson's Law: The Effect of Excess Time on Subsequent Performance"; Elliot Aronson and David Landy (1967), "Further Steps Beyond Parkinson's Law: A Replication and Extension of the Excess Time Effect"; as well as David Landy, Kathleen McCue, and Elliot Aronson (1969), "Beyond Parkinson's Law: III. The Effect of Protractive and Contractive Distractions on the Wasting of Time on Subsequent Tasks".

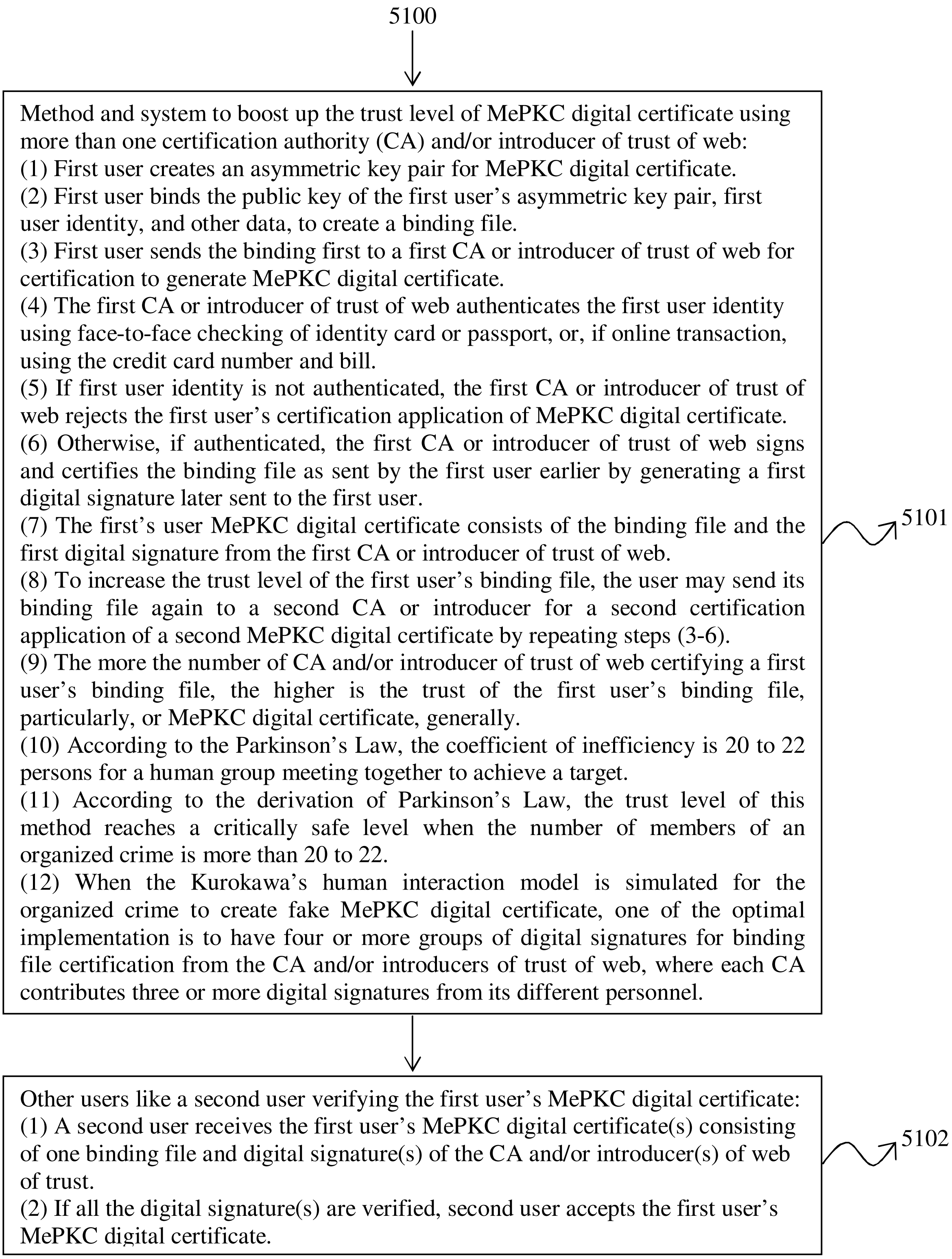
One more possible explanation is the longer time to achieve a common agreement as in the BGP (Byzantine Generals Problem) together with the capability to detect the faulty node. For organized crime group, all the members have to achieve a common agreement and detect those possible faulty members before any action is taken. As in the BGP for a very well-known fact, the larger is a network like the human group, the more messages or time are needed to achieve the common consensus. Therefore, to make the organized crime group to be inefficient, we have to design a PKI similar to the Kurokawa's human interaction models.

Figure 12.11 illustrates the group efficiency of exploratory group formed from leaders of some committee meetings without the condition for common consensus among the members. This is an intermediate step to tell that when common consensus among all the members is not needed, the group efficiency increases as the members of exploratory groups and committee meetings increase.

Figure 12.12 illustrates the group efficiency of exploratory group formed from leaders of some committee meetings with the condition for common consensus among all the members. Here, all the personnel in the CA represent a committee meeting, and each CA/introducer represents a member of the exploratory group. Since other users only accept a MePKC digital certificate when all the CA/introducer's digital signatures are verified, the organized crime group consisting of the malicious CA and/or introducer has lower efficiency as the network size increases. Figure 12.13 illustrates the success probability of exploratory group formed from leaders of some committee meetings with the condition for common consensus among all the members of the organized crime group.

12.3.3 Model to Boost up the Trust of MePKC Digital Certificate

It can be deduced that the more the criminals needed to succeed faking a MePKC digital certificate, the lower is the success probability. One of the optimal implementation is to have four (m = 4) or more groups of digital signatures for binding file certification from the CA and/or introducers of trust of web, where each CA contributes three (n = 3) or more digital signatures from its different personnel. In this case, the success probability of the organized crime group is less than 6%.



Method and system to boost up the trust level of MePKC digital certificate using more than one certification authority (CA) and/or introducer of trust of web:

(1) First user creates an asymmetric key pair for MePKC digital certificate.

(2) First user binds the public key of the first user's asymmetric key pair, first user identity, and other data, to create a binding file.

(3) First user sends the binding first to a first CA or introducer of trust of web for certification to generate MePKC digital certificate.

(4) The first CA or introducer of trust of web authenticates the first user identity using face-to-face checking of identity card or passport, or, if online transaction, using the credit card number and bill.

(5) If first user identity is not authenticated, the first CA or introducer of trust of web rejects the first user's certification application of MePKC digital certificate.

(6) Otherwise, if authenticated, the first CA or introducer of trust of web signs and certifies the binding file as sent by the first user earlier by generating a first digital signature later sent to the first user.

(7) The first's user MePKC digital certificate consists of the binding file and the first digital signature from the first CA or introducer of trust of web.

(8) To increase the trust level of the first user's binding file, the user may send its binding file again to a second CA or introducer for a second certification application of a second MePKC digital certificate by repeating steps (3-6).

(9) The more the number of CA and/or introducer of trust of web certifying a first user's binding file, the higher is the trust of the first user's binding file, particularly, or MePKC digital certificate, generally.

(10) According to the Parkinson's Law, the coefficient of inefficiency is 20 to 22 persons for a human group meeting together to achieve a target.

(11) According to the derivation of Parkinson's Law, the trust level of this method reaches a critically safe level when the number of members of an organized crime is more than 20 to 22.

(12) When the Kurokawa's human interaction model is simulated for the organized crime to create fake MePKC digital certificate, one of the optimal implementation is to have four or more groups of digital signatures for binding file certification from the CA and/or introducers of trust of web, where each CA contributes three or more digital signatures from its different personnel.

Other users like a second user verifying the first user's MePKC digital certificate:

(1) A second user receives the first user's MePKC digital certificate(s) consisting of one binding file and digital signature(s) of the CA and/or introducer(s) of web of trust.

(2) If all the digital signature(s) are verified, second user accepts the first user's MePKC digital certificate.

Figure 12.14 Method and system to boost up the trust level of MePKC digital certificate by using more than one certification authority (CA) and/or introducer of trust of web

Figure 12.14 illustrates the operations of the method and system to boost up the trust level of the MePKC digital certificate. Now, the first PKI risk informed by Carl Ellison and Bruce Schneier (2000) on “Who do we trust, and for what?” questioning on how well the CA maintains its private keys well and the third risk on “How secure is the verifying computer?” questioning on the possibility of attacker adding its own public key to the list of certificate verification, can also be improved by having more than one CA/introducer certifying a digital certificate. This is possible because users can generate their own asymmetric key pairs. The CA or introducer of trust of web may be a government authority, and people working in the fields of religion, law, police, security, politics, army, finance, diplomacy, etc., who have a high trust level in the society like judge, Commissioner for Oaths, lawyer, etc.

CHAPTER 13 MePKC TIMESTAMPING SCHEME FOR EVIDENCE OF INTELLECTUAL PROPERTY (IP) ORIGINALITY

13.1 Proof of Copyright Ownership Using Digital Timestamp in Malaysia

Copyright is an intellectual property needing no registration generally for an original work of authorship like literary, artistic, musical, or computer program affixed in any tangible medium of expression. It lasts for an author's life plus 50 years in Malaysia. Without registration, the author has to prove its time of creation using other methods. In Malaysia, proof of copyright and priority is generally established by way of a statutory declaration (SD). The cost is low and the processing fast. However, its disadvantages are low confidentiality and possible collusion between the author and Commissioner for Oaths. Here, digital timestamping under Malaysia's Digital Signature Act and Regulations (Act 562) is proposed to prove copyright ownership in electronic works (Lee, Radhakrishna & Khaw, 2007). Digital timestamping providing lower cost, faster processing, higher confidentiality, better security, and management, is a variant of digital signature.

13.1.1 Introduction

Intellectual properties (IPs) have become increasingly important in developed countries. Taplin (2004) reported that IP assets accounted for up to 70% and 40% of market values of all corporate assets in United States and Japan, respectively. Chandran (2007) stated that for any company, and especially for pharmaceutical companies, IP is more valuable than any of its tangible physical assets, where IP constituted more than 80% of the total revenues of any company while tangible assets accounted for only 20%. Again, Ocean Tomo LLC (Ocean Tomo, No date; Wikipedia Contributors, 2008d) figured out that the components of S&P 500 market value in 2005 also had 79.7% to be intangible assets.

Among the IP, patent has the largest economic value. In fact, IP is a group of rights consisting of copyright, patents, registered designs, trademarks, and know-how

(Curzon, 1998; Multimedia Development Corporation Sdn Bhd (MDC), 2002a, 2002c, 2002d, 2002e).

Certain IPs require official registration while others don't. Copyright requires no registration in Malaysia. This interdisciplinary paper looks at various methods for proof of copyright, and seeks to promote the idea of digital timestamping as a viable solution to evidentiary requirements. Both the legal and technical aspects of digital timestamping are discussed.

In the US, in order to claim statutory damages, a copyright has to be registered at the US Copyright Office. The statutory damages are pre-established damages for cases where a correct sum is deemed difficult to be calculated (Wikipedia Contributors, 2006). The basic level of statutory damages in US ranges from USD$750 to USD$30,000 per work at the discretion of court (World Intellectual Property Organization (WIPO), No date; Wikipedia Contributors, 2007e).

In Malaysia, a statutory declaration (SD) or affidavit is commonly used to prove the copyright authorship under the Copyright Act 1987 (Act 332) (MDC, 2002a). Copyright offender in Malaysia is subject to a fine not exceeding MYR$25,000 or to imprisonment not exceeding 3 years or to both under Copyright Act 1987 Section 43.

Digital timestamping is a cryptographic scheme being a variant of digital signature. In Malaysia, it is enforced under the Digital Signature Act 1997 and Regulations 1998 (Act 562) (MDC, 2002b). Currently, Malaysia has only 1 registered service provider of digital timestamping, i.e. MSC Trustgate.com Sdn. Bhd. (478231-X), a subsidiary of Multimedia Development Corporation Sdn. Bhd (MDeC, which is previously known as MDC) (Malaysian Communications and Multimedia Commission (MCMC), No date).

Haber and Stornetta (1990, 1991) firstly proposed a digital timestamping scheme with its described applications for scientific priority of patent inventions [9-10]. Here, we wish to extend its application to proof of copyright authorship and/or ownership in the Malaysian perspective. Cost estimation and comparisons with other methods are presented.

13.1.2 Related Works: Copyright

In Malaysia copyright is governed by the Copyright Act 1987, which extends protection for a qualified person to all types of literary, musical, artistic works, film, sound recording, broadcasts, derivative work, published edition, and live performances (Khaw, 2001).

Copyright subsists in original literary (including computer programs), musical, or artistic works, sound recordings, films, broadcasts or cable programmes, architectural plans, and the typographical arrangements of published editions of works (Curzon, 1998).

In Malaysia, the copyright owner is given the right to reproduce, prepare derivative works, distribute, communicate their works, perform, or rent their works for the duration of the author's lifetime plus 50 years after his death. In European Union, the protection period is 70 years after the death of the last author; whereas corporate authorship is 70 years from the year the work is created. In the US, the Sonny Bono Copyright Term Extension Act of 1998 extends the protection period of individual works to 70 years after the death of the last author; whereas corporate authorship is 120 years from the date of creation or 95 years after publication, whichever is shortest. Hence, the security design of timestamping scheme has to last for about 100 years in Malaysia.

It is transmissible by assignment or will as personal property. Copyright exists automatically once it is expressed in any tangible material form, e.g. if one doodles on a piece of paper while chatting on the phone, then that doodle is automatically protected without the need for any registration. However, if one were to compose a tune in one's head and hum it in the shower that is not protected as it has not been reduced to material form. Were it to be recorded then it would be protected.

The first owner of a copyright work is the author. Then, copyright may be assigned or transmitted. In cases where a work was created under a contract of employment or pursuant to a commission, the copyright in the work is transferred to the employer or commissioner, as the case may be. It must be noted that copyright is vested in the copyright owner and not the author, unless he is also the owner.

There is no registration system for copyright in Malaysia (Intellectual Property Corporation of Malaysia (Perbadanan Harta Intelek Malaysia) (MyIPO), No date). In the event of any copyright dispute, authorship and/or ownership would have to be proved by the party initiating the action. Internationally, the Berne Convention for the Protection of Literary and Artistic Works (WIPO, 1979; Wikipedia Contributors, 2007i) in 1887 set out the scope of copyright protection, applicable to all member countries. It was formed in order to formulate greater uniformity in copyright law among countries and to give copyright owners certain minimum levels of copyright protection without the requirements of registration in member countries (Articles 1 and 2 (WIPO, 1979)). The Berne Convention requires a signatory country to recognize a copyrighted work of authors from other signatory countries in the same way as it recognizes the copyright of its own national or domiciliary.

13.1.3 Related Works: Methods of Proving Copyright Ownership

Currently, there are six methods to prove the copyright ownership as follows:

(i) Self-addressed envelope with original work mailed back to oneself;

(ii) Log book;

(iii) Lodging a copy with solicitor or bank;

(iv) Hashing;

(v) Registration and/or preregistration at copyright office; and

(vi) Statutory declaration (SD).

For self-addressed envelope with original work mailed back to oneself, this is perhaps the oldest method being used to prove the scientific priority and copyright authorship or ownership. The original work is inserted into a self-addressed envelope, which is then sealed and mailed back to the author. The postal timestamp acts as a proof for the creation time. Whenever there is a case of copyright infringement, the sealed envelope is opened in public to settle the dispute. However this is known as the “poor man’s copyright” as there is scope for abuse. There is nothing to prevent tampering or pre-posting envelopes to oneself and inserting allegedly original works into the envelope later. As such they will not stand up as credible evidence in court (UK Copyright Service (UKCS), No date a).

For log book, an author may keep a habit like an inventor and scientist, where a log book is properly recorded and endorsed by one or more third parties regularly. In case of dispute, the log book and the endorsers can provide the proof of authorship.

For lodging a copy with solicitor or bank, the party, who endorses a log book, can be specifically a solicitor as in log book. Sometimes, a copy of the copyrighted works can be lodged with a solicitor and/or bank, who act as copyright witnesses. However, copyright witnessing is not their main concern and they are unlikely to understand the essential implications of the service (UKCS, No date a). Other weaknesses are data loss due to corruption and disaster as well as loss of evidence for future cases.

For hashing, hashes are cryptographic operations performed to transform one or more fields into a unique fixed bit stream as the contents of a piece of digital evidence (Oppliger & Rytz, 2003; Maurer, 2004) can be preserved. These mechanisms ensure that digital information “has not been altered.” Cryptographic hashes can be used to protect evidence from tampering for long periods of time. Using algorithms such as SHA-2 and RIPEMD-256, content integrity can be attested to for many years even under distributed brute force attack. Thus, by applying cryptographic hashes and digital signature technology to the problem of evidence preservation, the “who” and “what” of digital data can be identified and maintained. However this still leaves time to be established to claim scientific priority and copyright evidence (Duren & Hosmer, 2002).

For registration and/or preregistration at copyright office, in UK and US, there are copyright offices managing the copyright registration as strong evidence upon a fee payment. In UK, copyright registration service is offered by The UK Copyright Service (UKCS) (UKCS, No date b). For online registration, UKCS charges GBP$35 for 5 years or GBP$60 for 10 years per work, where it covers an upload up to 10MB per registered work and upload over 10MB are subject to an additional fee of 2 pence (= GBP$0.02) per additional 1MB or part thereof up to virtually unlimited size. For postal registration, UKCS charges GBP$40 for 5 years or GBP$70 for 10 years per work, plus different additional fees according to the formats of hard copies, up to a maximum size of 8.5 GB. Meanwhile, the US Copyright Office (USCO) offers registration and preregistration services. Preregistration is for unpublished work, where creation of work has begun aiming for commercial distribution, e.g. motion picture, musical work, sound recording, computer program, book, or advertising photograph. Preregistration is not a registration and its fee is USD$100. After a work is published, registration at USCO is required and the electronic filing of a basic copyright registration is USD$35 and paper application is USD$45 (USCO, No date).

For statutory declaration (SD), this is the most commonly used method of establishing copyright in Malaysia. By Malaysia Copyright Act 1987 Section 42, an affidavit or statutory declaration made before a Commissioner for Oaths (Curzon, 1998) or Notary Public and asserting that copyright subsists in a particular work and the person named therein is the copyright owner may be admitted as prima facie proof of the facts stated therein. This is a convenient and economical way of proving copyright and ownership. Section 42 places the burden on the infringer to dispute and challenge the prima facie evidence adduced by the copyright owner (International Intellectual Property Alliance (IIPA), 2001). Section 42 was introduced to facilitate proof of copyright and ownership in cases involving foreign copyright owners.

There are some problems for SD like unreliability of information in the SD itself, need to verify information in the SD itself, challenges to information in the SD, and SD proof for patent invention is hard to establish. In the US, the legal case of laser inventor Gordon Gould (Taylor, 2002; Wikipedia Contributors, 2007n) uses SD and it has lasted for about 30 years. Hence, it is inefficient for IP evidence, especially for patent.

However in practice, there have problems with the admission of Section 42 affidavits. Courts are cautious with respect to presumptions, and have required prosecutors to prove subsistence issues through other documents such as record company receipts of first publication, letters of authority, or sometimes even live testimony of right holder representatives (CLJ Legal Network Sdn. Bhd., 1998) (CLJ - Crime, Law, and Justice). Failure to comply with these requirements has in some cases led to acquittals or the rejection of the affidavits as evidence of ownership. The root of the problem has been that not many judges are well versed in Copyright law and defence counsels are quick to challenge any actions on technical grounds. Further weaknesses are its low confidentiality and possible collusion between the author/owner and Commissioner for Oaths.

13.1.4 Digital Timestamping Scheme to Prove Copyright Ownership

Ismail (2001) discussed some timestamping schemes for Malaysian applications. Among them, linking scheme (Haber & Stornetta, 1990, 1991) and tree scheme (Benaloh & de Mare, 1991) limit the power of timestamping authority (TSA). Here, a modified scheme is proposed to limit the TSA power but has simpler implementation as in Figures 13.1-13.3. The TSA cannot collude with a user to backdate a timestamp more than one day using publication of daily superhash at a repository authority like newspaper.

13.1.5 Malaysia Digital Signature Act 1997 and Regulations 1998 (Act 562)

A "digital signature" is a transformation of a message using an asymmetric cryptosystem such that a person having the initial message and the signer's public key can accurately determine (Section 2 (MDC, 2002b)):

(i) whether the transformation was created using the private key that corresponds to the signer's public key; and

(ii) whether the message has been altered since the transformation was made.

Thus it is able to identify the signer and also pick up evidence of any tampering. Section 64 deems a digitally signed document to be a written document as if it had been written on paper if:

(i) it bears in its entirety a digital signature; and

(ii) that digital signature is verified by the public key listed in a certificate which:

  (1) was issued by a licensed certification authority; and

  (2) was valid at the time the digital signature was created.

0.0 Initialization vector, IV = a random bitstream for the first day or yesterday superhash for second day and beyond. Let i = 1, 2, 3, …, N, where N is the number of total timestamp users for one day, and j = 1, 2, 3, …, ∞.

1.0 For j-th day with N users.

1.1 User U_i creates hash H_{1i} of a digital document D.

1.2 User sends H_{1i} to timestamping authority (TSA).

1.3 TSA creates hash H_{2i} from the concatenation C_{1i} of H_{1i} and a timestamp T_i having the present time synchronized to an Internet time server like “time.windows.com” or “time.nist.gov”.

1.4 TSA signs the H_{2i} using its private key K_{pte} and creates hash H_{3i} as signed timestamp.

1.5 TSA stores the concatenation C_{2i} of H_{3i} and T_i.

1.6 TSA sends C_{2i} to U_i and repository authority R.

1.7 User U_i verifies the signed timestamp H_{3i} using T_i and TSA’s public key K_{pub} as in Fig. 2.

1.8 If H_{3i} is verified, user U_i accepts the H_{3i}; otherwise, U_i rejects the H_{3i} and goes to step 1.2.

2.0 TSA creates a daily pre-superhash PH_j from the concatenation of ordered C_{2i} on j-th day.

$PH_j = hash(C_{21} \| C_{22} \| C_{23} \| \ldots \| C_{2i} \| \ldots \| C_{2N})$

3.0 TSA creates daily superhash SH_j from the concatenation of IV and PH_j on j-th day.

3.1 If j = 1, IV = a random bitstream; otherwise if $j \geq 2$, IV = SH_{j-1}.

3.2 $SH_j = hash(IV \| PH_j)$

3.3 TSA stores the superhash SH_j.

4.0 TSA sends the SH_j to repository authority R for deposit and publication to limit the TSA power.

5.0 Go to step 1.0 for a following day.

Figure 13.1 Generation of timestamp and superhash

0.0 Digital or electronic document D, concatenation C_{2i} of signed timestamp H_{3i} and timestamp T_i.

1.0 User creates hash H_{v1i} of a digital document.

2.0 User creates hash H_{v2i} from the concatenation C_{v1i} of H_{v1i} and T_i.

3.0 User applies TSA's public key K_{pub} on signed timestamp H_{3i} to create decrypted signature H_{v3i}.

4.0 If $H_{v2i} = H_{v3i}$, the signed timestamp H_{3i} is verified and the user accepts it; otherwise if $H_{v2i} \neq H_{v3i}$, signed timestamp H_{3i} is rejected.

Figure 13.2 Verification of a signed timestamp

0.0 First day IV_1 or yesterday superhash SH_{j-1}, daily superhash SH_j, concatenations C_{2i} of signed timestamps H_{3i} and timestamps T_i on j-th day from repository R. To-be-verified concatenation C_{v2i} supplied by a user U_i.

1.0 User U_k checks the presence of C_{v2i} among the C_{2i} in the R's record.

2.0 If C_{v2i} equals to one of the C_{2i}, then go to step 3.0; otherwise, user U_k rejects the C_{v2i} and ends.

3.0 User U_k creates a daily pre-superhash PH_j from the concatenation of ordered C_{2i} on j-th day.

$PH_j = hash(C_{21} \parallel C_{22} \parallel C_{23} \parallel \ldots \parallel C_{v2i} \parallel \ldots \parallel C_{2N})$

4.0 User U_k creates a to-be-verified daily superhash SH_{vj} from the concatenation of IV and PH_j.

4.1 If $j = 1$, $IV = IV_1$; otherwise if $j \geq 2$, $IV = SH_{j-1}$.

4.2 $SH_{vj} = hash(IV \parallel PH_j)$

5.0 If $SH_{vj} = SH_j$, then user U_k verifies the C_{v2i}; otherwise, user U_k rejects the C_{v2i}.

Figure 13.3 Verification of daily superhash to limit TSA power

Further under Section 66, a certificate issued by a licensed certification shall be an acknowledgement of a digital signature verified by reference to the public key listed in the certificate, regardless of whether words of an express acknowledgement appear with the digital signature and regardless of whether the signer physically appeared before the licensed certification authority when the digital signature was created, if that digital signature is:

(i) verifiable by that certificate; and

(ii) affixed when that certificate was valid.

Section 67 presumes the validity and truth of the contents of the certificate issued by a recognized Certification Authority or Repository as to the digital signature being that of the subscriber. Apart from digital signatures, Section 70 also recognizes digital timestamps. A "time-stamp" means (Section 2 (MDC, 2002b)):

(i) to append or attach to a message, digital signature or certificate a digitally signed notation indicating at least the date, time and identity of the person appending or attaching the notation.

Further Section 62 of the DSA Regulations provides that a recognised date/time stamp service shall:

(i) on receipt of a document for time-stamping, immediately time-stamp the date and time of its receipt on the document and digitally sign the time-stamp; and

(ii) at the end of each business day cause to be published only the hash result of the document in at least one recognised repository all documents time-stamped by it in that day.

Thus a digital timestamp is able to provide evidence of "who", "what", and "when" accurately to establish priority of copyright. It helps the author to prove when the idea was expressed – this is the moment when copyright protection begins and is especially effective for protecting prepublication work. The third-party timestamp and certificate provides evidence to help resolve or avoid legal disputes without the need for costly and time-consuming litigation.

13.1.6 Malaysia Evidence Act 1950 (Act 56)

Amendments to the Evidence Act 1950 (Act 56) allowed the admissibility of "computer-generated documents" (International Law Book Services (ILBS), 2003).

Under Section 3, a matter recorded, stored, processed, retrieved, or produced by a computer is a “document”. Section 62 explains that a document produced by a computer is primary evidence. Section 78A accepts public documents produced by computers. Section 90A admits in evidence documents produced by computers subject to certain conditions. Timestamp produced by computers is a document that can act as primary and prima facie evidence.

13.1.7 Security Analysis

The security of the proposed digital timestamping scheme depends on the lowest security strength among the hash algorithms and digital signature scheme on timestamp. Hash algorithms like SHA-256, SHA-384, and SHA-512, have no key and the proposed bits of security range from 128 to 256 bits with protection periods at 30 years to foreseeable future, respectively, based on the symmetric key scale.

For digital signature scheme of timestamp, it needs a key pair to sign and verify a digital signature. Due to human memory limits, memorizable private key is currently not practical until the invented big secret creation methods are proposed in this research project. Hence, encrypted private key or two-factor authentication using a symmetric key and a token is normally used. In other words, the weakest point of digital timestamping scheme is at the digital signature protocol, which depends on the safety of symmetric key encrypting a private key and the availability of token.

The security of the proposed timestamping scheme in Figures 13.1-13.3 is similar to the linking (Haber & Stornetta, 1990, 1991) and tree schemes (Benaloh & de Mare, 1991) because only the superhash generation is different. The daily superhash in the proposed scheme here is generated from the concatenation of yesterday superhash and the concatenation of all the timestamps at the present day.

These schemes rely on the security of hash function and digital signature for the individual timestamp. For the security against the malicious TSA, it relies on the daily superhash safely kept at an independent repository authority. As long as there is no collusion between the TSA and repository, malicious TSA can be detected. If there exists collusion between TSA and repository, public can detect this event by keeping a published daily superhash for future verification. Another alternative is to increase the number of repository. Table 13.1 compares the proposed scheme with the linking and tree schemes.

Table 13.1 Comparisons of our scheme with linking and tree timestamping schemes

Feature	Linking Scheme	Tree Scheme	Proposed Scheme
Signature Scheme	Same. Equal security strength.		
Single Timestamp	Same. Equal security strength.		
Number of Computer Units	One	Many	One
Generation Speed of Superhash	Slow	Fast	Moderate
Malicious TSA	Same. Equal security strength.		

13.1.8 Computation Load

The computation load is heavy during the signing process and light during the hashing process. There is only one signing process for one timestamp. Meanwhile, there are many hashing processes for timestamp and superhash. Hence, the computation load is efficient and similar to the linking scheme and tree scheme (Ismail, 2001). The signing process is also fast enough because it is signing a hash value rather than a whole digital file.

13.1.9 Implementation Costs

The implementation cost of this digital timestamping scheme is estimated based on the one-time and annual operating costs as in Table 13.2.

The USCO receives 600,000 copyrighted works annually from a 300 million population. Malaysia with a population of 27 million is estimated to have 54,000 copyrighted works annually by assuming the US and Malaysia have same level of creativity. Let one timestamp to be charged at MYR$5 for 5-year repository period, then a total of MYR$270,000 can be obtained from the sale of timestamps. One can breakeven within the first year. To absorb the hidden cost and inflation as well as to maintain a fixed price per timestamp for about 5 to 10 years, a fee of MYR$10 per timestamp shall be charged for a repository period of 30 years.

Table 13.2 Cost estimation table of a digital timestamping scheme in Malaysia on 8 June 07

Item	Unit	Cost (RM)
Repository Licence - establishment stage - operation stage - granting fee - annual operating fee	1	2,500 2,500 30,000 2,500
Digital timestamp Licence - establishment stage - operation stage - granting fee - annual operating fee	1	2,500 2,500 30,000 2,500
Computer	4	10,000
Windows Vista Business Edition	4	5,080
Norton AntiVirus 2007 (1 year)	4	558
ZoneAlarm Internet Security Suite (3 PCs) (1 year)	2	349
Spybot – Search 7 Destroy 1.4	4	2,000
InstallShield 12 Express (Windows) + Visual Studio .NET Professional 2005	2	7,661
Corporate Web Hosting (1 year)	1	1,000
Furniture	1	6,000
Annual Company Secretary Fee	1	3,000
Annual Accounting Fee	1	5,000
Annual Auditing Fee	1	5,000
Annual Office Rental	1	24,000
Annual Advertisement Fee	1	12,000
Annual Transportation Fee	1	3,000
Annual Legal Fee	1	6,000
Annual Engineer Salary	2	72,000
Annual CreditCard Commission (VISA & MasterCard @ 2.35%)	54k	1,269
Total (one-time cost + annual fee, where n is year)		100,741 + 138,176n

N.B. USD$1.00 = MYR$3.4850 (RM) (The Star Online, 2007)

A regular renewal can be made. Otherwise, longer protection periods need higher fees such as MYR$20 per timestamp for 100 years subject to the TSA sustainability.

13.1.10 Comparisons

Table 13.3 compares the proposed method with other methods to prove copyright authorship/ownership. In Malaysia, an SD is charged at MYR$4 for the first copy and its following copies. The duplication-free and travelling-free factors make timestamp cheaper. Other advantages of copyright evidence using digital timestamp are independent evidence, contractual contingency, confidentiality, fast processing, limited authority power, low cost, and better management.

Table 13.3 Comparisons of methods to prove copyright authorship/ownership

Features	Self-Addressed Posting	Log Book	Lodging a Copy with Bank, etc.	Hashing	Registration	Statutory Declaration	Digital Timestamping
Independent Evidence	N	N	Y	Y	Y	Y	Y
Contractual Contingency	N	N	N	N	Y	N	Y
Confidentiality	N	N	N	Y	Y	N	Y
Duplication-Free Cost	N	N	N	Y	N	N	Y
Travelling-Free Cost	N	N	N	Y	Y	N	Y
Fast Processing	N	N	N	Y	N	Y	Y
Limited Authority Power	N	N	N	N	Y	N	Y
Low Cost	Y	Y	N	Y	N	Y	Y
Better Management	N	N	N	Y	Y	N	Y
Low Complexity	Y	Y	Y	N	N	Y	N

N.B.: N = No, Y = Yes.

13.1.11 Conclusion

From both the legal and technical perspectives, digital timestamping offers comparatively better proof, security, efficiency, and economy than the other methods discussed, especially SD. This section mainly proposes a new alternative to prove copyright ownership from the legal aspect using a cryptographic timestamping scheme. The proposed timestamping scheme is slightly modified from the previous schemes.

13.2 IP Evidence Using Digital Timestamp

13.2.1 Introduction

Among the IPs, patent and copyright are the sources of income. Gregory Aharonian (No date) listed out the incomes due to patent and copyright in the website of STO (Source Translation and Optimization). Gordon Gould (Wikipedia Contributors, 2007n), who is the inventor of laser, has a thirty-year patent war of lawsuit to claim his invention for patent (Taylor, 2002). The evidence to prove the creation time is critical for his patent filing because the US follows the first-to-invent system. Sarcastically, Gould is not awarded the Nobel Prize in Physics for his invention of laser but there are many other Nobel laureates based on laser.

In a book excerpt by Nick Taylor (2002):

"In 1957 Gordon Gould, then an obscure physicist and perennial graduate student, conceived one of the revolutionary inventions of the twentieth century -- the laser. But before he could submit a patent application, a prominent professor of physics whose office was next door to Gould's filed his own laser patent claims. Gould fought to reclaim the rights to his work, beginning a battle that would last nearly thirty years."

Here, the notary system has weakness. Otherwise, the lawsuit would not have been prolonged to 30 years. The disadvantages are low confidentiality and possible collusion between the inventor/author and Commissioner for Oaths. In the digital era, it is also hard to prove the creation time of a copyrighted work. Need is required to differentiate between originality and plagiarism. Consequently, digital timestamp service is proposed here to provide IP evidence in Malaysia and other geo-political regions, particularly patent and copyright.

The suitability of timestamp evidence for patent filing system is investigated. There are two patent filing systems: First-to-invent and first-to-file. These patent filing systems have the mixed advantages and disadvantages like first inventor in the interference case, possible idea accumulation towards commercial value, reasonable time for prototype development, notary service, patent translation costs, patent filing costs, and intentional delay of patent filing.

Based on the good and bad points, a hybrid system may be introduced, which is called "limited first-to-invent patent filing system", where the period between the first date of invention claim and first filing date is within a reasonable figure like one to two years. The legislation of the related laws in the fields of cyberlaw and intellectual property (IP) law is needed to enforce the IP evidence using digital timestamp as proposed here.

13.2.2 Related Works: Patent Filing Systems

In this world, there are two patent filing systems: First to invent, and first to file. By the last decade, there are only three countries in the world implementing the first-to-invent patent filing system (Inventors Assistance League, No date; Coster, 2002; Wikipedia Contributors, 2007j); whereas the other countries are all implementers of first-to-file patent filing system. Till today, the USA is the only country implementing the first-to-invent patent filing system.

Nevertheless, now even the USA is on the brink of patent laws reformation since the last decade (Samuelson, 2004; Mossinghoff, 2005; Wikipedia Contributors, 2008b, 2008e) for the issues like patent filing system, number of core and edge invention claims in a patent, unity of invention (Wikipedia Contributors, 2008ba), and inventorship (Wikipedia Contributors, 2008ao).

First to invent means the first inventor has the right to file for patent first upon proving the first date of invention claim whenever there is an inventor interference case. First to file means the inventor, who firstly files for the patent application throughout the world, has the right to claim as the inventor and the filing date is the invention priority date.

13.2.3 Related Works: Digital Timestamping Scheme

The digital timestamping scheme proposed in Section 13.1 to prove copyright authorship/ownership can be extended to other IPs like patent here. Please refer to Sections 13.1.9 and 13.1.10 for the implementation costs, as well as comparisons for advantages and disadvantages, respectively.

For clarity, the added advantages includes confidentiality, infinite complimentary duplication of timestamp, savings on transportation and operating costs, lower overall cost, faster processing, better management, and limited authority power of timestamping service provider. The timestamping authority (TSA) has limited power that it cannot backdate because of the daily superhash stored regularly in a trusted repository everyday. In case the TSA is malicious, the accuracy of the timestamp can be trusted at the unit of day. To increase the fault tolerance to malicious repository, the number of repositories can be increased and resorts to the Byzantine Agreement Protocol (BAP) of the Byzantine Generals Problem (BGP).

13.2.4 Proposing Limited First-to-Invent Patent Filing System

IP evidence with digital timestamp is a timely proposal in view of the inadequacies of existing methods to prove IP creation time. Additionally, the advent of computing technology has paved the way for more reliable and efficient means to digitally protect information (IP inclusive).

At the same time, the proposed service will prevent patent lawsuits similar to Gould's from recurring, which is based on first-to-invent system in the USA rather than first-to-file system in the rest of the world.

Considering the advantages and disadvantages of these systems (Inventors Assistance League, No date; Coster, 2002; Mossinghoff, 2005), like first inventor in the interference case, possible idea accumulation towards commercial value, reasonable time for prototype development, notary service, patent translation costs, patent filing costs, and intentional delay of patent filing, a hybrid system shall be introduced, which is a limited first-to-invent patent filing system, where the period between the first date of invention claim and first filing date is within a reasonable figure like one to two years.

In a book excerpt by Burgess and Power (2008):

"The U.S. Chamber of Commerce estimates that counterfeit and pirated products account for 5 percent to 7 percent of the global economy, and results in the loss of more than 750,000 jobs and approximately $250 billion in sales to the United States alone. … …

The threats of economic espionage and intellectual property (IP) theft are global, stealthy, insidious, and increasingly common. According to the U.S. Commerce Department, IP theft is estimated to top $250 billion annually and also costs the United States approximately 750,000 jobs. The International Chamber of Commerce puts the global fiscal loss at more than $600 billion a year."

Limited first-to-invent patent filing system has all the features of first-to-invent patent filing system except that the first date of invention claim and first filing date is limited to be within a reasonable figure like one to two years, depending on the further research studies for better optimization. The main aim of this limited first-to-invent system is for data protection (Haskin, 2008) to resist the prevalent hacking activities (McClure, Scambray & Kurtz, 2001; Beaver, 2004; Pankaj, 2005; Berinato, 2007a, 2007b, 2007c, 2007d, 2007e, 2007f; Wikipedia Contributors, 2007q, 2008n; McMillan, 2008b) and economic espionage (Fialka, 1999; Fink, 2002, 2003; McNamara, 2003; Nasheri, 2004; Burgess & Power, 2008). Hacking and economic espionage are the main killing weaknesses of the first-to-file patent filing system.

Beaver (2004) reported that a networked computer without proper firewall (Ogletree, 2000) settings would be hacked within 30 minutes. Yet in the latest news, Markoff (2008) informed that the hacking period dropped to less than 5 minutes after a hacker had operated for 30 seconds to access a prey computer. This reflects how serious and dangerous the current computer communications network security (Stallings, 2000) is in this networked info-computer era.

Identity theft can happen when a hacker copies a prey's computer data as disk image (Wikipedia Contributors, 2008bf) using disk cloning software (Wikipedia Contributors, 2008be), and then put the disk image into a second computer and modify, add, delete, etc. on some contents, which creates a second type of zombie computer (Wikipedia Contributors, 2008bj) and/or botnet (Wikipedia Contributors, 2008bd, 2008bi). This second type of zombie computer, when connected to the Internet, can fool other prey hackers watching this zombie computer version 2. Of course, if there are any confidential information, business secret, and other intangible assets, in the prey computers, then they shall be considered as disclosed and released to the public domain.

Tables 13.4-13.6 show the advantages and disadvantages of first-to-file, first-to-invent, and limited first-to-invent patent filing systems, respectively. From these tables, it can be concluded that limited first-to-invent patent filing system shall be highly appreciated and MePKC digital timestamp shall be used as the IP evidence.

Table 13.4 Advantages and disadvantages of first-to-file patent filing system

Advantages	[1] No interference case to determine the first author [2] Eliminate intentional delay of patent filing
Disadvantages	[1] No chance for idea accumulation towards commercial value [2] Less number of claims per patent application [3] Very high patent filing cost [4] Risk for hacker stealing the electronic file that can be easily copied in this networked computer world

Table 13.5 Advantages and disadvantages of first-to-invent patent filing system

Advantages	[1] Possible for idea accumulation towards commercial value [2] Has time for prototype development [3] More claims per patent application [4] Lower patent filing cost
Disadvantages	[1] Public notary service used to prove the first inventor is not efficient and practical by referring to the exemplary case of the laser inventor of Gordon Gould which has lasted for 30 years [2] Interference case to determine the first inventor is hard [3] Intentional delay of patent filing for public disclosure

Table 13.6 Advantages & disadvantages of limited first-to-invent patent filing system

Advantages	[1] Possible for idea accumulation towards commercial value [2] Has time for prototype development [3] More claims per patent application [4] Lower patent filing cost [5] Digital timestamp is more efficient and practical than public notary service. [6] Interference case to determine the first inventor is easier [7] Reasonable delay of patent filing for public disclosure
Disadvantages	[1] Need survey and research to determine the window period between the first date of invention claim and first patent filing date, which is suggested to be from 1 to 2 years. [2] Need international revision of patent law, especially PCT under the WIPO that may take a long time

CHAPTER 14 HACK-PROOF DATA STORAGE USING INNOVATED DIP SWITCH

14.1 Abstract

A dual in-line package (DIL/DIP) switch has been modified to collectively link all the poles using a single actuator and called secure DIP switch. The actuator can be a raised/recessed slide, raised/recessed rocker or piano-type (aka side/level), selectively switching on or off one/two groups of poles oppositely. A specific inventive application is when a 10/12-way secure DIP switch is integrated with two modular jack RJ45 sockets and a second storage device preferably via USB connection, a secure data storage resisting the computer hacking in a malicious computer network is created. This new component is simple, cost-effective, and hack-proof. Yet a novel variant is N_1PST+N_2PST DIP switch with reverse activation.

14.2 Introduction

Hacking or cracking into a computer from a malicious computer network (Wikipedia Contributors, 2007q, 2008n) is a great threat to the information security of private and confidential data in this electronic society. History of hacking and cracking can be traced (Wikipedia Contributors, 2008n). To resist the hacking and cracking, network settings and firewall software (Ogletree, 2000) are among the available best tools. However, these tools are complicated and not user-friendly to a networking novice like common Internet user. They are only good to network administrator who has undergone training and/or understood the operating manual.

In other words, network settings and firewall software are excellent at the server side but not the client side. Technical difficulty and affordable cost are two main factors discouraging the users to adopt these two anti-hacking approaches effectively. Furthermore, end users normally do not require data sharing via web hosting like server. This indicates that private and confidential data of end users can actually be partitioned from the data without security concern. For more information on the imperative demand of hack-proof data storage, please refer to Section 13.2.4.

In addition to the financial loss of confidential information and business secret, there are cronies of organized crime using the hacked secrets, flash mob approach, and sound snatching to conspire for more serious crimes like to worsen a good human relationship and/or to fasten a cheating human interaction. Married couples may be made divorced. Lovers may be made suspicious between themselves. Relatives, friends, colleagues, and organization members may be made trust-less and negatively emotional. Cheaters may succeed to establish trust, cultivate positively false emotion, and build a dishonest relationship leading to a marriage for sharing or even controlling the power, wealth, reputation, and fame of a single man or woman with good social status. In short, the criminals may cheat for sex, trust, emotion, power, money, and assets.

Here, method and device are proposed to secure a hacking-free (or hack-proof) data storage for end users. This method uses a new component called secure DIP switch integrated with two modular jack sockets and a second storage device like hard disk drive (HDD) or USB (Universal Serial Bus) flash drive. Private and confidential data is stored in the second storage device. Secure DIP switch controls the normal networking mode while it is switched into one direction and hacking-free mode while it is switched into the opposite direction. This method is simple, cost-effective, and hack-proof. End users can use this method to have hacking-free working environment without risking the firewall.

14.3 Related Prior Arts

Besides complicated networking settings and firewall software, a simple hardware device was proposed by Fonseca (2003) by using a simple push/pull level of a switch box to connect or disconnect the networking connection for the hacking elimination. Fonseca called it as data line switch and filed for patent in the US on 24 July 2001. Later, Macuch (2005) designed the data line switch for the applications of coaxial and DSL cables to control the computer connection to the Internet. Macuch filed for a design patent in the USA on 17 November 2003. Of course, there is yet another current practice by some end users to plug and unplug the networking cable.

Nevertheless, this method suffers from the hook damage of RJ45/RJ11 and inconvenience access of networking port.

Here, a proposed component with similar function to data line switch is also applicable to modular jack (aka modular connector) (Wikipedia Contributors, 2007m) like RJ45 and RJ11. Modular connector was firstly invented by Hardesty (1975), who filed it for patent in the US on 6 July 1973. RJ stands for registered jack (Wikipedia Contributors, 2007l). RJ45 and RJ11 are used as Ethernet jack and telephone jack, respectively. Our new component is innovated from the dual in-line package (DIL/DIP) switch by adding a collective actuator, which can be slide, rocker, or piano-type (aka side/level).

The miniature DIP switch was found in the US patent database to be firstly invented by Lockard (1977, 1979), who filed for patent in the US on 25 March 1975 for the first time. Since then, there are various innovations on the DIP switch. Hoffman (1982) had improved the manufacturing of DIP switch. Liataud and Maloney (1983) had reduced the size, decreased the cost, and increased the reliability of DIP switch. Brown (1983) had created the piano-type DIP switch. In the late decade, Lin (1999) and Tai (2001) from Taiwan, R.O.C., had concomitantly decreased the size, improved the manufacturing process, and increased the reliability of DIP switch.

Normal slide switch wipes in parallel with the pin pairs. The slide actuator of our proposed component wipes transversely to the pin pairs. The first slide switch that can be found in the US patent database was invented by Bailey (1969). Even though the wiping directions of normal slide switch and our switch are different by 90^{o} degrees, their function is the same, i.e. to connect and disconnect the poles, except the 10-way secure DIP switch oppositely switches two groups of poles.

14.4 Proposed Secure DIP Switch

For conventional *n*-way nPST (n Poles Single Throw) DIP switch, all the *n* poles are independently switched on or off in parallel with the pin pairs **101** and **102**. A simple structural diagram of a 10-way DIP switch is shown in Figure 14.1. Here, a modified DIP switch called *secure DIP switch* is innovatively proposed, where all the individual switches of the DIP switch are joined and controlled simultaneously by a transverse slider acting as an actuator in Figure 14.2. Alternative actuators are raised/recessed slide, raised/recessed rocker, and piano-type (aka side/level). When a USB connection is considered, an 8-way secure DIP switch for Ethernet cable will become 10/12-way, or an extra 2/4-way secure DIP switch. The slider **103** can be wiped transversely to the pin pairs **104** and **105** to either switch on the networking connection and off the connection of the second storage device, or oppositely. This means secure DIP switch is 10/12-way nPDT (n Poles Double Throws).

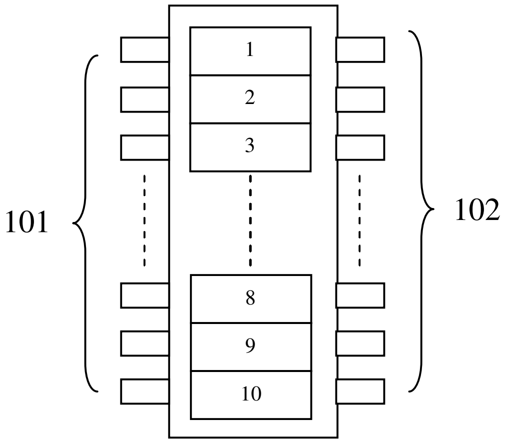


Figure 14.1 Structural diagram of conventional 10-way DIP switch

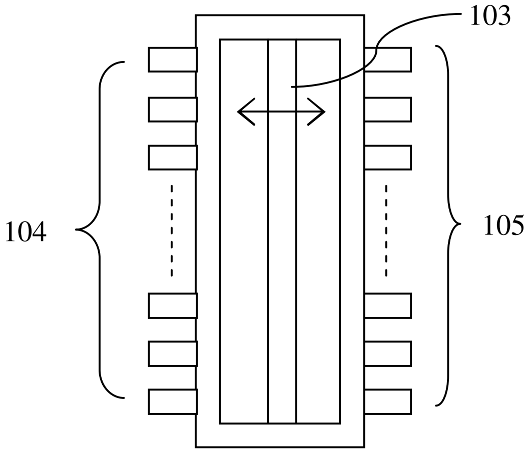


Figure 14.2 Structural diagram of proposed 10/12-way secure DIP switch

There are two groups of poles in opposite connections: 8-way RJ45/RJ11 networking connection and 2/4-way USB connection. 10Mbps and 100Mbps Ethernet over twisted pair can use 4-way connection, but 1Gbps/1000Mbps Ethernet must use 8-way connection. For USB connection, it can be 4 ways or 2 ways by saving the power and ground cables. It is then integrated with two RJ45 sockets and two USB sockets to form a simple and cost-effective innovation (Lee, 2008a, 2008b, 2008c).

If Category 5/5e cable (Wikipedia Contributors, 2007p) defined in ANSI/TIA/EIA-568-A and TIA/EIA-568-B (Wikipedia Contributors, 2007k), respectively, is used, the RJ45 socket will be backwards compatible with RJ11 for two running pairs and one running pair, respectively.

14.5 Method and Device to Secure Hacking-Free Data Storage

Insofar as the secure DIP switch is specifically designed for a method and device to secure a hacking-free data storage. An 8/10-way secure DIP switch is integrated with two modular connector RJ45 sockets to connect or disconnect the networking connection, and two optional USB sockets to oppositely disconnect or connect the second storage device on a PCB (Printed Circuit Board). The integration without USB sockets functioning as an *RJ switch* can be implemented as a wall plate for new installation or as an external interconnection box for old design and inconvenient switch access.

For the computer of the end user, a second storage device is needed. This can be either an internal or external hard disk drive (HDD). It can also be a USB flash drive. For external HDD and USB flash drive, they are hot-swappable when USB port is used. For internal HDD of the type of SATA (Serial Advanced Technology Attachment), a switch is needed to control the data connection. This switch called *HDD switch* can be an 8-way secure DIP switch installed at the back panel of computer with old design or at the front panel of computer with new design. Similarly, the connection of external HDD and USB flash drive via USB port can adopt a 2/4-way switch. This can get rid of the plug-and-play which can cause reliability problem after frequent plugging and unplugging.

For a real implementation, an RJ switch is constructed as an interconnection box from an 8-way secure DIP switch and two RJ45 sockets. The end user uses a computer connected to an external HDD via USB port. The storage device can also be a USB flash drive. There is an Ethernet cable connecting the computer and the RJ45 socket of the interconnection box. Another Ethernet cable connects the second RJ45 socket of the interconnection box and the networking wall plate. Clearly, these can be easily understood by any normal end user. The 8-way switch can also be made 10/12-way if the optional USB connection is added. Then, there are two operating modes as in Table 14.1.

Table 14.1 Operating modes of method and device to secure hack-proof data storage

Operating Mode	Networking Connection	Second Storage Device
Hacking-free	Disconnected	Connected
Network access	Connected	Disconnected

For secure hacking-free operating mode, the actuator is switched to disconnect the networking connection and then connect the second storage device. The end user can create, open, modify, and store one's private and confidential data in the second storage device. When network access is needed, the second storage device is disconnected and then the network is connected. The end user can now surf the Internet and one's data in the second storage device is safe from hacking via the malicious computer network. Once the demand for network access has finished, the end user can switch back to the hacking-free operating mode to manipulate the private and confidential data.

14.6 Costs and Reliability

The current cost of a DIL switch in Malaysia ranges from MYR$3.88 to MYR$46.77 depending on the contact ratings of voltage and current as well as the operating life (Farnell, 2007). The FOREX (Foreign Exchange) of USD$1.00 was about MYR$3.50 in September 2007 and October 2008. Mass purchase over 500 pieces can reduce the unit price of DIL switch to MYR$2.56. Subsequently, it can be claimed that the added manufacturing cost is low and yet the added value of hacking-free data storage is high.

The voltage and current of secure DIP switch will depend on the power over the Ethernet cable (i.e. PoE (Power over Ethernet), PoL (Power over LAN) or Inline Power) (Wikipedia Contributors, 2007o), phone cable and USB connection. Supplying power over Ethernet is strongly recommended to follow the IEEE Standard. Clause 33 of “IEEE 802.3-2005 - Section Two” (LAN/MAN Standards Committee, 2005) provides 48 volts DC over two of the four available pairs on a Cat. 3 / Cat. 5 cable with a maximum current of 400 mA for a maximum load power of 15.4 Watts (Wikipedia Contributors, 2007o).

For the Ethernet cable over LAN in Malaysia, it is normally Cat. 5 T568B (Wikipedia Contributors, 2007k, 2007p). Contact rating of phone cable for network usage is below the contact rating of Ethernet cable. If USB power cables travel through the DIP switch, then it is 4 ways and the contact rating is 5.25 V DC and 500 mA. Otherwise, it needs 2 ways and the contact rating of USB data cables is below 2.8 V and 20 mA for high speed USB 2.0.

The reliability (aka operating life or service life) of DIP switch ranges from 1,000 to 35,000 operations. The death of DIP switch depends on the change of contact resistance and the mechanical wear out of the actuator. It is expected that the improvements by Lin (1999) and Tai (2001) can further increased the operating life of DIP switch in parallel with the reduction of manufacturing materials, weight, and cost. It is a question on the balance of costs and reliability.

This innovation is expected to be broadly used in the office environment, where there exists a lot of private and confidential data. If the hacking-free operating mode and network access operating mode are activated once a day for five times per week, then the DIP switch can last for 3.85 years for the DIP switch with operating life of 1,000 operations. The contact ratings, operating life, and cost of DIP switch are closely correlated. Survey and research are needed for optimum manufacturing design and supply chain management.

Yet another potentially broad application for men with good social status and women with good conditions, this hack-proof data storage is also critical to protect their human interaction network, daily itinerary, future plans, and financial accounts from being maliciously conspired by the cronies of organized crime by using the hacked secrets, flash mob approach, and sound snatching.

14.7 Other Forms of Innovation

An innovation of the improved 8-way 8PST DIL switch as in Figures 14.1-14.2 is to become a 10-way 8PST+2PST DIL switch with an actuator activating 8PST and 2PST in opposite direction, where 8PST controls the network connection of RJ-XX and 2PST is extendable to other nPST to control the hot-swappable USB or SATA data/power connection to create a hack-proof data storage as in Figure 14.3.

As in Figures 14.1-14.2, the 8-way 8PST DIL switch acting as RJ switch for wired Ethernet network can be modified to become 4-way 4PST DIL switch acting as hot-swappable USB switch to control the wireless network connection using the wireless USB network adapter operating on the wireless communication protocols like Bluetooth, Wi-Fi, 3G, WiMAX, etc.

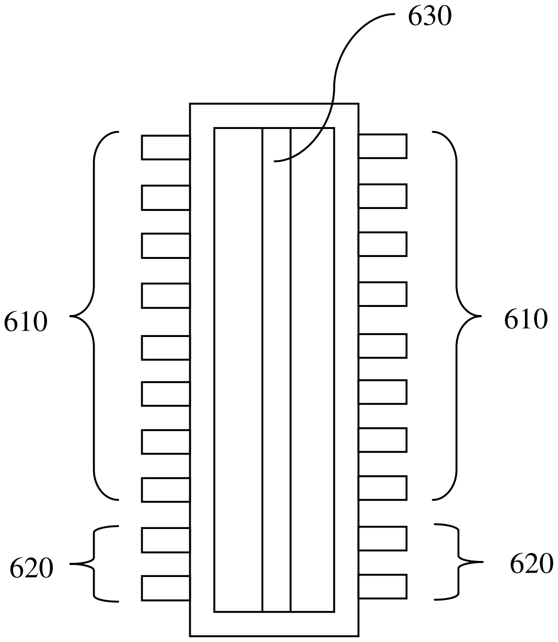


Figure 14.3 Innovated 10-way 8PST+2PST DIL switch activated in opposite direction

In Figure 14.3, the 10-way 8PST+2PST DIL switch with reverse activation **630** can be modified to have the first 8PST **610** acting as RJ switch or to become 4PST acting as a USB switch for wireless USB network adapter in similarity with Figures 14.1-14.2, and the second 2PST **620** is extendable to other nPST for other types of data connection, like SATA and USB, to a storage device like HDD and USB flash drive.

The improved DIL switch so far can be other types of switch performing these enhanced functions to create hacking-free data storage, where they can also switch on and off a few little switches to control the data and power connections like keylock switch, selector switch, pushbutton switch, rocker switch, rotary switch, slide switch, toggle switch, etc., with and without a light indicator of network connection.

There are also some originally novel prototypes for this innovated DIP switch in the forms of layout-design of integrated circuit in Malaysia (Lee, 2005b, 2006c, 2007b, 2007c, 2007d, 2007e, 2008d, 2008e, 2008f, 2008g).

14.8 Conclusion

Unless there is an advanced hacker who can interpret the weak electromagnetic radiation across the secure DIP switch, this proposed method and device for securing a hacking-free data storage can be claimed to be fully resisting the hacking attacks. It is a simple integration consisting of a secure DIP switch, two RJ45 sockets, and two optional USB sockets. The proposed switch adds little manufacturing costs but highly added values, which may be a 10-way switch for a RJ45/RJ11 and a USB connection. This hacking-free method and device is simple, cost-effective, and hack-proof.

CHAPTER 15 CONCLUSIONS

15.1 Concise Summary

In a nutshell, this doctoral research project has contributed a lot of originally novel knowledge contribution in the forms of methods, systems, and devices in the fields of information engineering, generally, and security engineering, particularly.

The ways to evaluate a researcher is firstly discussed followed by the essential condition to qualify for a doctoral degree. Contribution impact by referring to the applications of research results for public usages is highly recommended.

Then in the first part, five methods and systems to create big and yet memorizable secrets are presented in details. These five methods are as follows:

(i) Chinese-character-encoded passphrase

(ii) Two-dimensional (2D) key

(iii) Multilingual key

(iv) Multi-tier geo-image key

(v) Multi-factor multimedia key using software token

(vi) Hybrid combinations of any of the abovementioned five methods

Afterwards, the multimedia noises (or errors), enhanced frequency analysis, information rate, and unicity distance are studied to show how to increase the randomness of password/key to get higher entropy and securer protection.

To cater for the demands of multiple unique secrets to support various offline and online accounts, the multihash key using the hash iteration and hash truncation is invented here. To create more slave keys from a master key, three methods are proposed: Using a filename, using a random number, and using a two-tier structure. Later, there are three variants of multihash key to generate more slave keys. In addition, multihash key is shown on how to act as a further authentication factor, as well as simple key escrow method and system.

In the second part, applications of secret(s) and MePKC (Memorizable Public-Key Cryptography) for twelve novel methods and systems are presented. These applications show the future great contribution impacts that this doctoral research project can trigger and cultivate. These twelve components are as follows:

(i) Memorizable symmetric key to resist quantum computer attack

(ii) Memorizable public-key cryptography (MePKC)

(iii) Other cryptographic, information-hiding, and non-cryptographic applications of secret beyond 128 bits

(iv) Identification hardening of embedded data in steganography

(v) Electronic fund transfer using MePKC

(vi) Electronic software licensing using MePKC

(vii) MePKC human-computer and human-human authentication schemes

(viii) MePKC digital certificate having more than one asymmetric key pair

(ix) Three-tier MePKC digital certificates for ladder authentication

(x) Archiving the voice/video calls of wired/wireless phones

(xi) Multipartite electronic commerce transactions using MePKC

(xii) Trust boosting of MePKC digital certificate by using more than one certification authority and/or introducer of Trust of Web

Later, the MePKC digital timestamping scheme is proposed to act as the digital evidence of copyright authorship and/or ownership to replace the SD (Statutory Declaration) in Malaysia. For generalization, this MePKC digital timestamping scheme can be applied as other IP (Intellectual Property) evidences, especially the patent. After having this scheme, as well as knowing the great loss due to the hacking and economic espionage, an innovated patent filing system called “limited first-to-invent patent filing system” is proposed. This filing system is the same as the first-to-invent patent filing system expect that there is a limited window period from the first date of invention claim to the first patent filing date. This period shall be about one to two years awaiting further research, survey, and evaluation. The MePKC digital timestamping scheme shall be used to act as the IP evidence to prove the first date of invention claim of this limited first-to-invent patent filing system.

Lastly, to secure the plaintext and decrypted ciphertext in the networked computer system, the MePKC by itself is not enough due to the threat of virtual hacking over the malicious and insecure computer communications network. Here, a hack-proof (or hacking-free) data storage using an innovated DIP (Dual In-Line Package) switch together with a second data storage device is proposed to secure the plaintext and decrypted ciphertext. The files in the second data storage are always offline whenever the computer has an Internet connection. When there is a need to use the files in the second data storage, then the DIP switch has to switch off the Internet connection first before activating the second data storage.

Let's create and maintain a networked info-computer age for a more paperless, petroleum-less, and environment-friendly human society by having safer multipartite electronic computer communications as from the original and novel knowledge contribution of this research project.

15.2 Other Supporting Reading Materials in This Research Project

15.2.1 NIST Publication

NIST (National Institute of Standards and Technology), USA, has a lot of publications on information engineering, generally, and security engineering, particularly. These publications are very useful for further reading to discover and cultivate novel and innovative ideas in applying the research outputs of this thesis.

The list of these NIST publications include personal identity verification (PIV) card (Branstad, Clay & Hash, 2005; NIST, 2005a, 2005b, 2006b; Dray, Giles, Kelley & Chandramouli, 2006; McCallister & Ferraiolo, 2006; MacGregor, Schwarzhoff & Mehta, 2007; Polk, Dodson & Burr, 2007; Bailey, Chandramouli, Ghadiali & Branstad, 2008), discrete logarithm cryptography (Barker, Johnson & Smid, 2007), information security handbook (Bowen, Hash & Wilson, 2006; NIST, 2006d, 2007a, 2008; Singhal, Winograd & Scarfone, 2007; Tracy, Jansen, Scarfone & Winograd, 2007), security requirements (NIST, 2001, 2004, 2006a, 2007d; Dent & Mitchell, 2004; Campbell & Easter, 2007a, 2007b, 2007c, 2008), wireless security (Karygiannis & Owens, 2002; Frankel, Eydt, Owens & Scarfone, 2007; Scarfone & Dicoi, 2007), SSL (Frankel, Hoffman, Orebaugh & Park, 2007), RFID (Radio Frequency Identification) (Karygiannis, Eydt, Barber, Bunn & Phillips, 2007), glossary of key information security terms (Kissel, 2006), public-key cryptography (Nechvatal, 1991), computer authentication (NIST, 1985a, 1994b, 1997), key management (NIST, 1992), password (NIST, 1993), escrowed encryption (NIST, 1994a), computer security (NIST, 1995b), HMAC (Keyed-Hash Message Authentication Code) (NIST, 2002a, 2007c), electronic mail (Email) (Tracy, Jansen, Scarfone & Butterfield, 2007), biometrics (Wilson, Grother & Chandramouli, 2007), and application-specific key management guidance (Barker, Burr, Jones, Polk, Rose & Smid, 2008).

15.2.2 Security and Privacy

Besides the security, the entity privacy of an individual, family, and organization has to be considered as well. Debra S. Herrmann (2007) has written a book entitled “Complete Guide to Security and Privacy Metrics” for anyone would like to know more about security and privacy in applying and implementing the research outputs from this thesis.

15.2.3 Other Resources

There are also other resources of reading materials for interested readers to know more. These include etymology dictionary of Malay language (Chong, 1997) to know the relationship among various languages, font formats (Wikipedia Contributors, 2008aj, 2008ak) for various styles to check the possibility to enlarge the key space of multilingual key, computer (Stallings, 2006a), computer communications network security (Stallings, 2000, 2005, 2006b, 2007); cryptography (Koblitz, 1994; Nichols, 1999), practical cryptosystem implementation (Ferguson & Schneier, 2003), open source network security tools (Schiffman, 2003), as well as finger reading (Lee, Tang, Chen & Fang, 2002) to know the existence of

ESP (Extra-Sensory Perception) in particular and psychic abilities in general (Parapsychological Association, No date; Mitchell, 1974; Ostrander & Schroeder, 1974; Liu, 2001; Editors of Time-Life Books, 2004a, 2004b; Wikipedia Contributors, 2008bk, 2008bl, 2008bm, 2008bn, 2008bo, 2008bp) for the leaking possibilities of key the secret.

15.3 Suggestions for Future Research

While reading the recommended supporting reading materials for this research project, readers may also consider developing any of the suggested research topics as discussed in this Section 15.3.

15.3.1 Fixed-Width Font Supporting All the Unicode Graphic Symbols

For current font file format, it can only support up to 65,536 characters or graphic symbols. This is insufficient for multilingual key if all the Unicode graphic symbols have to be included into a single font file. Moreover, the future enlarged font file to support all the Unicode graphic symbols has to be fixed-width font. This is a potential IP (Intellectual Property) of design patent (aka industrial design) and copyright to be developed in the coming future by interested person(s).

15.3.2 512-Bit Multihash Key Needs Hash Function beyond 1024 Bits

So far the popular and security intensively tested hash function is SHA (Secure Hash Algorithm) family. The longest message digest of this SHA is SHA-512 of SHA-2 with 512 bits. This has limited the application of multihash key to 256-bit security for symmetric key and 128-bit security with 30-year protection for asymmetric private key. To achieve the higher security strength at 256 bits of symmetric key strength for 512-bit asymmetric private key, multihash key needs to use 1024-bit hash function to generate 512-bit final slave key.

For 1024-bit hash function, there exists a scalable polymorphic hash function (Roellgen, No date) to achieve this kind of message digest. Nevertheless, its security strength is not well tested by the peer researchers in information security. Therefore, while NIST is in the process of opening an website to accept the recommendation of SHA-3, even though its maximum hash value requirement is 512 bits, related researchers have to prepare themselves to go for a longer message digest up to 1024 bits to realize the 256-bit to 512-bit MePKC (Memorizable Public-Key Cryptography).

15.3.3 MePKC Extension to Other Non-Conventional Cryptographic Schemes

In this thesis, the MePKC has been applied for encryption, digital signature, authentication, digital cheque (aka electronic cheque), software licensing, public-key certificate of public-key infrastructure (PKI), BAP (Byzantine Agreement Protocol), electronic commerce, multihash signature, and digital timestamping.

Besides these conventional cryptographic schemes, interested researchers may apply MePKC for other non-conventional cryptographic schemes like key exchange, blind signature, multisignature, group-oriented signature, undeniable signature, threshold signature, fail-stop signature, group signature, proxy signature, signcryption, forward-secure signature, designated-verifier signature, copy protection, electronic cash, electronic voting, MAC (Message Authentication Code), key escrow, online verification of credit card, etc.

The blind signature scheme includes its further applications for electronic cash (aka e-cash, electronic money, e-money, electronic currency, e-currency, digital cash, digital money, digital currency, or scrip), and electronic voting (aka e-voting, electronic election, e-election, electronic poll, e-poll, digital voting, digital election, or digital poll).

15.3.4 Big Secret(s) for Information-Hiding and Non-Cryptographic Applications

In addition to the big secret(s) applications for cryptographic schemes, Section 9.4 has listed other applications of big secret(s) including the information hiding and non-cryptographic applications. The information-hiding applications include steganography, symmetric watermarking, and asymmetric watermarking. The non-cryptographic applications are to be the seeds of PRNG (Pseudo-Random Number Generator) and CSPRNG (Cryptographically Secure PRNG).

Hence, there are lots of spacious rooms to evaluate the key sizes and corresponding bits of strength of these other applications of big secret(s). It is highly expected for the existence of some literatures about their practically secure key lengths and protection periods like the cryptographic schemes ("Cryptographic Key Length Recommendation," No date; E. Barker, W. Barker, Burr, Polk, & Smid, 2007a, 2007b; Gehrmann & Näslund, 2005, 2006, 2007).

15.3.5 Safety Box Using Computerized Lock

For safety box using computerized lock (Domenicone, 2000), its key pad is purely numeric and the display panel is single-line. The short-term memory limits of digits have been studied by Miller (1956) to be an average of 7 items plus or minus 2 (7 ± 2) (Jones, 2002; Doumont, 2002), and further studies show that they depends on languages (Jones, 2002) in general and phonological short-term memory of 2-second period (Baddeley, Thomson & Buchanan, 1975) in particular. It is 9.9 digits in Chinese language (Hoosain & Salili, 1988) and 5.8 digits in Welsh language (Ellis & Hennelly, 1980).

In other words, for single-line numeric passcode of this type of safety box, a user using English, Chinese, or Welsh language will have a passcode with average entropy of 23.25, 32.89, or 19.27 bits. The strength of these key lengths is insecure whenever a brute force attack can be launched towards the safety box.

Therefore, 2-dimensional (2D) key is highly appreciated to be applied into the safety box using computerized lock. For the key pad, it can remain to be purely numeric in decimal digits or enlarged to become in hexadecimal digits.

15.3.6 Provable Security Studies

The only researcher, who is Kok-Wah Lee @ Xpree Jinhua Li, contributing to the originally novel knowledge in this thesis, is educated in electrical engineering in general and computer communications in particular. Hence, a lot of the proofs of the inventions and innovations here are based on building up engineering prototypes. Consequently, researchers in provable security, who are also mathematicians, are expected to analyze thoroughly the security strength and loopholes of the algorithms, methods, systems, devices, and apparatuses in security engineering as proposed in this thesis.

15.3.7 Statistical Surveys for Various Security Schemes

Besides the provable security research over the inventions and innovations proposed here, researchers in statistics can also consider conducting surveys like some surveys (Adams & Sasse, 1999; Schneier, 2006; Florencio & Herley, 2007) to know about the minimum, mean, maximum, and median key lengths of those methods and systems to create big and yet memorizable secret as proposed here. Similar statistical surveys can also be carried out for multihash key to know the statistical values of master keys and slave keys.

15.4 Conclusions

To emphasize for the thrice time on the imperative aim of this research project, here is the last paragraph.

Let’s create and maintain a networked info-computer age for a more paperless, petroleum-less, and environment-friendly human society by having safer multipartite electronic computer communications as from the original and novel knowledge contribution of this research project.

APPENDIX A WRITING SYSTEMS OF THE WORLD

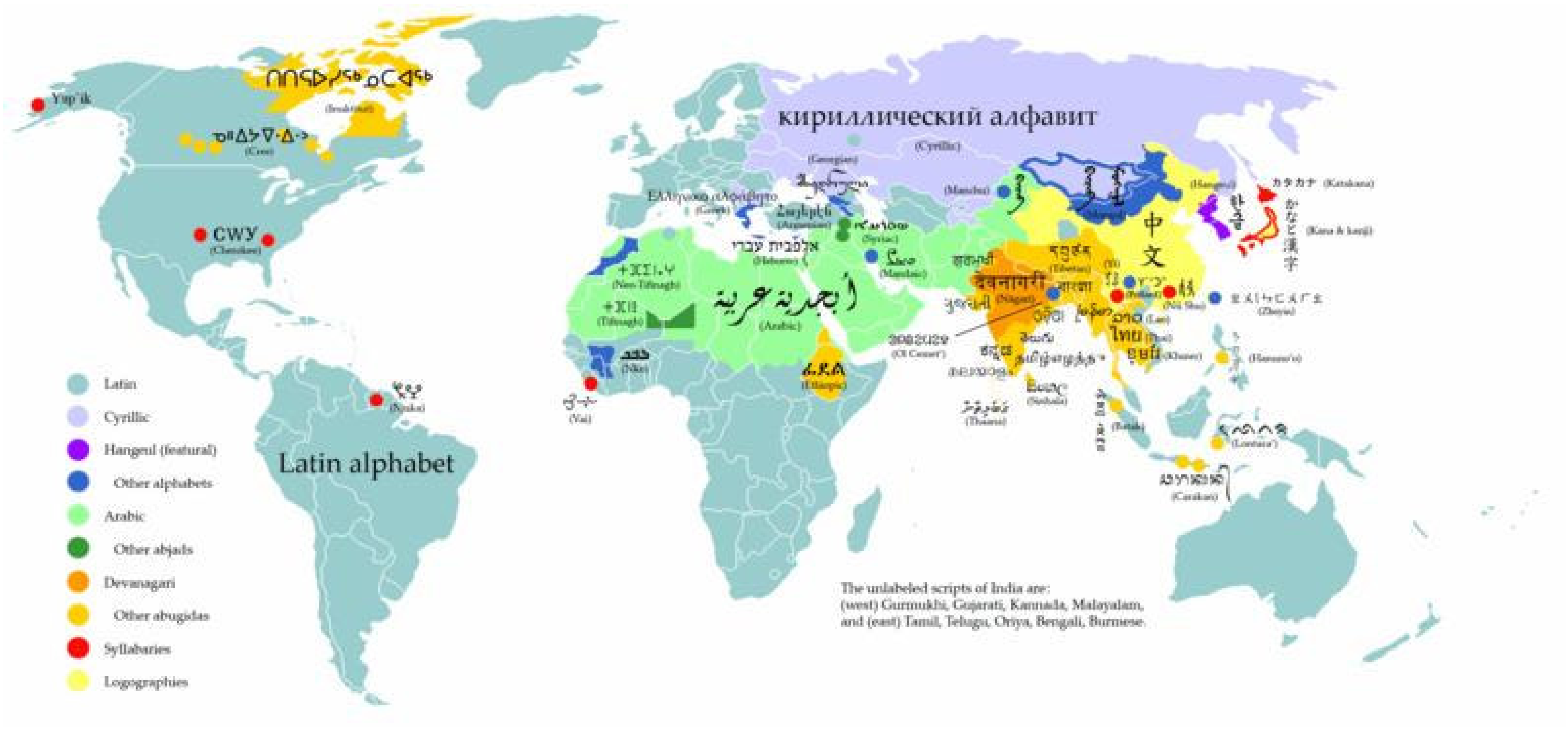


Figure A.1 Writing systems of the world

Reference: Wikipedia Contributors. (2008, August 27). *Writing system*, [Online]. Wikipedia the Free Encyclopedia.

Available: http://en.wikipedia.org/wiki/Image:WritingSystemsoftheWorld4.png [2008, September 1].

Legend of writing systems of the world today:

🟦 Latin (alphabetic)

🟪 Cyrillic (alphabetic)

🟪 Hangul (featural alphabetic)

🟦 Other alphabets

🟩 Arabic (abjad)

🟩 Other abjads

🟧 Devanagari (abugida)

🟨 Other abugidas

🟥 Syllabaries

🟨 Chinese characters (logographic)

Table A.1 Functional classification of writing systems

Type	Symbol Representation	Example
Pictographic	Pictorgram or iconic picture	Hieroglyph, Cuneiform
Ideographic	Ideogram	Way-finding sign, mathematical notation
Logographic	Morpheme	Chinese character
Syllabic	Syllable	Japanese kana
Alphabetic	Phoneme (consonant or vowel)	Latin alphabet
Abugida	Phoneme (consonant + vowel)	Indian Devanāgarī
Abjad	Phoneme (consonant)	Arabic alphabet
Featural	Phonetic feature	Korean hangul

Table A.2 List of languages by number of native speakers

Language	Family	Ethnologue (Y2005)
1. Mandarin	Sino-Tibetan, Chinese	873,000,000
2. Hindi + Urdu	Indo-European, Indo-Iranian, Indo-Aryan	366,000,000
3. Spanish	Indo-European, Italic, Romance	358,000,000
4. English	Indo-European, Germanic, West	341,000,000
5. Arabic	Afro-Asiatic, Semitic	206,000,000
6. Portuguese	Indo-European, Italic, Romance	177,500,000
7. Bengali	Indo-European, Indo-Iranian, Indo-Aryan	171,000,000
8. Russian	Indo-European, Slavic, East	170,000,000
9. Japanese	Japanese-Ryukyuan	122,000,000
10. German	Indo-European, Germanic, West	100,000,000
11. Punjabi	Indo-European, Indo-Iranian, Indo-Aryan	88,000,000
12. French	Indo-European, Italic, Romance	79,572,000
13. Wu	Sino-Tibetan, Chinese	77,200,000
14. Javanese	Austronesian, Malayo-Polynesian, Sunda-Sulawesi	75,500,000
15. Korean	Considered either language isolate or Altaic	74,000,000
16. Telugu	Dravidian, South Central	69,700,000
17. Marathi	Indo-European, Indo-Iranian, Indo-Aryan	68,000,000
18. Vietnamese	Austro-Asiatic, Mon-Khmer, Vietic	67,400,000
19. Tamil	Dravidian, Southern	66,000,000
20. Italian	Indo-European, Italic, Romance	61,500,000
21. Cantonese	Sino-Tibetan, Chinese	54,800,000
22. Sindhi	Indo-European, Indo-Iranian, Indo-Aryan	54,500,000
23. Turkish	Altaic, Turkic, Oghuz	50,625,000
24. Min	Sino-Tibetan, Chinese	46,200,000
25. Gujarati	Indo-European, Indo-Iranian, Indo-Aryan	46,100,000
26. Maithili	Indo-European, Indo-Iranian, Indo-Aryan	45,000,000
27. Polish	Indo-European, Slavic, West	42,700,000
28. Ukrainian	Indo-European, Slavic, East	39,400,000
29. Persian	Indo-European, Indo-Iranian, Iranian	39,400,000
30. Malayalam	Dravidian, Southern - India	35,800,000
31. Kannada	Dravidian, Southern	35,400,000
32. Tamazight	Afro-Asiatic, Berber, Northern	32,300,000

Ref.: Wikipedia Contributors. (2008ad, July 22). List of languages by number of native speakers, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_languages_by_number_of_native_speakers&oldid=227300820 [2008, July 23].

APPENDIX B CHILD-MADE 2D KEYS

Authored by Wei-Dong Chui (徐伟栋), Wei-Jian Chui (徐伟坚), and Kok-Wah Lee (李国华)
in January 2009

In this part, it is shown that children are also capable to create simple 2D keys by using the key styles of ASCII art to draw some Chinese characters. The authors of these child-made 2D keys in January 2009 in this Appendix B are 13-year-old Wei-Jian Chui born in 1996 (Figure B.1) and 9-year-old Wei-Dong Chui born in 2000 (Figure B.2). To get the key size of every 2D key, just multiply the number of ASCII characters of a 2D key by the value of 6.57 bits, or to be more accurate $\log_2 95$. A note here: Kok-Wah LEE being the main author has integrated each four Chinese characters created by them to form a meaningful Chinese phrase for easy remembrance.

```
AAAAAA
AAAAAA
AVVVVA
AAAAAA
VVVVVV
AAAAAA
AAAAAA
```

Two [二]

```
AAVVAA
AAVVAA
AAVVAA
VVVVVV
AAVVAA
AAVVAA
AAVVAA
```

Ten [十]

```
AAAAAA
AAAAAA
AAAAAA
VVVVVV
AAAAAA
AAAAAA
AAAAAA
```

One [一]

```
VVVVVV
VAAAAV
VAAAAV
VVVVVV
VAAAAV
VAAAAV
VVVVVV
```

Day [日]

Figure B.1 2D keys using ASCII art and Chinese characters meaning “twenty one days” [二十一日]

```
AVVVVA
AAAAAA
VVVVVV
AAVAAA
AVAAVA
VVVVVV
AAAAAV
```

Cloud [云]

```
AVVVVA
AAAVAA
AAAVAA
VVVVVV
AAVVAA
AVAAVA
VAAAAV
```

Sky [天]

```
AAAAAA
AAAAAA
VVVVVV
AAVVAA
AAVVAA
VVVVVV
AAAAAA
```

Job [工]

```
AAAVAAA
AAAVAAA
VVVVVVV
AAVVVAA
AVAVAVA
VAAVAAV
AAAVAAA
```

Wood [木]

Figure B.2 2D keys using ASCII art and Chinese characters meaning “cloudy sky nurtures the woods” [云天工木]

REFERENCES

[1] *1790 Analytics Patent Analysis Intellectual Property Evaluation*, [Online]. (No date). Available: http://www.1790analytics.com [2008, September 3].

[2] Abadi, M., Bharat, K., and Marais, J. (2000, October 31). *System and method for generating unique passwords*. USPTO Issued Patent US6141760, Filing Date: 31 October 1997, Issue Date: 31 October 2000.

[3] Abadi, M., Lomas, T. M. A., and Needham, R. (1997, December 16). *Strengthening passwords* (Tech. Rep. No. SRC-1997-033). Palo Alto, CA, USA: Hewlett-Packard Company, HP Labs, Systems Research Center (SRC).

[4] Abadi, M., Needham, R. M., and Lomas, T. M. A. (2000, June 20). *Method and apparatus for strengthening passwords for protection of computer systems*. USPTO Issued Patent US6079021, Filing Date: 2 June 1997, Issue Date: 20 June 2000.

[5] Abril, P. S., and Plant, R. (2007, January). The patent holder's dilemma: buy, sell, or troll? *Communications of the ACM, 50*(1), 37-44.

[6] Adams, A., and Sasse, M. A. (1999, December). Users are not the enemy. *Communications of the ACM, 42*(12), 41-46.

[7] Adams, A., Sasse, M. A., and Lunt, P. (1997, August 12-15). Making Passwords Secure and Usable. *Proceedings of the HCI on People and Computers XII*, Bristol, UK, 1-19.

[8] Aguilera, M. K., and Toueg, S. (1999, August 27). A simple bivalency proof that t-resilient consensus requires t+1 rounds. *Information Processing Letters, 71*(3-4), 155-158.

[9] Aharonian, G. (No date). *Patent/copyright infringement lawsuits/licensing awards*, [Online]. Source Translation and Optimization (STO). Available: http://www.bustpatents.com/awards.htm [2007, October 10].

[10] Ahuja, V. (2000, May/June). Building trust in electronic commerce. *IT Professional, 2*(3), 61-63.

[11] Allan, A. (2004, December 6). *Passwords are near the breaking point* (Tech. Rep. No. Gartner G00124970). Stamford, CT, USA: Gartner, Inc.

[12] Anderson, M., Jaffe, F., Hibbert, C., Virkki, J., Kravitz, J., Chang, S., and Palmer, E. (2000, February 1). *Method and system for processing electronic documents*. USPTO Issued Patent US6021202, Filing Date: 19 December 1997, Issue Date: 1 February 2000.

[13] Anderson, M., Jaffe, F., Hibbert, C., Virkki, J., Kravitz, J., Chang, S., and Palmer, E. (2001, March 27). *Method and system for processing electronic documents*. USPTO Issued Patent US6209095, Filing Date: 31 August 1999, Issue Date: 27 March 2001.

[14] Anderson, R. (2001). *Security engineering: A guide to building dependable distributed systems*. New York, NY, USA: John Wiley & Sons, Inc.

[15] Anderson, R. J., and Petitcolas, F. A. P. (1998, May). On the limits of steganography. *IEEE Journal on Selected Areas in Communications, 16*(4), 474-481.

[16] Arnold, M., Schmucker, M., and Wolthusen, S. D. (2003). *Techinques and applications digital watermarking and content protection*. Norwood, MA, USA: Artech House, Inc.

[17] Aronson, E., and Gerard, E. (1966, March). Beyond Parkinson's law: The effect of excess time on subsequent performance. *Journal of Personality and Social Psychology, 3*(3), 336-339.

[18] Aronson, E., and Landy, D. (1967, July). Further steps beyond Parkinson's Law: A replication and extension of the excess time effect. *Journal of Experimental Social Psychology, 3*(3), 274-285.

[19] Atreya, M., Hammond, B., Paine, S., Starrett, P., and Wu, S. (2002). *Digital signatures*. Berkeley, CA, USA: McGraw-Hill/Osborne.

[20] Baddeley, A. D., Thomson, N., and Buchanan, M. (1975, December). Word length and the structure of short-term memory. *Journal of Verbal Learning and Verbal Behavior, 14*(6), 575-589.

[21] Bailey, D., Chandramouli, R., Ghadiali, N., and Branstad, D. (2008, February). *Guidelines for the accreditation of personal identity verification (PIV) card issuers (PCI's)* (draft) (NIST Special Publication 800-79-1 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[22] Bailey, J. R. (1969, December 23). *Slide switch.* USPTO Issued Patent US3485966, Filing Date: 2 October 1968, Issue Date: 2 October 1968.

[23] Baltzley, C. A. (2000, November 28). *Public key cryptosystem with roaming user capability.* USPTO Issued Patent US6154543, Filing Date: 25 November 1998, Issue Date: 28 November 2000.

[24] Baltzley, C. A. (2001a, August 16). *Public key cryptosystem with roaming user capability.* USPTO Published Patent Application US2001/0014158, Filing Date: 28 March 2001.

[25] Baltzley, C. A. (2001b, September 18). *Public key cryptosystem with roaming user capability.* USPTO Issued Patent US6292895, Filing Date: 19 June 2000, Issue Date: 18 September 2001.

[26] Barham, B., and Foltz, J. (No date). University patenting and scientific advancement, [Online]. Available: http://www.lafollette.wisc.edu/publicservice/stembarhamfoltz.ppt [2008, September 3].

[27] Barker, E., Barker, W., Burr, W., Polk, W., and Smid, M. (2007a, March). *Recommendation for key management – Part 1: General (revised)* (NIST Special Publication 800-57). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST, 61-71.

[28] Barker, E., Barker, W., Burr, W., Polk, W., and Smid, M. (2007b, March). *Recommendation for key management – Part 2: Best practices for key management organization* (NIST Special Publication 800-57). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[29] Barker, E., Burr, W., Jones, A., Polk, T., Rose, S., and Smid, M. (2008, August). *Recommendation for key management – Part 3: Application-specific key management guidance* (NIST Special Publication 800-57). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[30] Barker, E., Johnson, D., and Smid, M. (2007, March). *Recommendation for pair-wise key establishment schemes using discrete logarithm cryptography (revised)* (NIST Special Publication 800-56A). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[31] Barker, E., and Kelsey, J. (2007, March). *Recommendation for random number generation using deterministic random bit generators (revised)* (NIST Special Publication 800-90). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[32] Barney, J. A., and Barney, J. R. (2003, April 29). *Method and system for rating patents and other intangible assets*. USPTO Issued Patent US6556992, Filing Date: 14 September 2000, Issue Date: 29 April 2003.

[33] Barni, M., and Bartolini, F. (2004). *Watermarking systems engineering: Enabling digital assets security and other applications*. New York, NY, USA: Marcel Dekker, Inc.

[34] Batista, P. D., Campiteli, M. G., and Kinouchi, O. (2006, July). Is it possible to compare researchers with different scientific interests?. *Scientometrics, 68*(1), 179-189.

[35] Bauer, F. L. (2002). *Decrypted secrets: Methods and maxims of cryptology* (3rd ed.). Munich, Bavaria, Germany: Springer, 271-300.

[36] Bayh, B. (2006, December). Bayh-Dole: Don't turn back the clock. *Les Nouvelles*, [Online] *2006*(December), 215-218. Licensing Executives Society International (LESI). Available: http://www.lesi.org/BirchBayh/Bayh.pdf [2008, July 17].

[37] Beach, G. [2001, April 15]. *How do we manage our expanding collections of passwords and PINs?*, [Online]. CIO.com. Available: http://www.cio.com/article/print/30161 [2008, September 10].

[38] Beaver, K. (2004). *Hacking for dummies*. Indianapolis, Indiana, USA: Wiley Publishing, Inc.

[39] Behr, F. Jr., Fossum, V., Mitzenmacher, M., and Xiao, D. (2002). *Estimating and comparing entropies across written natural languages using PPM compression* (Tech. Rep. No. TR-12-02), [Online]. Cambridge, MA, USA: Harvard University, Harvard School of Engineering and Applied Sciences. Available: ftp://ftp.deas.harvard.edu/techreports/tr-2002.html; ftp://ftp.deas.harvard.edu/techreports/tr-12-02.ps.gz [2008, July 23].

[40] Behr, F., Fossum, V., Mitzenmacher, M., and Xiao, D. (2003, March 25-27). Estimating and comparing entropies across written natural languages using PPM compression. *Proceedings of the Data Compression Conference 2003 (DCC 2003)*, Snowbird, Utah, USA, 416.

[41] Bellare, M., Miner, S. K. (1999, August 15-19). A forward-secure digital signature scheme. *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology 1999 (CRYPTO '99)*, LNCS 1666, Santa Barbara, CA, USA, 431-448.

[42] Benaloh, J., and de Mare, M. (1991). *Efficient broadcast time-stamping (extended abstract)*, [Online]. Technical Report 1, Department of Mathematics and Computer Science, Clarkson University. CiteSeer. Available: http://citeseer.ist.psu.edu/benaloh91efficient.html [2007, January 10].

[43] Berinato, S. (2007a, September 17). *Hacker economics 1: Malware as a service - Who's stealing your passwords? Global hackers create a new online crime economy*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135500 [2008, September 10].

[44] Berinato, S. (2007b, October 8). *Hacker economics 2: The conspiracy of apathy*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135550 [2008, September 10].

[45] Berinato, S. (2007c, October 8). *Hacker economics 3: MPACK and the next wave of malware*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135551 [2008, September 10].

[46] Berinato, S. (2007d, October 8). *Key malware terms - A layman's glossary of malware terms*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135453 [2008, September 10].

[47] Berinato, S. (2007e, October 8). *How Gozi's first second unfolds*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135451 [2008, September 10].

[48] Berinato, S. (2007f, October 8). *Death by iFrame*, [Online]. CIO.com. Available: http://www.cio.com/article/print/135452 [2008, September 10].

[49] Bessen, J. (2004). *Where have all the great inventors gone*?, [Online]. Research on Innovation. Available: http://www.researchoninnovation.org/GreatInventors.pdf [2008, August 20].

[50] Beutelspacher, A. (1994). *Cryptology* (J. C. Fisher, Trans.). Washington, D.C., USA: The Mathematical Association of America. (Original work published 1991 in German language).

[51] Bierly, P. E. III, Kolodinsky, R. W., and Charette, B. J. (No date). Understanding the complex relationship between creativity and ethical ideologies. *Journal of Business Ethics* [Online]. Available: http://dx.doi.org/10.1007/s10551-008-9837-6 [2008, July 18].

[52] Bishop, M. (2003). *Computer security: Art and science*. Boston, MA, USA: Addison-Wesley.

[53] Blake, I., Seroussi, G., and Smart, N. (1999, July). *Elliptic curves in cryptography*. In London Mathematical Society Lecture Note Series (No. 265). Cambridge, Cambridgeshire, UK: Cambridge University Press.

[54] Blake, I. F., Seroussi, G., and Smart, N. P. (Eds.). (2005, April). *Advances in elliptic curve cryptography*. In London Mathematical Society Lecture Note Series (No. 317). Cambridge, UK: Cambridge University Press.

[55] Blaze, M., Diffie, W., Rivest, R. L., Schneier, B., Shimomura, T., Thompson, E., and Wiener, M. (1996). *Minimal key lengths for symmetric ciphers to provide adequate commercial security*. Chicago, Illinois, USA: A Report by an Ad Hoc Group of Cryptographers and Computer Scientists.

[56] Blonder, G. (1996, September 24). *Graphical password*. USPTO Issued Patent US5559961, Filing Date: 30 August 1995, Issue Date: 24 September 1996.

[57] Boatwright, M., and Luo, X. (2007, September 28-29). What do we know about biometrics authentication? *ACM Proceedings of the 4th Annual Conference on Information Security Curriculum Development 2007*, Kennesaw, Georgia, USA, 205-209.

[58] Boneh, D., and Franklin, M. (2006, September 26). *Systems and methods for identity-based encryption and related cryptographic techniques*. USPTO Issued Patent US7113594, Filing Date: 13 August 2002, Issue Date: 26 September 2006.

[59] Bonnet, M., Baroniunas, W., and Webbink, M. (2008, August 28). *Methods and systems for tracking and auditing intellectual property in packages of open source software*. USPTO Published Patent Application US2008/0209399, Filing Date: 27 February 2007.

[60] Borenstein, N., and Freed, N. (1992, June). Base64 content-transfer-encoding. In *Request for comments (1341): MIME (Multipurpose Internet Mail Extensions): Mechanisms for specifying and describing the format of Internet message bodies* (RFC 1341). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[61] Bose, B. K. (2006). Tips for getting a transactions article published, [Online]. *IEEE Power Electronics Society Newsletter, 2006*(Second Quarter). Available: http://vlsi-india.org/vsi/download/publications/archive/ieee-trans-tips.pdf [2008, August 2].

[62] Bowen, P., Hash, J., and Wilson, M. (2006, October). *Information security handbook: A guide for managers* (NIST Special Publication 800-100). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[63] Boyd, C. (1989). *Digital multisignatures*, [Online]. In H. J. Beker, and F. C. Piper (Eds.), Cryptography and Coding (pp. 241-246). Oxford, UK: Oxford University Press. Available: http://sky.fit.qut.edu.au/~boydc/papers/ima89.pdf [2008, July 15].

[64] Brainard, J., Juels, A., Rivest, R. L., Szydlo, M., and Yung, M. (2006, October 30 – November 3). Fourth-factor authentication: Somebody you know. *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS 2006)*, Alexandria, Virginia, USA, 168-178.

[65] Branstad, D., Clay, A., and Hash, J. (2005, July). *Guidelines for the certification and accreditation of PIV card issuing organizations* (NIST Special Publication 800-79). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[66] Branstetter, L., and Ogura, Y. (2005, August). Is academic science driving a surge in industrial innovation? evidence from patent citations, [Online]. NBER Working Paper No. 11561. Available: http://www.nber.org/papers/w11561 [2008, September 16].

[67] Bray, M. J., and Lee, J. N. (2000, September-November). University revenues from technology transfer Licensing fees vs. equity positions. *Journal of Business Venturing, 15*(5-6), 385-392.

[68] Breitzman, A. (2006, April 13). Analysis of European patent referencing to IEEE papers, conferences, and standards, [Online]. IEEE. Available: http://www.ieee.org/portal/innovate/freecontent/contentitems/whitepaper2.html [2008, September 3].

[69] Breitzman, A. (2007, March 15). *Analysis of patent referencing to IEEE papers, conferences, and standards 1997-2006*, [Online]. IEEE. Available: http://www.ieee.org/portal/cms_docs_iportals/iportals/publications/patentcitation/IEEE_Report_3-15-2006.pdf.pdf [2008, September 3].

[70] Breitzman, A. (2008, May 21). *Analysis of patent referencing to IEEE papers, conferences, and standards 1997-2007*, [Online]. IEEE. Available: http://www.ieee.org/portal/cms_docs_innovate/innovate/freecontent/pdfs/IEEE_and_Patents_5-21-08.pdf [2008, September 3].

[71] Breitzman, A. F., and Narin, F. (2001, January 16). *Method and apparatus for choosing a stock portfolio, based on patent indicators*. USPTO Issued Patent US6175824, Filing Date: 14 July 1999, Issue Date: 16 January 2001.

[72] Brown, A. S., Bracken, E., Zoccoli, S., and Douglas, K. (2004, June 15). Generating and remembering passwords. *Applied Cognitive Psychology, 18*(6), 641-651.

[73] Brown, P. F., Pietra, V. J. D., Mercer, R. L., Pietra, S. A. D., and Lai, J. C. (1982, March). An estimate of an upper bound for the entropy of English. *Computational Linguistics, 18*(1), 31-40.

[74] Brown, R. P. (1983, June 21). *Side actuated miniature DIP switch*. USPTO Issued Patent US4389549, Filing Date: 23 November 1981, Issue Date: 21 June 1983.

[75] Buck. W. (2000a). *Mahabharata*. New Delhi, India: Motilal Banarsidass.

[76] Buck. W. (2000b). *Ramayana*. New Delhi, India: Motilal Banarsidass.

[77] Bugaj, S. V. (1996, June). Passwords for real humans. *USENIX Login, 21*(3), 41.

[78] Burgess, C., and Power, R. (2008, February 8). *Secrets stolen, fortunes lost: Preventing intellectual property theft and economic espionage in the 21st century*. Burlington, MA, USA: Syngress Publishing (now is at Elsevier's Science & Technology publishing) (URL: http://www.syngress.com) (URL: http://www.elsevierdirect.com).

[79] Burr, W. E., Dodson, D. F., and Polk, W. T. (2004, June). *Electronic authentication guideline* (NIST Special Publication 800-63 Version 1.0). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[80] Burr, W. E., Dodson, D. F., and Polk, W. T. (2006, April). *Electronic authentication guideline* (draft) (NIST Special Publication 800-63 Version 1.0.2 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[81] Burr, W. E., Dodson, D. F., Perlner, R. A., Polk, W. T., Gupta, S., and Nabbus, E. A. (2008, February 20). *Electronic authentication guideline* (NIST Special Publication 800-63-1). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[82] Cachin, C. (1998, April 14-17). An information-theoretic model for steganography. *Proceedings of the 2nd International Workshop on Information Hiding (IH '98)*, LNCS 1525, Portland, Oregon, USA, 306-318.

[83] Callahan, D. W. (2001, August/September). The Ph.D. process. *IEEE Potentials, 20*(3), 6-10.

[84] Cameron, K. (2005, May). *The laws of identity*, [Online]. Microsoft Corporation. Available: http://msdn.microsoft.com/en-us/library/ms996456.aspx [2008, August 25].

[85] Cameron, K., and Jones, M. B. (2006, January). *Design rationale behind the identity metasystem architecture*, [Online]. Microsoft Corporation. Available: http://research.microsoft.com/~mbj/papers/Identity_Metasystem_Design_Rationale.pdf [2008, August 25].

[86] Campbell, J., and Easter, R. J. (2007a, June 14). *Security requirements for cryptographic modules (Annex B): Approved protection profiles for FIPS PUB 140-2* (draft) (NIST FIPS Pub 140-2 Annex B (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[87] Campbell, J., and Easter, R. J. (2007b, October 18). *Security requirements for cryptographic modules (Annex C): Approved random number generators for FIPS PUB 140-2* (draft) (NIST FIPS Pub 140-2 Annex C (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[88] Campbell, J., and Easter, R. J. (2007c, December 18). *Security requirements for cryptographic modules (Annex A): Approved security functions for FIPS PUB 140-2* (draft) (NIST FIPS Pub 140-2 Annex A (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[89] Campbell, J., and Easter, R. J. (2008, January 16). *Security requirements for cryptographic modules (Annex D): Approved key establishment techniques for FIPS PUB 140-2* (draft) (NIST FIPS Pub 140-2 Annex D (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[90] Canning, P. (2006, May 22-25). Monetizing usage of scholarly collections with patent citation analysis. *Proceedings of the 27th IATUL Conference on Embedding Libraries in Learning and Research (2006 IATUL)*, Porto, Norte, Portugal.

[91] *CastleCops*, [Online]. (No date). Available: http://www.castlecops.com [2008, September 25].

[92] Cavoukian, A. (2006, October). *7 laws of identity: The case for privacy-embedded laws of identity in the digital age*, [Online]. IPC (Office of the Information and Privacy Commissioner) in Ontario, Canada. Available: http://www.ipc.on.ca/images/Resources/up-7laws_whitepaper.pdf [2008, August 25].

[93] Cayre, F., Fontaine, C., and Furon, T. (2005a, January 17-20). Watermarking security part one: Theory. *Proceedings of the SPIE Conference on Security, Steganography, and Watermarking of Multimedia Contents VII*, Vol. 5681, San Jose, CA, USA, 746-757.

[94] Cayre, F., Fontaine, C., and Furon, T. (2005b, January 17-20). Watermarking security part two: Practice. *Proceedings of the SPIE Conference on Security, Steganography, and Watermarking of Multimedia Contents VII*, Vol. 5681, San Jose, CA, USA, 758-768.

[95] Cayre, F., Fontaine, C., and Furon, T. (2005c, October). Watermarking security: Theory and practice. *IEEE Transcations on Signal Processing, 53*(10), 3976-3987.

[96] Chandran, P. M. (2007, August 7). *Fully loaded – Realizing true potential of the intangible*, [Online]. IPFrontline Magazine of Intellectual Property and Technology, PatentCafe.com. Available: http://www.ipfrontline.com/depts/article.asp?id=15652&deptid=3 [2007, August 16].

[97] Chatterjee, S. (No date). Does increased equity ownership lead to more strategically involved boards?. *Journal of Business Ethics* [Online]. Available: http://dx.doi.org/10.1007/s10551-008-9797-x [2008, July 20].

[98] Chaum, D. (1982, August 23-25). Blind signatures for untraceable payments. *Proceedings of the 2nd Annual International Cryptology Conference on Advances in Cryptology 1982 (aka Proceedings of CRYPTO ‘82 on Advances in Cryptology) (CRYPTO ‘82)*, Santa Barbara, CA, USA, 199-203.

[99] Chaum, D. L. (1988, July 19). *Blind signature systems*. USPTO Issued Patent US4759063, Filing Date: 22 August 1983, Issue Date: 19 July 1988.

[100] Chaum, D., and van Antwerpen, H. (1989, August 20-24). Undeniable signatures. *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology 1989 (CRYPTO '89)*, LNCS 435, Santa Barbara, CA, USA, 212-216.

[101] Chaum, D., and van Heyst, E. (1991, April 8-11). Group signatures. *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques 1991 (aka Proceedings of the Advances in Cryptology – Eurocrypt '91)*, LNCS 547, Brighton, UK, 257-265.

[102] Chen, B. X. (2008, October 8). *Google's super satellite captures first image*, [Online]. Wired.com. Available: http://blog.wired.com/wiredscience/2008/10/geoeye-1-super.html [2008, October 10].

[103] Chen, M. H. [陈明华]. (2004, November). *有组织犯罪问题对策研究* [Strategic study of organized crime problems]. Beijing [北京], China [中国]: Publishing House of China University of Political Science and Law [中国政法大学出版社]. (in Chinese language).

[104] Chong, M. W. (1997). *Kamus etimologi bahasa Melayu* [Etymology dictionary of Malay language]. Shah Alam, Selangor, Malaysia: Penerbit Fajar Bakti Sdn. Bhd. (in Malay language).

[105] CLJ Legal Network Sdn. Bhd. (1998). Solid Gold Publishers Sdn. Bhd. v. Chan Wee Ho & Ors [1998] 5 CLJ 735. In *Law Database of Malaysia*, [Online]. Available: http://www.cljlaw.com [2008, May 20].

[106] Cohen, H., and Frey, G. (2006). *Handbook of elliptic and hyperelliptic curve cryptography.* Boca Raton, FL, USA: Taylor & Francis Group, Chapman & Hall/CRC.

[107] Collberg, C. S., and Thomborson, C. (2002, August). Watermarking, tamper-proofing, and obfuscation – Tools for software protection. *IEEE Transactions on Software Engineering, 28*(8), 735-746.

[108] Comer, D. (Ed.). (1993). *How to write a PhD dissertation or bedtime reading for people who do not have time to sleep*, [Online]. Available: http://homes.cerias.purdue.edu/~spaf/Archive/dec.html [2008, August 2].

[109] *Computing reviews*, [Online]. (No date). ACM (Association for Computing Machinery). Available: http://www.reviews.com [2008, August 21].

[110] Cooke, M. J., and Lebby, G. L. (1998, March 8-10). An optimal design for multilayer feedforward networks. *Proceedings of the 30th Southeastern Symposium on System Theory 1998 (SSST 1998)*, Morgantown, WV, USA, 507-511.

[111] Corell, S. (2000, May 1). Ten risks of PKI: In favour of smart card-based PKI. *Network Security, 2000*(5), 12-14.

[112] Coster, R. (2002, April). *From first-to-invent to first-to-file: The Canadian experience*, [Online]. American Intellectual Property Law Association, Arlington, VA, US. Available: http://www.torys.com/publications/pdf/ARTech-19T.pdf [2007, October 19].

[113] Cover, T. M., and King, R. C. (1978, July). A convergent gambling estimate of the entropy of English. *IEEE Transactions on Information Theory, 24*(4), 413-421.

[114] Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences* [Online], 24(1), 87–185. Available: http://www.bbsonline.org/documents/a/00/00/04/46/index.html [2008, July 21].

[115] Cox, I. J., Doërr, G., and Furon, T. (2006, November 8-10). Watermarking is not cryptography. *Proceedings of the 5th International Workshop on Digital Watermarking 2006 (IWDW 2006)*, LNCS 4283, Jeju Island (aka Jejudo), Jeju Province (aka Jeju-do), South Korea, 1-15.

[116] Cox, I., Miller, M., Bloom, J., and Fridrich, J. (2007, November 16). *Digital watermarking and steganography* (2nd ed.). San Fransisco, CA, USA: Morgan Kaufmann Publishers, Inc.

[117] Cryer, P. (2000). *The research student's guide to success* (2nd ed.). Buckingham, Buckinghamshire, UK: Open University Press.

[118] Crystal, D. (1999). *The Penguin dictionary of language* (2nd ed.). Middlesex, England: Penguin Books.

[119] *Cryptographic Key Length Recommendation*, [Online]. (No date). Available: http://www.keylength.com [2008, October 23].

[120] Curzon, L. B. (1998). *Dictionary of law* (4th ed.). Kuala Lumpur, Malaysia: International Law Book Services.

[121] Dang, Q. (2007a, July 18). *Randomized hashing digital signatures* (draft) (NIST Special Publication 800-106 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[122] Dang, Q. (2007b, July 18). *Recommendation for using approved hash algorithms* (draft) (NIST Special Publication 800-107 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[123] Davies, J. H. E. (1997, March 4). *Personal identification devices and access control systems.* USPTO Issued Patent US5608387, Filing Date: 26 May 1994, Issue Date: 4 March 1997.

[124] de Angeli, A., Coventry, L., Johnson, G., and Renaud, K. (2005, July). Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *International. Journal of Human-Computer Studies, 63*(1-2), 128-152.

[125] de Koning Gans, G., Hoepman, J.-H., and Garcia F. D. (2008, September 8-11). A practical attack on the MIFARE Classic. *Proceedings of the 8th IFIP WG 8.8/11.2 International Conference on Smart Card Research and Advanced Applications (CARDIS 2008)*, London, England, UK, 267-282.

[126] de Winter, B. (2008, October 7). Researchers show how to crack popular smart cards. *InfoWorld*, [Online]. Available: http://www.infoworld.com/article/08/10/07/Researchers_show_how_to_crack_popular_smart_cards_1.html [2008, October 16].

[127] Deitel, H. M., Deitel, P. J., and Nieto, T. R. (2000). *Internet and world wide web: How to program*. Upper Saddle River, New Jersey, USA: Prentice Hall.

[128] Deitel, H. M., Deitel, P. J., and Nieto, T. R. (2001). *e-Business & e-commerce: How to program*. Upper Saddle River, New Jersey, USA: Prentice Hall.

[129] Dent, A. W., and Mitchell, C. J. (2004, October 31). *User's guide to cryptography and standards*. Norwood, MA, USA: Artech House, Inc.

[130] desJardins, M. (1994, December). How to succeed in graduate school: A guide for students and advisors (Part I of II). *ACM Crossroads* [Online], *1*(2). Available: http://www.acm.org/crossroads/xrds1-2/advice1.html [2008, July 23].

[131] desJardins, M. (1995, February). How to succeed in graduate school: A guide for students and advisors (Part II of II). *ACM Crossroads* [Online], *1*(3). Available: http://www.acm.org/crossroads/xrds1-3/advice2.html [2008, July 23].

[132] Desmedt, Y. (1987, August 16-20). Society and group oriented cryptography: A new concept. *Proceedings of the Conference on the Theory and Applications of Cryptographic Techniques 1987 (aka Proceedings of the Advances in Cryptology – CRYPTO '87) (aka Proceedings of the 7th Annual International Cryptology Conference on Advances in Cryptology 1987)*, LNCS 293, Santa Barbara, CA, USA, 120-127.

[133] Desmedt, Y., and Frankel, Y. (1989, August 20-24). Threshold cryptosystems. *Proceeding of the 9th Annual International Cryptology Conference on Advances in Cryptology 1989 (CRYPTO '89)*, LNCS 435, 307-315.

[134] DeVaney, D. B., Harper, D. J., and Short, T. D. (1987, October 6). *Mobile telephone switching office*. USPTO Issued Patent US4698839, Filing Date: 3 June 1986, Issue Date: 6 October 1987.

[135] Devenport, M. (2005, April). *Inventorship vs authorship: Who has the "write" to patent?*, [Online]. The Johns Hopkins University School of Medicine, Licensing and Technology Development. Available: http://www.hopkinsmedicine.org/webnotes/licensing/0504.cfm [2008, July18].

[136] Dharmarajan, M. R. (2005, June 16). *Method and apparatus for password generation*. USPTO Published Patent Application US2005/0132203, Filing Date: 12 December 2003.

[137] Diffie, W., and Hellman, M. E. (1976, November). New directions in cryptography. *IEEE Transaction on Information Theory, IT-22*(6), 644-654.

[138] Diffie, W., and Woods, W. A. (2006, June 22). *Method for generating mnemonic random passcodes*. USPTO Published Patent Application US2007/0300076, Filing Date: 22 June 2006.

[139] Doggett, J., Jaffe, F. A., and Anderson, M. M. (1997, October 14). *Electronic funds transfer instruments*. USPTO Issued Patent US5677955, Filing Date: 7 April 1995, Issue Date: 14 October 1997.

[140] Domenicone, R. (2000, May 3). *Safety box assembly*. EPO Published Patent Application EP0703341, Filing Date: 25 September 1995.

[141] Donner, I. H. (1999, December 7). *Intellectual property audit system*. USPTO Issued Patent US5999907, Filing Date: 6 December 1993, Issue Date: 7 December 1999.

[142] Donner, I. H. (2001, July 17). *Method of performing intellectual property (IP) audit optionally over network architecture*. USPTO Issued Patent US6263314, Filing Date: 3 March 2000, Issue Date: 17 July 2001.

[143] Doumont, J.-L. (2002, June). Magical numbers: the seven-plus-or-minus-two myth. *IEEE Transactions on Professional Communication, 45*(2), 123-127.

[144] Dray, J. F., Giles, A., Kelley, M., and Chandramouli, R. (2006, September). *PIV card to reader interoperability guidelines* (NIST Special Publication 800-96). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[145] Duane, M. J. (2008, Spring). Lending a hand: The need for public participation in patent examination and beyond. *Chicago-Kent Journal of Intellectual Property, 7*(2), 57-74.

[146] Dudas, J. W. (2007, August 21). *Changes to practice for continued examination filings, patent applications containing patentably indistinct claims, and examination of claims in patent applications (final rule)*, [Online]. US Federal Register Part II (37 CFR Part 1), 72(161), 46715-46843. Department of Commerce, US: Patent and Trademark Office. Available: http://www.ipfrontline.com/depts/article.asp?id=15810&deptid=8; http://www.ipfrontline.com/downloads/ContinuationRulesFinal.pdf [2007, August 22].

[147] Duren, M., and Hosmer, C. (2002, August 6-9). Can digital evidence endure the test of time?. *Proceedings of the 2002 Digital Forensics Research Workshop (DFRWS 2002)*, [Online], Syracuse, New York, USA. Available: http://www.dfrws.org/2002/program.shtml [2007, May 20].

[148] Eastlake, D. 3rd, Crocker, S., and Schiller, J. (1994, December). *Request for comments (1750): Randomness recommendations for security* (RFC 1750). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[149] Eastlake, D. 3rd, and Hansen, T. (2006, July). *Request for comments (4634): US Secure Hash Algorithms (SHA and HMAC-SHA)* (RFC 4634). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[150] Eastlake, D. 3rd, and Jones, P. (2001, September). *Request for comments (3174): US Secure Hash Algorithm 1 (SHA1)* (RFC 3174). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[151] Ebert, L. B. (2004, Fall). Patent grant rates at the United States Patent and Trademark Office. *Chicago-Kent Journal of Intellectual Property, 4*(1), 108-116.

[152] Ebert, L. B. (2005, Spring). Comment on "Patent grant rates at the United States Patent and Trademark Office". *Chicago-Kent Journal of Intellectual Property, 4*(2), 186-195.

[153] Ebert, L. B. (2007, August 8). *Patent grant rate lower than many academics think*, [Online]. IPFrontline Magazine of Intellectual Property and Technology, PatentCafe.com. Available: http://www.ipfrontline.com/depts/article.asp?id=15684&deptid=4 [2007, August 16].

[154] Editors of Time-Life Books (Eds.). (2004a). *Mind over matter*. London, England, UK: Caxton Publishing Group.

[155] Editors of Time-Life Books (Eds.). (2004b). *The psychics*. London, England, UK: Caxton Publishing Group.

[156] Eggers, J. J., Su, J. K., and Girod, B. (2000). *Asymmetric watermarking schemes*, [Online]. Available: http://citeseer.ist.psu.edu/eggers00asymmetric.html [2008, July 17].

[157] Elliott, D. R. (2007, September 11). *Method for obtaining and allocating investment income based on the capitalization of intellectual property*. USPTO Issued Patent US7269566, Filing Date: 20 August 2002, Issue Date: 11 September 2007.

[158] Ellis, N. C., and Hennelly, R. A. (1980). A bilingual word-length effect: Implications for intelligence testing and the relative ease of mental calculation in Welsh and English. *British Journal of Psychology, 71*, 43-51.

[159] Ellison, C., and Schneier, B. (2000). Ten risks of PKI: What you’re not being told about public key infrastructure. *Computer Security Journal, 16*(1), 1-8.

[160] Elphinstone, L., and Schweitzer, R. (1998). *How to get a research degree: A survival guide.* St. Leonards, New South Wales, Australia: Allen & Unwin.

[161] Farnell. (2007). *The Farnell inOne catalogue 2007/2008*, [Book, Online]. Available: http://my.farnell.com [2008, January 24].

[162] Feistel, H. (1974a, March 19). *Block cipher cryptographic system.* USPTO Issued Patent US3798359, Filing Date: 30 June 1971, Issue Date: 19 March 1974.

[163] Feistel, H. (1974b, March 19). *Centralized verification system.* USPTO Issued Patent US3798360, Filing Date: 30 June 1971, Issue Date: 19 March 1974.

[164] Feistel, H. (1974c, March 19). *Step code ciphering system.* USPTO Issued Patent US3798605, Filing Date: 30 June 1971, Issue Date: 19 March 1974.

[165] Ferguson, N., Schneier, B. (2003). *Practical cryptography*. Indianapolis, Indiana, USA: Wiley Publishing, Inc.

[166] Fialka, J. J. (1999, March 1). *War by other means: Economic espionage in America.* New York, NY, USA: W. W. Norton & Company (URL: http://www.wwnorton.com).

[167] Finegan, E. (2004). *Language: Its structure and use* (4th ed.). Boston, MA: Thomson Wardsworth.

[168] Fink, S. (2002, January 15). *Sticky fingers: Managing the global risk of economic espionage* (hardcover). New York, NY, USA: Dearborn Trade Publishing (now called as Kaplan Publishing) (URL: http://www.dearborntrade.com) (URL: http://www.kaplanpublishing.com).

[169] Fink, S. (2003, December 9). *Sticky fingers: Managing the global risk of economic espionage* (paperback). Bloomington, IN, USA: iUniverse.com, a self-publishing company.

[170] Fischer, M., and Lynch, N. (1982, June 13). A lower bound for the time to assure interactive consistency. *Information Processing Letters, 14*(4), 183-186.

[171] Fitzi, M., and Hirt, M. (2006, July 23-26). Optimally efficient multi-valued Byzantine agreement. *Proceedings of the 25th Annual ACM Symposium on Principles of Distributed Computing 2006 (PODC 2006)*, Denver, Colorado, USA, 163-168.

[172] Florencio, D., and Herley, C. (2007, May 8-12). A large-scale study of web password habits. *Proceedings of the 16th ACM International Conference on World Wide Web*, Banff, Alberta, Canada, 657-666.

[173] Fonseca, D. E. (2003, December 9). *Data line switch*. USPTO Issued Patent US6660950, Filing Date: 24 July 2001, Issue Date: 9 December 2003.

[174] Ford, W. (1994). *Computer communications security: Principles, standard protocols and techniques*. Englewood Cliffs, NJ, USA: Prentice Hall.

[175] Ford, W., and Baum, M. S. (2001). *Secure electronic commerce: Building the infrastructure for digital signatures and encryption* (2nd ed.). Upper Saddle River, New Jersey, USA: Prentice Hall.

[176] Fossum, V. (No date). *Entropy, compression, and information content*, [Online]. Available: http://www.eecs.umich.edu/~vfossum/pubs/entropy_explanation.pdf; http://www.cs.iupui.edu/~xkzou/teaching/csci590/entropy_explanation.pdf [2008, July 23].

[177] Frankel, S., Eydt, B., Owens, L., and Scarfone, K. (2007, February). *Establishing wireless robust security networks: A guide to IEEE 802.11i* (NIST Special Publication 800-97). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[178] Frankel, S., Hoffman, P., Orebaugh, A., and Park, R. (2007, August). *Guide to SSL VPNs* (draft) (NIST Special Publication 800-113 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[179] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and Stewart, L. (1999, June). *Request for comments (2617): HTTP authentication: Basic and digest access authentication* (RFC 2617). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[180] Fridrich, J., and Goljan, M. (2004, December 14). *Reliable detection of LSB steganography in color and grayscale images*. USPTO Issued Patent US6831991, Filing Date: 22 June 2001, Issued Date: 14 December 2004.

[181] Fridrich, J., Goljan, M., and Soukal, D. (2004, January 18-22). Searching for the stego-key. *Proceedings of the SPIE on Security, Steganography, and Watermarking of Multimedia Contents VI*, San Jose, California, USA, 70-82.

[182] Fried, I., and Evers, J. (2006, February 14). *Gates: End to passwords in sight*, [Online]. CNET Networks, Inc. Available: http://news.cnet.com/Gates-End-to-passwords-in-sight/2100-7355_3-6039177.html?tag=nw.11 [2008, August 23].

[183] Fumy, W., and Landrock, P. (1993, June). Principles of key management. *IEEE Journal on Selected Areas in Communications, 11*(5), 785-793.

[184] Furht, B., and Kirovski, D. (2006a, April 18). *Multimedia watermarking techniques and applications*. Boca Raton, FL, USA: Taylor & Francis Group, Auerbach Publications.

[185] Furht, B., and Kirovski, D. (2006b, May 3). *Multimedia encryption and authentication techniques and applications*. Boca Raton, FL, USA: Taylor & Francis Group, Auerbach Publications.

[186] Furnell, S. (2005, March). Authenticating ourselves: will we ever escape the password? *Network Security, 2005*(3), 8-13.

[187] Furon, T., and Duhamel, P. (2003, April). An asymmetric watermarking method. *IEEE Transactions on Signal Processing, 51*(4), 981-995.

[188] Furon, T. (2005, September 15-17). A survey of watermarking security. *Proceedings of the 4th International Workshop on Digital Watermarking 2005 (IWDW 2005)*, LNCS 3710, Siena, Tuscany, Italy, 201-215.

[189] Gabber, E., Gibbons, P. B., Matias, Y., and Mayer, A. (1997, February 24-28). How to make personalized web browsing simple, secure, and anonymous. *Proceedings of 1st International Conference on Financial Cryptography 1997 (FC 1997)*, LNCS 1318, Anguilla, British West Indies, 17-31.

[190] Ganesan, R. (1996a, July 9). *Yaksha, an improved system and method for securing communications using split private key asymmetric cryptography*. USPTO Issued Patent US5535276, Filing Date: 9 November 1994, Issue Date: 9 July 1996.

[191] Ganesan, R. (1996b, September 17). *System and method for centralized session key distribution, privacy enhanced messaging and information distribution using a split private key public cryptosystem*. USPTO Issued Patent US5557678, Filing Date: 18 July 1994, Issue Date: 17 September 1996.

[192] Ganesan, R. (1998a, April 7). *Computer system for securing communications using split private key asymmetric cryptography*. USPTO Issued Patent US5737419, Filing Date: 7 June 1996, Issue Date: 7 April 1998).

[193] Ganesan, R. (1998b, May 5). *Securing E-mail communications and encrypted file storage using yaksha split private key asymmetric cryptography*. USPTO Issued Patent US5748735, Filing Date: 7 June 1996, Issue Date: 5 May 1998.

[194] Ganesan, R. (1998c, November 17). *Computer system for centralized session key distribution, privacy enhanced messaging and information distribution using a split private key public cryptosystem*. USPTO Issued Patent US5838792, Filing Date: 8 August 1996, Issue Date: 17 November 1998.

[195] Ganesan, R. (1999, May 18). *Programmed computer for identity verification, forming joint signatures and session key agreement in an RSA public cryptosystem*. USPTO Issued Patent US5905799, Filing Date: 15 October 1996, Issue Date: 18 May 1999.

[196] Ganesan, R., Sandhu, R. S., Cottrell, A. P., and Austin, K. (2006a, May 31). *Augmented single factor split key asymmetric cryptography-key generation and distributor*. USPTO Published Patent Application US2007/0033392, Filing Date: 31 May 2006.

[197] Ganesan, R., Sandhu, R. S., Cottrell, A. P., and Austin, K. (2006b, May 31). *Secure login using augmented single factor split key asymmetric cryptography*. USPTO Published Patent Application US2007/0186095, Filing Date: 31 May 2006.

[198] Ganesan, R., Sandhu, R. S., Cottrell, A. P., and Austin, K. (2006c, May 31). *Secure login using single factor split key asymmetric cryptography and an augmenting factor*. USPTO Published Patent Application US2007/0033393, Filing Date: 31 May 2006.

[199] Ganesan, R., Sandhu, R. S., Cottrell, A. P., Schoppert, B. J., and Bellare, M. (2006, May 2). *Protecting one-time-passwords against man-in-the-middle attacks*. USPTO Published Patent Application, US2007/0033642, Filing Date: 2 May 2006.

[200] Ganesan, R., and Yacobi, Y. (1996, December 24). *System and method for identity verification, forming joint signatures and session key agreement in an RSA public cryptosystem*. USPTO Issued Patent US5588061, Filing Date: 20 July 1994, Issue Date: 24 December 1996.

[201] Garcia, F. D., de Koning Gans, G., Muijrers, R., van Rossum, P., Verdult, R., Schreur, R. W., and Jacobs, B. (2008, October 6-8). Dismantling MIFARE Classic. *Proceedings of the 13th European Symposium on Research in Computer Security (ESORICS 2008)*, Málaga, Andalusia, Spain, 97-114.

[202] Gardiner, D. (2008). *Phishing ativity tends rport Q1/2008*, [Online]. APWG (Anti-Phishing Working Group). Available: http://www.antiphishing.org/reports/apwg_report_Q1_2008.pdf [2008, September 19].

[203] Gehringer, E. F. (2002, June 6-8). Choosing passwords: Security and human factors. *Proceedings of the IEEE International Symposium on Technology and Society (ISTAS 2002)*, Raleigh, North Carolina, USA, 369-373.

[204] Gehrmann, C., and Näslund, M. (Eds.). (2005, March 1). *ECRYPT Yearly report on algorithms and keysizes (2004)* (Report No. IST-2002-507932 D.SPA.10). Katholieke Universiteit Leuven, Leuven-Heverlee, Belgium: European Network of Excellence in Cryptology (ECRYPT).

[205] Gehrmann, C., and Näslund, M. (Eds.). (2006, January 29). *ECRYPT Yearly report on algorithms and keysizes (2005)* (Report No. IST-2002-507932 D.SPA.21). Katholieke Universiteit Leuven, Leuven-Heverlee, Belgium: European Network of Excellence in Cryptology (ECRYPT).

[206] Gehrmann, C., and Näslund, M. (Eds.). (2007, January 26). *ECRYPT Yearly report on algorithms and keysizes (2006)* (Report No. IST-2002-507932 D.SPA.16). Katholieke Universiteit Leuven, Leuven-Heverlee, Belgium: European Network of Excellence in Cryptology (ECRYPT).

[207] Gennaro, R. (2005, April). An improved pseudo-random generator based on the discrete logarithm problem. *Journal of Cryptology, 18*(2), 91-110.

[208] Glick, L. (1995). *Criminology*. Needham Heights, MA, USA: Allyn & Bacon.

[209] Gladden, C. A., and Parelman, M. H. (1979, May 1). *Rapidly deployable emergency communication system.* USPTO Issued Patent US4152647, Filing Date: 23 February 1978, Issue Date: 1 May 1979.

[210] Goal, P. M., and Kriese, S. J. (2004, August 26). *Method and system for automated password generation*. USPTO Published Patent Application US2004/0168068, Filing Date: 20 February 2003.

[211] Goldstein, H. (2006, November). Patent power. *IEEE Spectrum, 43*(11), 44-47.

[212] Goldwasser, S. (1997, October 20-22). New directions in cryptography: twenty some years later (or cryptograpy and complexity theory: a match made in heaven). *Proceedings of the 38th Annual Symposium on Foundations of Computer Science (SFCS 1997)*, Miami Beach, Fl, USA, 314-324.

[213] *Good practices in the protection of witnesses in criminal proceedings involving organized crime*, [Online]. (2008). UNODC (United Nations Office on Drugs and Crime). Available: http://www.unodc.org/documents/organized-crime/Witness-protection-manual-Feb08.pdf [2008, July 21].

[214] Gordon, R. G. Jr. (Ed.). (2005, January 1). *Ethnologue: Languages of the world* (15th ed.), [Online]. Dallas, Texas, USA: SIL International. Available: http://www.ethnologue.com/web.asp [2008, September 15].

[215] Gouda, M. G., Liu, A. X., Leung, L. M., and Alam, M. A. (2005, June 7-10). Single password, multiple accounts. *Proceedings of the 3rd International Conference on Applied Cryptography and Network Security (ACNS 2005)*, LNCS 3531, New York City, NY, USA, Industry / Short Paper Track, 1-12.

[216] Graefe, J. M., Lashley, L., Guimaraes, M. A. M., Guodabia, E., Gupta, A. K., Henry, H., and Austin, R. (2007, September 28-29). Credit card transaction security. *ACM Proceedings of the 4th Annual Conference on Information Security Curriculum Development 2007*, Kennesaw, Georgia, USA, 118-123.

[217] Haber, S., and Stornetta, W. S. (1990, August 11-15). How to time-stamp a digital document. *Proceedings of the 10th Annual International Cryptology Conference on Advances in Cryptology 1990 (CRYPTO '90)*, LNCS 537, Santa Barbara, CA, USA, 437-455.

[218] Haber, S., and Stornetta, W. S. (1991, January). How to time-stamp a digital document. *Journal of Cryptology, 3*(2), 99-111.

[219] Hachez, G., and Quisquater, J.-J. (2002). *Which directions for asymmetric watermarking?*, [Online]. Available: http://citeseer.ist.psu.edu/564734.html [2008, July 17].

[220] Halderman, J. A., Waters, B., and Felten, E. W. (2005, May 10-14). A convenient method for securely managing passwords. *Proceedings of the 14th International Conference on World Wide Web 2005*, Chiba, Japan, 471-479.

[221] Hankerson, D., Menezes, A., and Vanstone, S. (2004). *Guide to elliptic curve cryptography*. New York, NY, USA: Springer-Verlag New York, Inc.

[222] Hankerson, D., Menezes, A., and Vanstone, S. (2005, August). *Guide to elliptic curve cryptography* [椭圆曲线密码学导论] (张焕国, Trans. to Chinese language). Beijing [北京], China [中国]: Publishing House of Electronics Industry [电子工业出版社]. (Original published in 2004 in English language).

[223] Haperen, P. V. (1997, May 16). *Graphical password entry*. UK Published Patent Application GB2313460, Filing Date: 16 May 1997.

[224] Hardesty, E. C. (1975, January 14). *Electrical connecting devices for terminating cords and methods of assembling the devices to cords*. USPTO Issued Patent US3860316, Filing Date: 6 July 1973, Issue Date: 14 January 1975.

[225] Harn, L., and Kresler, T. (1989, July 20). New scheme for digital multisignatures. *Electronics Letters, 25*(15), 1002-1003.

[226] Hart, G. W. (1994, September). To decode short cryptograms. *Communications of the ACM, 37*(9), 102-108.

[227] Hartung, F., and Kutter, M. (1999, July). Multimedia watermarking techniques. *Proceedings of the IEEE, 87*(7), 1079-1107.

[228] Haskin, J. (2008, January 11). *Busting the 10 myths about data protection*, [Online]. CIO.com. Available: http://www.cio.com/article/print/171551 [2008, September 10].

[229] Hayes, J. (2001). *Fiber optics technician’s manual* (2nd ed.). Albany, NY, USA: Thomson Learning, Inc., Delmar.

[230] Haykin, S. (1999). *Neural networks: A comprehensive foundation* (2nd ed.). Upper Saddle River, NJ, USA: Prentice Hall.

[231] Haylock, S. E. (No date). *Fingerprints*, [Online]. Answers.com. Available: http://www.answers.com/topic/fingerprints-5 [2008, September 1].

[232] He, J. H. [何家弘] (Ed.). (2007, August). *新编犯罪侦查学* [New criminal investigation study]. Beijing [北京], China [中国]: Publishing House of China's Legal System [中国法制出版社]. (in Chinese language).

[233] Herrmann, D. S. (2007, January 22). *Complete guide to security and privacy metrics: Measuring regulatory compliance, operational resilience, and ROI*. Boca Raton, Florida, USA: Auerbach Publications, Taylor & Francis Group.

[234] Hilley, S. (2006, October). Internet war: Picking on the finance sector – Survey More vulnerabilities & phishing. *Computer Fraud & Security, 2006*(10), 2.

[235] Hirsch, J. E. (2005, November 15). An index to quantify an individual's scientific research output. *PNAS (Proceedings of the National Academy of Sciences), 102*(46), 16569-16572.

[236] Hoffman, N. E. (1982, June 1). *DIP swtich*. USPTO Issued Patent US4332987, Filing Date: 15 December 1980, Issue Date: 1 June 1982.

[237] Hoosain, R., and Salili, F. (1988). Language differences, working memory, and mathematical ability. *Practical aspects of memory: Current research and issues, 2*, 512-517.

[238] Huang, T. D. (1985, February 19). *Method for encoding Chinese characters*. USPTO Issued Patent US4500872, Filing Date: 18 March 1982, Issue Date: 19 February 1985.

[239] Huang, X. R. [黄秀如]. (2002). *词典的两个世界* [A history of dictionaries]. Taipei, Taiwan (ROC): Net and Books [网路与书]. (in Chinese language).

[240] Hunter, P. (2006, May). Microsoft declares war on phishers. *Computer Fraud & Security, 2006*(5), 15-16.

[241] *Identity Theft Resource Center (ITRC)*, [Online]. (No date). Available: http://www.idtheftcenter.org [2008, October 23].

[242] IEEE Communications Society. (2002). *A brief history of communications*. Piscataway, NJ: IEEE Communications Society.

[243] IEEE Publications. (2002, February 15 – 2008, June 20). *IEEE Publication Services and Products Board (PSPB) Operations Manual*, [Online]. IEEE Publications, Piscataway, NJ, USA. Available: http://www.ieee.org/portal/cms_docs_iportals/iportals/publications/PSPB/opsmanual.pdf [2008, July 18].

[244] Identity-related crime. (2007, November 29-30). In *UNODC and organized crime*, [Online]. UNODC (United Nations Office on Drugs and Crime). Available: http://www.unodc.org/unodc/en/organized-crime/index.html [2008, July 21].

[245] IIPA. (2001). Rock Records (M) Sdn. Bhd. v. Audio One Entertainment Sdn. Bhd. [2005] 1 CLJ 200. In *Special report 301 Malaysia*, [Online]. International Intellectual Property Alliance (IIPA). Available: http://www.iipa.com/rbc/2001/2001SPEC301MALAYSIA.pdf [2007, May 20].

[246] ILBS. (2003, March 20). *Laws of Malaysia: Evidence Act 1950 (Act 56).* Petaling Jaya, Selangor, Malaysia: International Law Book Services (ILBS).

[247] International Dunhuang Project (IDP) [國際敦煌項目]. (No date). *The International Dunhuang Project: The Silk Road online*, [Online]. Available: http://idp.bl.uk (British Library); http://idp.nlc.gov.cn (National Library of China) [2008, July 16].

[248] Inventors Assistance League. (No date). *First to invent vs. first to file*, [Online]. Available: http://www.inventions.org/resources/advisory/first.html [2007, October 22].

[249] *Inventorship vs. authorship*, [Online]. (2008). Albert Einstein College of Medicine of Yeshiva University, Offices of Biotechnology and Business Development. Available: http://www.aecom.yu.edu/biotechnology/page.aspx?id=3316 [2008, July18].

[250] Ismail, S. (2001). *Time-stamping for Malaysia*, [Online]. CiteSeer. Available: http://citeseer.ist.psu.edu/ismail01timestamping.html [2007, January 8].

[251] Itakura, K., and Nakamura, K. (1983, October). A public key cryptosystem suitable for digital multisignatures. *NEC Journal on Research & Development, 71*, 1-8.

[252] Ives, B., Walsh, K. R., Schneider, H. (2004, April). The domino effect of password reuse. *Communications of the ACM, 47*(4), 75-78.

[253] Jablon, D. P. (2006, March 7). *Cryptographic methods for remote authentication.* USPTO Issued Patent US7010692, Filing Date: 9 June 2004, Issue Date: 7 March 2006.

[254] Jacobs, A. (1998). *Dictionary of music* (6th ed.). England, United Kingdom: Penguin Books.

[255] Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaption. *Neural Networks, 1*(4), 295-307.

[256] Jansen, W., Gavrila, S., Korolev, V., Ayers, R., and Swanstrom, R. (2003, July). *Picture password: A visual login technique for mobile devices* (NIST IR 7030). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[257] Jermyn, I., Mayer, A., Monrose, F., Reiter, M., and Rubin, A. (1999, August 23-26). The design and analysis of graphical passwords. *Proceedings of 8th USENIX Security Symposium*, Washington D.C., USA, 1-14.

[258] Jones, D. M. (2002, October). *The 7±2 urban legend*, [Online]. MISRA C Conference. Available: http://citeseer.ist.psu.edu/jones02urban.html; http://www.knosof.co.uk/cbook/misart.pdf [2008, July 20].

[259] Jones, M. B. (2005, May). *Microsoft's vision for an identity metasystem*, [Online]. Microsoft Corporation. Available: http://msdn.microsoft.com/en-us/library/ms996422.aspx [2008, August 25].

[260] Kahn, D. (1996a, May 30 – June 1). The history of steganography. *Proceedings of the 1st International Workshop on Information Hiding (IH '96)*, LNCS 1174, Cambridge, Cambridgeshire, UK, 1-5.

[261] Kahn, D. (1996b, December 5). *The code-breakers: The comprehensive history of secret communication from ancient times to the Internet* (Revised and updated ed.). New York City, NY, USA: Scribner, 93-105, 125.

[262] Kanaley, R. (2001, February 4). *Login error trouble keeping track of all your sign-ons? Here's a place to keep your electronic keys, but you'd better remember the password.* San Jose Mercury News.

[263] Kang, M. M., Park, W. W., and Koo, J. R. (2003, June 28 – July 6). Agent for electronic commerce on the semantic web. *Proceedings of the 7th Korea-Russia International*

Symposium on Science and Technology 2003 (KORUS 2003), Ulsan, Republic of Korea, 2003, Vol. 2, 360-363.

[264] Karp, A. H. (2003, May 6). *Site-specific passwords* (Tech. Rep. No. HPL-2002-39R1). Palo Alto, CA, USA: Hewlett-Packard Company, HP Laboratories Palo Alto, Intelligent Enterprise Technologies Laboratory.

[265] Karp, A. H., and Poe, D. T. (2002, August 2). *System-specific passwords*. USPTO Published Patent Application US2004/0025026, Filing Date: 2 August 2002.

[266] Karygiannis, T., Eydt, B., Barber, G., Bunn, L., and Phillips, T. (2007, April). *Guidelines for securing radio frequency identification (RFID) systems* (NIST Special Publication 800-98). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[267] Karygiannis, T., and Owens, L. (2002, November). *Wireless network security: 802.11, Bluetooth and handheld devices* (NIST Special Publication 800-48). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[268] Kaufman, C., Perlman, R., and Speciner, M. (1995). *Network security: Private communication in a public world*. Upper Saddle River, New Jersey, USA: Prentice Hall, 39-100.

[269] Keller, S. S. (2005, January 31). *NIST-recommended random number generator based on ANSI X9.31 Appendix A.2.4 using the 3-key triple DES and AES algorithms*, [Online]. Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST. Available: http://csrc.nist.gov/groups/STM/cavp/documents/rng/931rngext.pdf [2008, May 19].

[270] Kelsey, J., Schneier, B., Hall, C., and Wagner, D. (1997, September 17-19). Secure applications of low-entropy keys. *Proceedings of the International Workshop on Information Security (ISW '97)*, LNCS 1396, Tatsunokuchi, Ishikawa, Japan, 121-134.

[271] Khatri, N., and Tsang, E. W. K. (2003, April). Antecedents and consequences of cronyism in organizations. *Journal of Business Ethics, 43*(4), 289-303.

[272] Khaw, L. T. (2001). *Copyright law in Malaysia* (2nd ed.). Kuala Lumpur, Malaysia: Malayan Law Journal.

[273] Kini, A., and Choobineh, J. (1998, January 6-9). Trust in electronic commerce: Definition and theoretical considerations. *Proceedings of the 31st Hawaii International Conference on System Sciences 1998 (HICSS 1998)*, Kohala Coast, HI, USA, Vol. 4, 51-61.

[274] Kissel, R. (Ed.). (2006, April 25). *Glossary of key information security terms* (NIST IR 7298). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[275] Klein, D. V. (1990, August 27). "Foiling the cracker": A survey of, and improvements to, password security (revised paper). *Proceedings of the USENIX 2nd Security Workshop Program*, Portland, Oregon, USA, 5-14.

[276] Klimek, P., Hanel, R., and Thurner, S. (2008, April 14). *To how many politicians should government be left?*, [Online]. Cornell University Library, Ithaca, NY, USA. Available: http://arxiv.org/pdf/0804.2202 [2008, July 18].

[277] Koblitz, N. (1994, September 2). *A course in number theory and cryptography* (2nd ed.). New York, NY, USA: Springer-Verlag New York, Inc.

[278] *Komyunitipatentorebyu* [コミュニティパテントレビュー] - *Community patent review*, [Online]. (No date). JPO (Japan Patent Office). Available: http://www.cprtrial-iip.org [2008, August 21].

[279] Konrad, K., Fuchs, G., and Barthel, J. (1999, October 19-22). Trust and electronic commerce – More than a technical problem. *Proceedings of the 18th IEEE Symposium on Reliable Distributed Systems 1999 (RELDIS 1999)*, Lausanne, Switzerland, 360-365.

[280] Kotadia, M. (2004, February 25). *Gates predicts death of the password*, [Online]. CNET Networks, Inc. Available: http://news.cnet.com/Gates-predicts-death-of-the-password/2100-1029_3-5164733.html?tag=nw.4 [2008, August 23].

[281] Kormann, D. P., and Rubin, A. D. (2000, June). Risks of the Passport single signon protocol. *Computer Networks, 33*(1-6), 51-58.

[282] Kučera, H., and Francis, W. N. (1967). *Computational analysis of present-day American English*. Providence, Rhode Island, USA: Brown University Press.

[283] Kuhn, D. R., Hu, V. C., Polk, W. T., and Chang, S.-J. (2001, February 26). *Introduction to public key technology and the federal PKI infrastructure* (NIST Special Publication 800-32). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[284] Kurokawa, K. (1988, July). Quality and innovation [Japanese electronics industry]. *IEEE Circuits and Devices Magazine, 4*(4), 3-8.

[285] Kurokawa, K. (1990, May 13-18). Quality and innovation. *Proceedings of the 1990 IEEE International Conference on Robotics and Automation (ROBOT 1990)*, Cincinnati, OH, USA, Vol. 3, 2180-2184.

[286] Kurokawa, K. (1991, January). Quality and innovation. *IEEE Control Systems Magazine, 11*(1), 47-51.

[287] Kurokawa, K. (1997, April/May). Modeling human interactions. *IEEE Potentials, 16*(2), Part 2, 26-28.

[288] Kurzban, S. A. (1985, Fall/Winter). Easily remembered passphrases – A better approach. *ACM SIGSAC Review, 3*(2-4), (SIGSAC: Special Interest Group on Security, Audit and Control), 10-21.

[289] Lampe, K. (Ed.). (No date). *Definitions of organized crime*, [Online]. Available: http://www.organized-crime.de/OCDEF1.htm [2008, July 21].

[290] Lamport, L. (1983, July). The weak Byzantine Generals Problem. *Journal of the ACM, 30*(3), 668-676.

[291] Lamport, L., Shostak, R., and Pease, M. (1982, July). The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems, 4*(3), 382-401.

[292] LAN/MAN Standards Committee. (2005, December 9). *Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*, [Online]. New York, NY, USA: IEEE Computer Society. Available: Access http://standards.ieee.org/getieee802/802.3.html (to get) http://standards.ieee.org/getieee802/download/802.3-2005_section2.pdf [2008, January 24].

[293] Landy, D., McCue, K., and Aronson, E. (1969, June). Beyond Parkinson's law: III. The effect of protractive and contractive distractions on the wasting of time on subsequent tasks. *Journal of Applied Psychology, 53*(3), Part 1, 236-239.

[294] Lauer, H. C. (No date). *Discussion on Ph.D. thesis proposals in computing science*, [Online]. Available: http://homes.cerias.purdue.edu/~spaf/Archive/Lauer.html [2008, August 2].

[295] Layman, M. D., and Potter, G. W. (1997). *Organized crime*. Upper Saddle River, New Jersey, USA: Prentice-Hall, Inc.

[296] Le, A. V., Matyas, S. M., Johnson, D. B., and Wilkins, J. D. (1993). A public key extension to the common cryptographic architecture. *IBM Systems Journal, 32*(3), 461-485.

[297] Le Quere, P. (2004, March 30). *High speed random number generation*. USPTO Issued Patent US6714955, Filing Date: 13 August 2001, Issue Date: 30 March 2004.

[298] Lee, K. W. (2003, June 5). *Artificial neural network based Byzantine Agreement Protocol*. Unpublished master's thesis, Multimedia University, Bukit Beruang, Melaka, Malaysia.

[299] Lee, K. W. (2005a, July 4-6). Semantic error occurrences in the multimedia communications. *Proceedings of IASTED 2005 International Conference on Education & Technology (ICET2005)*, Calgary, Alberta, Canada, 227-231.

[300] Lee, K. W. (2005b, September 19). *An nPST dual in-line package switch (DIL/DIP switch) linking all the little actuators into a single actuator*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[301] Lee, K. W. (2006a, June 27-28). Byzantine agreement for electronic commerce. *Proceedings of the Regional Computer Science Postgraduate Conference 2006 (USM-ReCSPC06)*, Penang, Malaysia, 114-119.

[302] Lee, K. W. (2006b, November 5). *2-dimensional key input method version 1.1*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[303] Lee, K. W. (2006c, November 29). *An Ethernet cable RJ45 switch for securing data communication and storage using conventional dual in-line package switch (DIL/DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[304] Lee, K. W. (2007a, January). *Account amount estimator of multihash key using key strengthening based on time response*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[305] Lee, K. W. (2007b, October 19). *An (nPST + reverse mPST) dual in-line package switch (DIL/DIP switch) with two bigger actuators linking two groups of switches activated*

oppositely, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[306] Lee, K. W. (2007c, December 11). *An Ethernet cable RJ45 switch for securing data communication and storage using Grayhill 78H02T (double 4PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[307] Lee, K. W. (2007d, December 11). *An Ethernet cable RJ45 switch for securing data communication and storage using Grayhill 78H01T (single 4PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[308] Lee, K. W. (2007e, December 11). *An Ethernet cable RJ45 switch for securing data communication and storage using Grayhill 78F01T (single 2PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[309] Lee, K. W. (2008a, May 8). *An improved dual in-line (DIL) switch for securing data communication and storage*. Malaysia Patent Application PI 20070733, MyIPO, Filing Date: 11 May 2007.

[310] Lee, K. W. (2008b, May 8). *An improved dual in-line (DIL) switch for securing data communication and storage*. PCT Patent Application PCT/MY2008/000040, WIPO, Filing Date: 8 May 2008.

[311] Lee, K. W. [李國華]. (2008c, May 9). *應用於保護資料通訊及存儲安全之改良式雙行開關* [An improved dual in-line (DIL) switch for securing data communication and storage]. Taiwan (ROC) Patent Application TW097117364, TIPO, Filing Date: 9 May 2008. (in Chinese language).

[312] Lee, K. W. (2008d, June 2). *A telephone cable RJ11 switch for securing data communication and storage using Grayhill 78H01T (single 4PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[313] Lee, K. W. (2008e, June 2). *A telephone cable RJ11 switch for securing data communication and storage using Grayhill 78F01T (single 2PST DIP switch)*, [Electronic Component,

Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[314] Lee, K. W. (2008f, June 2). *A USB switch for securing data communication and storage using Grayhill 78H01T (single 4PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[315] Lee, K. W. (2008g, June 2). *A USB switch for securing data communication and storage using Grayhill 78F01T (single 2PST DIP switch)*, [Electronic Component, Layout-Design of Integrated Circuit]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[316] Lee, K. W. (2008h, July 25). *Methods and systems to create big memorizable secrets and their applications in information engineering*. Malaysia Patent Application PI 20082771, MyIPO, Filing Date: 25 July 2008.

[317] Lee, K. W. (2008i, September 21). *2-dimensional key input method with multihash key version 2.0*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[318] Lee, K. W. (2008j, September 21). *Mobile ECC version 2.0*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[319] Lee, K. W. (2008k, September 21). *2-factor multimedia key with multihash key version 1.0*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[320] Lee, K. W., and Ewe, H. T. (2001, October 2001). Artificial neural networks based algorithm for Byzantine Generals Problem. *Proceedings of the MMU International Symposium on Information and Communications Technologies 2001 (MMU-M2USIC 2001)*, Petaling Jaya, Selangor, Malaysia, pp. TS 1.4(1-4).

[321] Lee, K. W., and Ewe, H. T. (2002, July 17-19). Artificial neural network based Byzantine Agreement Protocol. *Proceedings of the 6th IASTED International Conference on Artificial Intelligence and Soft Computing (IASTED-ASC 2002)*, Banff, Alberta, Canada, 368-373.

[322] Lee, K. W., and Ewe, H. T. (2003, October 2-3). Faulty node detection in the tripartite ANN based BAP. *Proceedings of the MMU International Symposium on Information and Communications Technologies 2003 (MMU-M2USIC 2003)*, Petaling Jaya, Selangor, Malaysia, 45-48.

[323] Lee, K. W., and Ewe, H. T. (2006, November 3-6). Coinware for multilingual passphrase generation and its application for Chinese language password. *Proceedings of the 2006 International Conference on Computational Intelligence and Security (CIS 2006)*, Guangzhou, Guangdong, China, 1511-1514 (Part 2).

[324] Lee, K. W., and Ewe, H. T. (2007a, August). Multiple hashes of single key with passcode for multiple accounts. *Journal of Zhejiang University Science A (JZUS-A), 8*(8), 1183-1190.

[325] Lee, K. W., and Ewe, H. T. (2007b, November 1). Performance study of Byzantine Agreement Protocol with artificial neural network. *Information Sciences, 177*(21), 4785-4798.

[326] Lee, K. W., and Ewe, H. T. (2007c, November 12-15). Speeding up the Byzantine Agreement Protocol with artificial neural network (BAP-ANN) using modified backpropagation learning algorithm (BPLA). *Proceedings of the Malaysia-Japan International Symposium on Advanced Technology 2007 (MJISAT 2007)*, Kuala Lumpur, Malaysia, pp. 65 & 169 [CD-ROM, Technical Session T5-5: AI Applications III].

[327] Lee, K. W., and Ewe, H. T. (2007d, December 15-19). Passphrase with semantic noises and a proof on its higher information rate. *Proceedings of the International Conference on Computational Intelligence and Security Workshops (CISW 2007)*, Harbin, Heilongjiang, China, 652-655.

[328] Lee, K. W., Radhakrishna, G., and Khaw, L. T. (2007, November 19-20). The proof of copyright ownership using digital timestamp in Malaysia. *Proceedings of the MMU International Symposium on Information and Communications Technologies 2007 (MMU-M2USIC 2007)*, Petaling Jaya, Selangor, Malaysia, pp. 20 & 37, [CD-ROM, file (f.) TS3A-1.pdf]. Multimedia University, Cyberjaya (Selangor) & Bukit Beruang (Melaka), Malaysia.

[329] Lee, K. W., and Tan, A. W. C. (2006a, November 19). *Multilingual key input method version 1.0*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[330] Lee, K. W., and Tan, A. W. C. (2006b, November 24). *MobileECC version 1.2*, [CD-ROM, Computer Program]. Available: Lee, K. W., Bukit Beruang, Melaka, Malaysia.

[331] Lee, K. W., Teh, C. E., and Tan, Y. L. (2006, May 30-31). *Decrypting English text using enhanced frequency analysis*. Seminar conducted at the meeting of the National Seminar on Science, Technology and Social Sciences (STSS 2006), Kuantan, Pahang, Malaysia.

[332] Lee, S.-C., Tang, D., Chen, C.-T., and Fang, S.-C. (2002, May). Finger-reading: Exploring the information field. *The International Journal of Healing and Caring, 2*(2), 1-25.

[333] Lee, W. C. Y. (1995). *Mobile cellular telecommunications: Analog and digital systems* (2nd ed.). Singapore: McGraw-Hill Book Company, 463-486.

[334] Lemley, M. A., and Moore, K. A. (2004). Ending abuse of patent continuations. *Boston University Law Review, 84*, 63-118.

[335] Lemley, M. A., and Sampat, B. N. (2007, November 9-10). Is the patent office a rubber stamp? *Proceedings of the 2nd Annual Conference on Empirical Legal Studies (CELS 2007)*, New York, NY, USA.

[336] Li, J.-F., Hu, G.-P., Wang, R.-H., and Dai, L.-R. (2005, March 18-23). Sliding window smoothing for maximum entropy based intonational phrase prediction in chinese. *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, Philadelphia, PA, USA, 285-288.

[337] Liataud, J. P., and Maloney, M. L. (1983, March 8). *DIP switch*. USPTO Issued Patent US4376234, Filing Date: 5 May 1981, Issue Date: 8 March 1983.

[338] *Licensing revenue remains steady*, [Online]. (No date). Baltimore, Maryland, USA: School of Medicine, Johns Hopkins University. Available: http://www.hopkinsmedicine.org/webnotes/licensing/0210.cfm [2008, September 3].

[339] Lilly, G. M. (2004, December 7). *Device for and method of one-way cryptographic hashing*. USPTO Issued Patent US6829355, Filing Date: 5 March 2001, Issue Date: 7 December 2004.

[340] Limayem, M., Khalifa, M., and Chin, W. W. (2004, November). Factors motivating software piracy: A longitudinal study. *IEEE Transactions on Engineering Management, 51*(4), 414-425.

[341] Lin, H. C. (1999, October 19). *Dual inline package switch*. USPTO Issued Patent US5967302, Filing Date: 6 March 1998, Issue Date: 19 October 1999.

[342] Liu, Q. Y. [刘清彦] (Trans.). (2001, June). *椭圆曲线公钥密码导引* [Strange talents]. Taipei [台北], Taiwan (ROC) [中华民国]: Lin Yu Cultural Enterprise Co. Ltd. [林郁文化事业有限公司]. (in Chinese language).

[343] Livingston, J. (1996). *Crime & Criminology*. Upper Saddle River, New Jersey, USA: Prentice-Hall, Inc.

[344] Lockard, J. L. (1977, March 15). *Miniature switch with substantial wiping action*. USPTO Issued Patent US4012608, Filing Date: 25 March 1975, Issue Date: 15 March 1977.

[345] Lockard, J. L. (1979, September 18). *Impedance programming DIP switch assembly*. USPTO Issued Patent US4168404, Filing Date: 3 May 1978, Issue Date: 18 September 1979.

[346] *Log This*, [Online]. (No date). Available: http://members.lycos.co.uk/wuul/logthis/readme.html [2008, March 25].

[347] Low, S. H., and Maxemchuk, N. F. (1998, May). Performance comparison of two text marking methods. *IEEE Journal on Selected Areas in Communications, 16*(4), 561-572.

[348] Lu, C.-S. (2002, December 9-11). Wireless multimedia error resilience via a data handling technique. *Proceedings of the 2002 IEEE Workshop on Multimedia Signal Processing*, St. Thomas, US Virgin Islands, USA, 316-319.

[349] Lu, C.-S. (2005). *Multimedia security: Steganography and digital watermarking techniques for protection of intellectual property*. Hershey, PA, USA: Idea Group Publishing.

[350] Luo, H., and Henry, P. (2003, September 7-10). A common password method for protection of multiple accounts. *Proceedings of the 14th IEEE 2003 International Symposium on Personal, Indoor and Mobile Radio Communication (PIMRC 2003)*, Beijing, China, vol. 3, 2749-2754.

[351] Luo, H. Y. [罗华炎]. (1990). *简明汉语语法* [Concise Chinese grammar]. Cheras, Kuala Lumpur, Malaysia: Yakin [雅景]. (in Chinese language).

[352] Luo, H. Y. [罗华炎]. (2003). *现代汉语语法* [Modern Chinese grammar]. Ipoh, Perak, Malaysia: Seni Hijau [艺青]. (in Chinese language).

[353] Lyons-Burke, K. (2000, October). *Federal agency use of public key technology for digital signatures and authentication* (NIST Special Publication 800-25). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[354] MacGregor, W., Schwarzhoff, T., and Mehta, K. (2007, June). *A scheme for PIV visual card topography* (NIST Special Publication 800-104). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[355] MacKenzie, P. D., Oprea, A., and Reiter, M. K. (2003, October 27-30). Automatic generation of two-party computations (extended abstract). *Proceedings of the 10th ACM Conference on Computer and Communications Security 2003*, Washington, D.C., USA, 210-219.

[356] MacKenzie, P. D., and Reiter, M. K. (2001a, May). *Networked cryptographic devices resilient to capture* (Tech. Rep. No. DIMACS TR 2001-19), [Online]. New Jersey, USA: Rutgers (The State University of New Jersey), DIMACS (Center for Discrete Mathematics & Theoretical Computer Science). Available: http://dimacs.rutgers.edu/TechnicalReports/abstracts/2001/2001-19.html [2008, July 16].

[357] MacKenzie, P. D., and Reiter, M. K. (2001b, May 14-16). Networked cryptographic devices resilient to capture. *Proceedings of the 2001 IEEE Symposium on Security and Privacy (S&P 2001)*, Oakland, CA, USA, 12-25.

[358] MacKenzie, P. D., and Reiter, M. K. (2001c, August 19-23). Two-party generation of DSA signatures (extended abstract). *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology (CRYPTO 2001)*, LNCS 2139, Santa Barbara, CA, USA, 137-154.

[359] MacKenzie, P. D., and Reiter, M. K. (2002, June 26). *Methods and apparatus for two-party generation of DSA signatures*. US Published Patent Application 2003/0059041, Filing Date: 26 June 2002.

[360] MacKenzie, P. D., and Reiter, M. K. (2003, November). Networked cryptographic devices resilient to capture. *International Journal of Information Security, 2*(1), 1-20.

[361] MacKenzie, P. D., and Reiter, M. K. (2004, August). Two-party generation of DSA signatures. *International Journal of Information Security, 2*(3-4), 218-239.

[362] MacKenzie, P. D., and Reiter, M. K. (2006, December 12). *Methods and apparatus for providing networked cryptographic devices resilient to capture*. USPTO Issued Patent US7149311, Filing Date: 7 February 2002, Issue Date: 12 December 2006.

[363] Macuch, P. L. (2005, November 22). *Coaxial and DSL cable switch for controlling a computer connection to the Internet*. USPTO Issued Design Patent D511749, Filing Date: 17 November 2003, Issue Date: 22 November 2005.

[364] Maghiros, I., Punie, Y., Delaitre, S., Lignos, E., Rodríguez, C., Ulbrich, M., Cabrera, M., Clements, B., Beslay, L., and van Bavel, R. (2005). *Biometrics at the frontiers: Assessing the impact on society* (Report No. EUR 21585 EN). Seville, Sevilla, Spain: European Commission, Joint Research Centre (JRC), Institute for Prospective Technological Studies (IPTS).

[365] Malackowski, J. E., and Barney, J. A. (2008, June). What is patent quality? A merchant banc's perspective. *les Nouvelles*, [Online] *2008*(June), 123-134. Licensing Executives Society International (LESI). Available: http://www.lesi.org/content/article_of_the_month.aspx [2008, August 18].

[366] Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S. (2003). *Handbook of fingerprint recognition*. New York, NY, USA: Springer-Verlag New York, Inc.

[367] Mannan, M., and van Oorschot, P. C. (2007, September 18-21). Security and usability: The gap in real-world online banking. *Proceedings of the New Security Paradigms Workshop 2007 (NSPW'07)*, North Conway, New Hampshire, USA, 1-14.

[368] Mannan, M., and van Oorschot, P. C. (2008, July 29). Digital objects as passwords. *Proceedings of the 3rd USENIX Workshop on Hot Topics in Security (HotSec '08)*, San Jose, California, USA, 1-6.

[369] Manoharan, S., and Wu, J. (2007, January 2-7). Software licensing: A classification and case study. *Proceedings of the 1st International Conference on the Digital Society 2007 (ICDS 2007)*, Le Gosier, Guadeloupe, French Caribbean, paper 33.

[370] Mambo, M., Usuda, K., and Okamoto, E. (1996, September 20). Proxy signature: Delegation of the power to sign messages. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E79-A*(9), 1338-1353.

[371] Manber, U. (1996). A simple scheme to make passwords based on one-way functions much harder to crack. *Computers & Security, 15*(2), 171-176.

[372] Markoff, J. (2008, October 20). A robot network seeks to enlist your computer. *The New York Times*, [Online]. Available: http://www.nytimes.com/2008/10/21/technology/internet/21botnet.html [2008, October 23].

[373] Marshall, D. (2003). *.NET security programming*. Indianapolis, IN, USA: Wiley.

[374] Matias, Y., Mayer, A., and Silberschatz, A. (1997, December 8-11). Lightweight security primitives for e-commerce. *Proceedings of the USENIX Symposium on Internet Technologies and Systems 1997*, Monterey, California, USA, 95-102.

[375] Matthews, P. H. (1997). *The concise Oxford dictionary of linguistics*. New York, NY: Oxford University Press.

[376] Maurer, U. (2004, June). New approaches to digital evidence. *Proceedings of the IEEE, 92*(6), 933-947.

[377] Maxim, P. S., and Whitehead, P. C. (1998). *Exploring crime* (4th ed.). Woburn, MA, USA: Butterworth-Heinemann.

[378] McCallister, E., and Ferraiolo, H.. (2006, August). *Personal identity verification demonstration summary* (NIST IR 7337). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[379] McClure, S., Scambray, J., and Kurtz, G. (2001, September 26). *Hacking exposed: Network security secrets & solutions* (3rd ed.). Berkeley, California, USA: Osborne / McGraw-Hill.

[380] McCulligh, M. R. (2003, November 4). *Password generation method and system*. USPTO Issued Patent US6643784, Filing Date: 14 December 1998, Issue Date: 4 November 2003.

[381] MCMC. (No date). Register of recognised date/time stamp services (Section 70). In *Registers Under The Digital Signature Act 1997*, [Online]. Malaysian Communications and Multimedia Commission (MCMC). Available: http://www.mcmc.gov.my/registers/DigitalSignature/RecognizesDateTimeStampService/index.asp [2007, May 23].

[382] McMillan, R. (2008a, June 10). *Microsoft hires anti-phishing crusader*, [Online]. CIO.com. Available: http://www.cio.com/article/print/390413 [2008, September 25].

[383] McMillan, R. (2008b, July 1). *Trojan lurks, waiting to steal admin passwords*, [Online]. CIO.com. Available: http://www.cio.com/article/print/421463 [2008, September 10].

[384] McNamara, J. (2003). *Secrets of computer espionage: Tactics and countermeasures*. Indianapolis, Indiana, USA: Wiley Publishing, Inc.

[385] MDC. (2002a, September). Copyright Act and Regulations (Act 332). In *Laws relating to intellectual property and cyberlaws in Malaysia* (pp. 288-408). Kuala Lumpur, Malaysia: MDC.

[386] MDC. (2002b, September). Digital Signature Act and Regulations (Act 562). In *Laws relating to intellectual property and cyberlaws in Malaysia* (pp. 479-591). Kuala Lumpur, Malaysia: MDC.

[387] MDC. (2002c, September). Industrial Designs Act and Regulations (Act 552). In *Laws relating to intellectual property and cyberlaws in Malaysia* (pp. 409-478). Kuala Lumpur, Malaysia: MDC.

[388] MDC. (2002d, September). Layout-Designs of Integrated Circuits Act and Regulations (Act 6001). In *Laws relating to intellectual property and cyberlaws in Malaysia* (pp. 606-625). Kuala Lumpur, Malaysia: MDC.

[389] MDC. (2002e, September). Patents Act and Regulations (Act 291). In *Laws relating to intellectual property and cyberlaws in Malaysia* (pp. 157-287). Kuala Lumpur, Malaysia: MDC.

[390] Menezes, A. J., Oorschot, P. C. V., and Vanstone, S. A. (1996). *Handbook of applied cryptography*. Boca Raton, FL, USA: CRC Press.

[391] Meurer, M. J., and Bessen, J. E. (2008, March 7). *Do patents perform like property?*, [Online]. Boston University School of Law Working Paper No. 08-08. Available: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1103143 [2008, August 20].

[392] Meyer, F. J., and Pradhan, D. K. (1991, April). Consensus with dual failure modes. *IEEE Transactions on Parallel and Distributed Systems, 2*(2), 214 – 222.

[393] Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The Psychological Review, 63*(2), 81-97.

[394] Mitchell, E. D. (1974). *Psychic exploration: A challenge for science*. New York, NY, USA: G. P. Putnam's Sons.

[395] Mittelholzer, T. (1999, September 29 – October 1). An information-theoretic approach to steganography and watermarking. *Proceedings of the 3rd International Workshop on Information Hiding (IH '99)*, LNCS 1768, Dresden, Saxony, Germany, 1-16.

[396] Mohanty, S. P. (1999). *Digital watermarking: A tutorial review*, [Online, Tech. Rep.]. Bangalore, Bayaluseeme, Karnataka, India: Indian Institute of Science. Available: http://citeseer.ist.psu.edu/572262.html [2008, July 17].

[397] Mollin, R. A. (2007a). *Codes: The guide to secrecy from ancient to modern times*. Boca Raton, FL, USA: Chapman & Hall/CRC, Taylor & Francis Group.

[398] Mollin, R. A. (2007b). *An introduction to cryptography* (2nd ed.). Boca Raton, FL, USA: Taylor & Francis Group, Chapman & Hall/CRC.

[399] Moseley, B. E. (2006, February 2). *Method and system for generating passwords*. USPTO Published Patent Application US2006/0026439, Filing Date: 2 August 2004.

[400] Mossinghoff, G. J. (2005, July 26). *Testimony in the US Senate - Perspectives on patents: Harmonization and other matters*, [Online]. Committee on the Judiciary of the United States Senate. Available: http://judiciary.senate.gov/testimony.cfm?id=1582&wit_id=4547 [2007, October 22].

[401] Mother tongue/bilingual literacy programme for ethnic minorities. (No date). *UNESCO Bangkok*, [Online]. Available: http://www.unescobkk.org/index.php?id=222 [2008, October 4].

[402] Moulin, P., and O'Sullivan, J. A. (2003, March). Information-theoretic analysis of information hiding. *IEEE Transactions on Information Theory, 49*(3), 563-593.

[403] MyIPO. (No date). *Copyright*, [Online]. Intellectual Property Corporation of Malaysia (MyIPO). Available: http://www.mipc.gov.my/index.php?option=com_content&task=view&id=18&itemid=19 [2007, May 20].

[404] Nash, A., Duane, W., Joseph, C., and Brink, D. (2001). *PKI: Implementing and managing e-security*. Berkeley, CA, USA: Osborne/McGraw-Hill.

[405] Nasheri, H. (2004, December 6). *Economic espionage and industrial spying*. Cambridge, Cambridgeshire, UK: Cambridge University Press.

[406] Nechvatal, J. (1991, April). *Public-key cryptography* (NIST Special Publication 800-2). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[407] Nguyen, D., and Widrow, B. (1989, June 18-22). The truck backer-upper: An example of self-learning in neural networks. *Proceedings of the International Joint Conference on Neural Networks 1989 (IJCNN 1989)*, Washington, D.C., USA, Vol. 2, 357-363.

[408] Nguyen, D., and Widrow, B. (1990, June 17-21). Improving the learning speed of two-layer neural networks by choosing initial values of the adaptive weights. *Proceedings of the International Joint Conference on Neural Networks 1990 (IJCNN 1990)*, San Diego, CA, USA, Vol. 3, 21-26.

[409] Nichols, R. K. (1999). *ICSA guide to cryptography*. Blacklick, OH, USA: McGraw-Hill.

[410] NIST. (1985a, May 30). *Computer data authentication* (NIST FIPS Pub 113). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[411] NIST. (1985b, May 30). *Password usage* (FIPS Pub 112). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[412] NIST. (1992, April 27). *Key management using ANSI X9.17* (NIST FIPS Pub 171). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[413] NIST. (1993, October 5). *Automated password generator* (NIST FIPS Pub 181). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[414] NIST. (1994a, February 9). *Escrowed encryption standard* (NIST FIPS Pub 185). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[415] NIST. (1994b, September 28). *Guidelines for the use of advanced authentication technology alternatives* (NIST FIPS Pub 190). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[416] NIST. (1995a, April 17). *Secure hash standard* (NIST FIPS Pub 180-1). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[417] NIST. (1995b, October). *An introduction to computer security: The NIST handbook* (NIST Special Publication 800-12). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[418] NIST. (1997, February 18). *Entity authentication using public key cryptography* (NIST FIPS Pub 196). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[419] NIST. (2000, January 27). *Secure signature standard (DSS)* (NIST FIPS Pub 186-2 (+ Change Notice)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[420] NIST. (2001, May 25). *Security requirements for cryptographic modules* (NIST FIPS Pub 140-2). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[421] NIST. (2002a, March 6). *The keyed-hash message authentication code (HMAC)* (NIST FIPS Pub 198). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[422] NIST. (2002b, August 1). *Secure hash standard* (NIST FIPS Pub 180-2 (+ Change Notice to include SHA-224)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[423] NIST. (2004, February). *Standards for security categorization of federal information and information systems* (NIST FIPS Pub 199). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[424] NIST. (2005a, July). *Questions and answers about the certification and accreditation of PIV card issuing organizations*, [Online]. Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST. Available: http://csrc.nist.gov/publications/nistpubs/800-79/sp800-79Q-As.pdf [2008, May 19].

[425] NIST. (2005b, July). *Questions and answers regarding certification and accreditation of PIV card issuing organizations*, [Online]. Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST. Available: http://csrc.nist.gov/publications/nistpubs/800-79/sp800-79Q-As-Part2.pdf [2008, May 19].

[426] NIST. (2006a, March). *Minimum secuirty requirements for federal information and information systems* (NIST FIPS Pub 200). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[427] NIST. (2006b, March). *Personal identity verification (PIV) of federal employees and contractors* (NIST FIPS Pub 201-1 (Change Notice 1)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[428] NIST. (2006c, March 13). *Secure signature standard (DSS)* (draft) (NIST FIPS Pub 186-3 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[429] NIST. (2006d, June). *Roadmap to NIST information security documents*, [Online Brochure]. Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST. Available: http://csrc.nist.gov/publications/CSD_DocsGuide_Trifold.pdf [2008, May 19].

[430] NIST. (2007a, March). *Guide to NIST information security documents*, [Online]. Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST. Available: http://csrc.nist.gov/publications/CSD_DocsGuide.pdf [2008, May 19].

[431] NIST. (2007b, June 12). *Secure hash standard (SHS)* (draft) (NIST FIPS Pub 180-3 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[432] NIST. (2007c, June 12). *The keyed-hash message authentication code (HMAC)* (draft) (NIST FIPS Pub 198-1 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[433] NIST. (2007d, July 13). *Security requirements for cryptographic modules* (draft) (NIST FIPS Pub 140-3 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[434] NIST. (2008, February 7). *Implementation guidance for FIPS PUB 140-2 and the cryptographic module validation program* (NIST FIPS Pub 140-2 IG). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[435] Noble, K. A. (1994). *Changing doctoral degrees: An international perspective*. Buckingham, Buckinghamshire, UK: SRHE (The Society for Research into Higher Education) & Open University Press.

[436] *Ocean Tomo*, [Online]. (No date). Available: http://www.oceantomo.com [2008, August 21].

[437] Ogletree, T. W. (2000). *Practical firewalls*. Indianapolis, Indiana, USA: Macmillan Computer Publishing, Que Corporation.

[438] Oppliger, R., and Rytz, R. (2003, September-October). Digital evidence: Dream and reality. *IEEE Security & Privacy Magazine, 1*(5), 44-48.

[439] Oram, A. (2008, February). Peer to Patent needs your expertise. *Communications of the ACM, 51*(2), 19-20.

[440] Ostrander, S., and Schroeder, L. (1974). *Handbook of psychic discoveries.* New York, NY, USA: Berkley Publishing Corporation.

[441] Pankaj, S. (2005). *Hacking*. Darya Ganj, New Delhi, India: APH Publishing Corporation.

[442] Paradowski, M. B. (No date). *The benefits of multilingualism*, [Online]. Bilingual/Bicultural Family Network. Available: http://www.biculturalfamily.org/benefitsofmultilingualism.html [2008, October 4].

[443] *Parapsychological Association*, [Online]. (No date). Available: http://www.parapsych.org [2008, October 25].

[444] Parkinson, C. N. (1958). *Parkinson's Law: Or the Pursuit of Progress*, [Online]. Available: http://www.adstockweb.com/business-lore/Parkinson's_Law.htm [2008, July18].

[445] Parkinson, C. N. (2002, September 5). *Parkinson's Law: Or the pursuit of progress* (new ed.). London, UK: Penguin Books Ltd.

[446] Parnas, D. L. (2007, November). Stop the numbers game. *Communications of the ACM, 50*(11), 19-21.

[447] *PasswordResearch.com*, [Online]. (No date). Available: http://www.passwordresearch.com [2008, September 10].

[448] Patel, D., and Luo, X. (2007, September 28-29). Take a close look at phishing. *ACM Proceedings of the 4th Annual Conference on Information Security Curriculum Development 2007*, Kennesaw, Georgia, USA, 210-213.

[449] Pathak, V., and Iftode, L. (2006, March). Byzantine fault tolerant public key authentication in peer-to-peer systems. *Computer Networks: The International Journal of Computer and Telecommunications Networking, 50*(4), 579-596.

[450] Pease, M., Shostak, R., and Lamport, L. (1980, April). Reaching agreement in the presence of faults. *Journal of the ACM, 27*(2), 228-234.

[451] *Peer to Patent - Community patent review*, [Online]. (No date). Available: http://www.peertopatent.org [2008, August 21].

[452] Pelzl, J., Wollinger, T., and Paar, C. (2004, April 5-7). High performance arithmetic for special hyperelliptic curve cryptosystems of genus two. *Proceedings of the International Conference on Information Technology: Coding and Computing 2004 (ITCC 2004)*, Las Vegas, Nevada, USA, Vol. 2, 513-517.

[453] Penguin Popular Classics. (1997). *Arabian nights: Ali Baba and the forty thieves.* London, UK: Penguin Books, 100-126.

[454] Petitcolas, F. A. P., Anderson, R. J., and Kuhn, M. G. (1999, July). Information hiding - A survey. *Proceedings of the IEEE: Special Issue on Protection of Multimedia Content, 87*(7), 1062-1078.

[455] Pfitzmann, B. (1996). *Digital signature schemes: General framework and fail-stop signatures*. Berlin, Germany: Springer-Verlag.

[456] Pfitzmann, B., and Waidner, M. (1990). *Formal aspects of fail-stop signatures* (Tech. Rep. No. Interner Bericht 22/90), [Online]. Karlsruhe, Baden-Württemberg, Germany: Universität Karlsruhe, Fakultät für Informatik. Available: http://ftp.arnes.si/pub/packages/crypto-papers/PfWa_90FSS_formal_IB.ps.gz [2008, July 17].

[457] PGP Corporation. (2006). *PGP Desktop 9.0 for windows user's guide*. Palo Alto, California, USA: PGP Corporation, 229-232.

[458] Phillips, E. M., and Pugh, D. S. (2005, June 1). *How to get a PhD* (4th ed.). Maidenhead, Berkshire,UK: McGraw-Hill Education, Open University Press.

[459] Pierer, H. V., and Oetinger, B. V. (Eds.). (2002). *A passion for ideas: How innovators create the new and shape our world*. West Lafayette, Indiana, USA: Purdue University Press.

[460] Platt, J. R. (2006a, August). Benefits of membership: Today's patents rely on IEEE-published science. *IEEE Microwave Magazine, 7*(4), 76-77.

[461] Platt, J. R. (2006b, November). Patents - Today's patents rely on IEEE-published science. *IEEE Signal Processing Magazine, 23*(6), 8 & 12.

[462] Polk, W. T., Dodson, D. F., and Burr, W. E. (2007, August). *Cryptographic algorithms and key sizes for personal identity verification* (NIST Special Publication 800-78-1). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[463] Popular Book [大众书局]. (2003). *汉语拼音学习卡* [Learning cards of Hanyu Pinyin]. Singapore: Popular Book. (in Chinese language).

[464] Pratt, W. (1997, March 5). *Graduate school survival guide*, [Online]. Available: http://www.math.waikato.ac.nz/~seano/grad-school-advice.html [2008, July 21].

[465] *Prepared testimony of Adam Jaffe for 15 Feb. 07*. (2007, February 15). IPBiz.com. Available: http://ipbiz.blogspot.com/2007/02/prepared-testimony-of-adam-jaffe-for-15.html; http://judiciary.house.gov/media/pdfs/jaffe070215.pdf [2008, August 20].

[466] *Project 10^{100}*, [Online]. (2008). Google. Available: http://www.project10tothe100.com/how_it_works.html [2008, October 13].

[467] Raina, K. (2003). *PKI security solutions for the enterprise: Solving HIPAA, E-Paper Act, and other compliance issues*. Indianapolis, Indiana, USA: Wiley Publishing, Inc.

[468] Re, V., Borean, C., Bozzi, C., Carassiti, V., Ramusino, A. C., Piemontese, L., Breon, A. B., Brown, D., Clark, A. R., Goozen, F., Hernikl, C., Kerth, L. T., Gritsan, A., Lynch, G., Perazzo, A., Roe, N. A., Zizka, G., Roberts, D., Schieck, J., Brenna, E., Citterio, A., Lanni, F., Palombo, F., Ratti, L., Manfredi, P. F., Angelini, C., Batignani, G., Bettarini, S., Bondioli, M., Bosi, F., Bucci, F., Calderini, G., Carpinelli, A., Ceccanti, M., Forti, F., Gagliardi, D., Giorgi, M. A., Lusiani, A., Mammini, P., Morganti, M., Morsani, F., Neri, N., Paoloni, E., Profeti, A., Rama, M., Rizzo, G., Sandrelli, F., Simi, G., Triggiani, G., Walsh, J., Burchat, P., Cheng, C., Kirkby, D., Meyer, T. I., Roat, C., Bona, A., Bianchi, F., Gamba, D., Trapani, P., Bosisio, L., Della Ricca, G., Dittongo, S., Lanceri, L., Pompili, A., Poropat, P., Rashevskaia, I., Vuagnin, G., Burke, S., Callahan, D., Campagnari, C., Dahmes, B., Hale, D., Hart, P., Kuznetsova, N., Kyre, S., Levy, S., Long, O., May, J., Mazur, M., Richman, J., Verkerke, W., Witherell, M., Beringer, J., Eisner, A. A., Frey, A., Grillo, A. A., Grothe, A., Johnson, R. P., Kroeger, W., Lockman, W. S., Pulliam, T., Rowe, W., Schmitz, R. E., Seiden, A., Spencer, E. N., Turri, M., Walkoviak, W., Wilder, M., Wilson, M., Charles, E., Elmer, P., Nielsen, J., Orejudos, W., Scott, I., Zobernig, H., and Laplace, S. (2002, December). The BaBar silicon-vertex tracker:

performance, running experience, and radiation-damage studies. *IEEE Transactions on Nuclear Science, 49*(6), Part 2, 3284-3289.

[469] Ren, J., and Taylor, R. N. (2007, June). Automatic and versatile publications ranking for research institutions and scholars. *Communications of the ACM, 50*(6), 81-85.

[470] *Research Evaluation*, [Online]. (No date). Available: http://www.scipol.co.uk/re.htm [2008, September 3].

[471] Risen, W. M., and Covello, D. F. (2000, January 25). *Method of protecting against a change in value of intellectual property, and product providing such protection*. USPTO Issued Patent US6018714, Filing Date: 8 November 1997, Issue Date: 25 January 2000.

[472] Ritter, T. (2001, June 27). *The redundancy of English*, [Online]. Available: http://www.ciphersbyritter.com/NEWS6/REDUN.HTM [2008, July 21].

[473] Rivest, R. (1992, April). *Request for comments (1321): The MD5 Message-Digest Algorithm* (RFC 1321). Sterling, Virginia, USA: Network Working Group, The Internet Engineering Task Force (IETF).

[474] Rivest, R. L., Shamir, A., and Adleman, L. (1978, February). A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM, 21*(2), 120-126.

[475] Rivest, R. L., Shamir, A., and Adleman, L. M. (1983, September 20). *Cryptographic communications system and method*. USPTO Issued Patent US4405829, Filing Date: 14 December 1977, Issue Date: 20 September 1983.

[476] Roellgen, C. B. (No date). *Scalable polymorphic hash function*, [Online]. Available: http://www.pmc-ciphers.com [2008, July 16].

[477] Roellgen, B. (2005, June 15). *Während der laufzeit veränderbare kryptographische methode* [Cryptographic method modifiable during run time]. EPO Issued Patent EP1069508, Filing Date: 4 July 2000, Issue Date: 15 June 2005. (Original work publish 2005 in German language).

[478] Ross, B., Jackson, C., Miyake, N., Boneh, D., and Mitchell, J. C. (2005, July 31 - August 5). Stronger password authentication using browser extensions. *Proceedings of the 14th USENIX Security Symposium (SEC 2005)*, Baltimore, MD, USA, 17-32.

[479] Ruan, F. M. [阮方民], and Wang, X. [王晓]. (2005, August). *有组织犯罪理论 – 中国黑社会性质组织犯罪防治研究* [New theory of organized crime – The research on prevention of ganglands' crime in China]. Hangzhou [杭州], Zhejiang Province [浙江省], China [中国]: Zhejiang University Press [浙江大学出版社]. (in Chinese language).

[480] Rubin, K., and Silverberg, A. (2003, August 17-21). Torus-based cryptography. *Proceedings of the 23rd Annual International Cryptology Conference 2003 (CRYPTO 2003)*, LNCS 2729, Santa Barbara, California, USA, 349-365.

[481] Rugg, G., and Petre, M. (2004). *The unwritten rules of PhD research*. Maidenhead, Berkshire,UK: McGraw-Hill Education, Open University Press.

[482] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S. (2001, May 15). *A statistical test suite for random and pseudorandom number generators for cryptographic applications* (NIST Special Publication 800-22). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[483] Samuelson, P. (1990, August). Should program algorithms be patented?. *Communications of the ACM, 33*(8), 23-27.

[484] Samuelson, P. (2004, June). Why reform the U.S. patent system?. *Communications of the ACM, 47*(6), 19-23.

[485] Sandhu, R., deSa, C., and Ganesan, K. (2003, June 19). *One time password entry to access multiple network sites*. USPTO Published Patent Application US2003/0115452, Filing Date: 19 December 2000.

[486] Sandhu, R., deSa, C., and Ganesan, K. (2005a, April 19). *System and method for password throttling*. USPTO Issued Patent US6883095, Filing Date: 19 December 2000, Issue Date: 19 April 2005.

[487] Sandhu, R., deSa, C., and Ganesan, K. (2005b, September 6). *High security cryptosystem.* USPTO Issued Patent US6940980, Filing Date: 19 December 2000, Issue Date: 6 September 2005.

[488] Sandhu, R., deSa, C., and Ganesan, K. (2005c, November 29). *System and method for crypto-key generation and use in cryptosystem.* USPTO Issued Patent US6970562, Filing Date: 19 December 2000, Issue Date: 29 November 2005.

[489] Sandhu, R., deSa, C., and Ganesan, K. (2006a, March 21). *Secure communications network with user control of authenticated personal information provided to network entities*. USPTO Issued Patent US7017041, Filing Date: 19 December 2000, Issue Date: 21 March 2006.

[490] Sandhu, R., deSa, C., and Ganesan, K. (2006b, May 30). *One time password entry to access multiple network sites.* USPTO Issued Patent US7055032, Filing Date: 21 May 2004, Issue Date: 30 May 2006.

[491] Sandhu, R., deSa, C., and Ganesan, K. (2006c, June 20). *System and method for generation and use of asymmetric crypto-keys each having a public portion and multiple private portions.* USPTO Issued Patent US7065642, Filing Date: 19 December 2000, Issue Date: 20 June 2006.

[492] Sandhu, R., deSa, C., and Ganesan, K. (2006d, June 27). *System and method for authentication in a crypto-system utilizing symmetric and asymmetric crypto-keys*. USPTO Issued Patent US7069435, Filing Date: 19 December 2000, Issue Date: 27 June 2006.

[493] Sandhu, R., deSa, C., and Ganesan, K. (2006e, November 2). *Laddered authentication security using split key asymmetric cryptography*. USPTO Published Patent Application US2006/0248333, Filing Date: 22 June 2006.

[494] Sandhu, R., deSa, C., and Ganesan, K. (2006f, December 12). *Method and system for authorizing generation of asymmetric crypto-keys*. USPTO Issued Patent US7149310, Filing Date: 19 December 2000, Issue Date: 12 December 2006.

[495] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006a, August 17). *Asymmetric key pair having a kiosk mode*. USPTO Published Patent Application US2006/0182276, Filing Date: 14 February 2005.

[496] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006b, August 17). *Roaming utilizing an asymmetric key pair*. USPTO Published Patent Application US2006/0182277, Filing Date: 14 February 2005.

[497] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006c, August 17). *Architecture for asymmetric crypto-key storage*. USPTO Published Patent Application US2006/0182283, Filing Date: 14 February 2005.

[498] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006d, August 17). *Technique for asymmetric crypto-key generation.* USPTO Published Patent Application US2006/0184786, Filing Date: 14 February 2005.

[499] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006e, August 17). *Authentication protocol using a multi-factor asymmetric key pair.* USPTO Published Patent Application US2006/0184787, Filing Date: 14 February 2005.

[500] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2006f, August 17). *Multiple factor private portion of an asymmetric key*. USPTO Published Patent Application US2006/0184788, Filing Date: 14 February 2005.

[501] Sandhu, R. S., Ganesan, R., Cottrell, A. P., Renshaw, T. S., Schoppert, B. J., and Austin, K. (2007, February 12). *Flexible and adjustable authentication in cyberspace*. USPTO Published Patent Application US2007/0199053, Filing Date: 12 February 2007.

[502] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2007a, March 8). *Technique for providing multiple levels of security*. USPTO Published Patent Application US2007/0055878, Filing Date: 14 February 2005.

[503] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2007b, March 22). *Asymmetric crypto-graphy with rolling key security*. USPTO Published Patent Application US2007/0067618, Filing Date: 17 January 2006.

[504] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2007c, November 8). *Multifactor split asymmetric crypto-key with persistent key security*. USPTO Published Patent Application US2007/0258585, Filing Date: 5 May 2006.

[505] Sandhu, R. S., Schoppert, B. J., Ganesan, R., Bellare, M., and deSa, C. J. (2007d, November 8). *Secure login using a multifactor split asymmetric crypto-key with persistent key security.* USPTO Published Patent Application US2007/0258594, Filing Date: 5 May 2006.

[506] Scalet, S. D. (2005, December 1). *How to write good passwords*, [Online]. CIO.com. Available: http://www.csoonline.com/article/print/220721 [2008, September 19].

[507] Scalise, C. T. (1999). *Intellectual property protection reform: Theory, evidence and policy*. Singapore: National University of Singapore, Singapore University Press.

[508] Scarfone, K., and Dicoi, D. (2007, August 2). *Wireless network security for IEEE 802.11a/b/g and Bluetooth* (draft) (NIST Special Publication 800-48 Revision 1 (Draft)). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[509] Schiffman, M. D. (2003). *Building open source network security tools: Components and techniques*. Indianapolis, Indiana, USA: Wiley Publishing Inc.

[510] Schmeh, K. (2001). *Cryptography and public key infrastructure on the Internet*. Chichester, West Sussex, UK: John Wiley & Sons Ltd. (Original work published 2001 in German language).

[511] Schneider, J. (2004, December 9). *Graphical event-based password system*. USPTO Published Patent Application US2004/0250138, Filing Date: 18 April 2003.

[512] Schneier, B. (1996). *Applied cryptography: Protocols, algorithms, and code in C* (2nd ed.). New York City, New York, USA: John Wiley & Sons.

[513] Schneier, B. (2000). *Secrets & lies: Digital security in a networked world*. New York, NY, USA: John Wiley & Sons.

[514] Schneier, B. (2006, December 14). *MySpace passwords aren't so dumb*, [Online]. Wired News. Available: http://www.wired.com/politics/security/commentary/securitymatters/2006/12/72300 [2008, June 11].

[515] Schneier, B. (2007, January 11). *Secure passwords keep you safer*, [Online]. Wired News. Available: http://www.wired.com/politics/security/commentary/securitymatters/2007/01/72458 [2008, June 11].

[516] *Sejarah Melayu* [The Malay Annals]. (1954). Singapore: Malaya Publishing House Ltd. (Original work published 1896 in Jawi language).

[517] Setty, N., and Gentry, R. L. (2002). *Inventorship - A practical guide to determining correct inventorship*, [Online]. Needle & Rosenberg, Atlanta, Georgia. Available: http://www.needlerosenberg.com/Library/w125248.doc [2008, July 18].

[518] Shannon, C. E. (1948, July & October). A mathematical theory of communication. *Bell System Technical Journal, 27*, 379-423 & 623-656.

[519] Shannon, C. E. (1949, October). Communication theory of secrecy systems. *Bell System Technical Journal, 28*, 656-715.

[520] Shannon, C. E. (1951, January). Prediction and entropy of printed english. *Bell System Technical Journal, 30*(1), 50-64.

[521] Shellabear, W. G. (1975). *Sejarah Melayu* [The Malay Annals]. Kuala Lumpur, Malaysia: Penerbit Fajar Bakti Sdn. Bhd. (In Malaysian language).

[522] Shelton, R. H. (2007, March 29). *System and method of licensing intellectual property assets.* USPTO Published Patent Application US2007/0073625, Filing Date: 27 September 2005.

[523] Siegel, L. J. (2005, March 1). *Criminology* (9th ed.). Florence, KY, USA: Cengage Learning, Inc., Wadsworth Publishing (URL: http://www.wadsworth.com) (URL: http://www.cengage.com).

[524] Silverman, J. H. (1986). *The arithmetic of elliptic curve*. New York, NY, USA: Springer-Verlag New York, Inc.

[525] Simmons, G. J. (1984, April 9-11). The subliminal channel and digital signatures. *Proceedings of the Advances in Cryptology (EUROCRYPT '84) (aka Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques 1984)*, LNCS 209, Paris, France, 364-378.

[526] Simmons, G. J. (1998, May). The history of subliminal channels. *IEEE Journal on Selected Areas in Communications, 16*(4), 452-462.

[527] 新加坡女郎患湿疹无指纹纹机场通关检查花 45 分鐘 [Singaporean Female having eczema has no fingerprint spending 45 minutes to get through the airport immigration office]. (2008, August 22-23). *星洲日报* [*Sin Chew Daily*], [Online], p. 17. Available: http://search.sinchew-i.com/node/192066 [2008, September 1].

[528] Singh, S. (1999). *The code book: The secret history of codes & code-breaking*. London, UK: Fourth Estate Limited, 1-32, 79-80.

[529] Singhal, A., Winograd, T., and Scarfone, K. (2007, August). *Guide to security web services* (NIST Special Publication 800-95). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[530] Siu, H. S., Chin, Y. H., and Yang, W. P. (1998a, April). Byzantine agreement in the presence of mixed faults on processors and links. *IEEE Transaction on Parallel and Distributed Systems, 9*(4), 335-345.

[531] Siu, H. S., Chin, Y. H., and Yang, W. P. (1998b, July). Reaching strong consensus in the presence of mixed failure types. *Information Sciences, 108*(1-4), 157-180.

[532] Smith, A. J. (1990, April). The task of the referee. *IEEE Computer, 23*(4), 65-71.

[533] Spafford, E. H. (1992a, May). Opus: Preventing weak password choices. *Computers & Security, 11*(3), 273-278.

[534] Spafford, E. H. (1992b, September 14-17). Observations on reusable password choices. *Proceedings of USENIX 3rd Symposium on UNIX Security*, Baltimore, MD, USA, 299-312.

[535] Spafford, E. H. (1993). *What is a Ph.D. dissertation?*, [Online]. Available: http://spaf.cerias.purdue.edu/~spaf/Archive/spaf.html [2008, August 2].

[536] Spours, P. (2006, April). How to exploit patents for profit, [Online]. *IP Review, 14*. Available: http://www.cpaglobal.com/ip-review-online/1221/how_to_exploit_patents_for_profit [2008, September 3].

[537] Stallings, W. (1995). *Protect your privacy: A guide for PGP users*. Englewood Cliffs, NJ, USA: Prentice Hall.

[538] Stallings, W. (2000). *Network security essentials: Applications and standards.* Upper Saddle River, New Jersey, USA: Prentice Hall.

[539] Stallings, W. (2005). *Wireless communications & networks* (2nd ed.). Upper Saddle River, NJ, USA: Pearson Prentice Hall.

[540] Stallings, W. (2006a). *Computer organization and architecture* (7th ed.). Upper Saddle River, NJ, USA: Pearson Prentice Hall.

[541] Stallings, W. (2006b). *Cryptography and network security: Principles and practices* (4th ed.). Upper Saddle River, NJ, USA: Pearson Prentice Hall.

[542] Stallings, W. (2007). *Data and computer communications* (8th ed.). Upper Saddle River, NJ, USA: Pearson Prentice Hall.

[543] Standing, L. (1973, May). Learning 10,000 pictures. *Quarterly Journal of Experimental Psychology, 25*(2), 207-222.

[544] Standing, L., Conezio, J., and Haber, R. (1970). Perception and memory for pictures: Single-trial learning of 2500 visual stimuli. *Psychonomic Science, 19*(2), 73-74.

[545] Stinson, D. R. (2002). *Cryptography: Theory and practice*. Boca Raton, FL, USA: Chapman & Hall/CRC Press.

[546] Stubblefield, N. B. (1908, May 12). *Wireless telephone*. USPTO Issued Patent US887357, Filing Date: 5 April 1907, Issue Date: 12 May 1908.

[547] Suo, X., Zhu, Y., and Owen, G. S. (2005, December 5-9). Graphical passwords: A survey. *Proceedings of the 21st Annual Computer Security Applications Conference 2005 (ACSAC 2005)*, Tucson, Arizona, USA, 463-472.

[548] Swanson, M. D., Kobayashi, M., and Tewfik, A. H. (1998, June). Multimedia data-embedding and watermarking techniques. *Proceedings of the IEEE, 86*(6), 1064-1087.

[549] Sweet, W. (2007, November). Keeping score in the IP game. *IEEE Spectrum, 44*(11), 39-41.

[550] Tai, C. L. (2001, December 25). *Dual in-line type finger-actuated switch*. USPTO Issued Patent US6333479, Filing Date: 4 January 2001, Issue Date: 25 December 2001.

[551] Tan, C. P. (1981, May). On the entropy of the Malay language. *IEEE Transactions on Information Theory, 27*(3), 383-384.

[552] Tan, C.-P., and Yap, S.-T. (2001, November 1). *Small-sample studies of the entropy of Chinese* (Tech. Rep. No. 6/2000). Kuala Lumpur, Malaysia: University of Malaya, Institute of Mathematical Sciences.

[553] Taplin, R. (2004, October). *Protect and survive: Managing intellectual property in the Far East – The case of Japan*, [Online]. Thomson Scientific. Available: http://scientific.thomson.com/free/ipmatters/bti/8249985/ [2007, May 16].

[554] Tareq Rajab Museum (TSR). (1998-2002). *Manuscripts in the TSR Museum*, [Online]. Tareq Rajab Museum, Hawelli, Kuwait. Available: http://www.trmkt.com/manu.html [2008, July 16].

[555] Tavakoli, N. (1991, June 24-28). Information content of images. *Proceedings of the 1991 IEEE International Symposium on Information Theory*, Budapest, Hungary, 264.

[556] Taylor, N. (2002). *Laser: The inventor, the Nobel laureate, and the thirty-year patent war*. New York City, NY, USA: Simon & Schuster.

[557] Thatcher, M. E., and Pingry, D. E. (2007, October). [Software patents] The good, the bad, and the messy. *Communications of the ACM, 50*(10), 47-52.

[558] *圣经（新约全书）（中英对照）*(和合本) [The Holy Bible (New Testament) (Chinese and English bilingualed.) (New King James Version)]. （1997）. Nashville, Tennessee, USA: Thomas Nelson, & Taichung City [台中市], Taiwan (ROC) [中華民國臺灣]: The Gideons International (Taiwan (ROC)) [國際基甸會中華民國總會]. (in Chinese and English languages).

[559] *The official Scrabble players dictionary* (3rd ed.). (2002). Heatherton, Victoria, Australia: Hinkler Books.

[560] *The proof of Parkinson*, [Online]. (1969, July 18). Time Magazine. Available: http://www.time.com/time/magazine/article/0,9171,901078,00.html [2008, July 18].

[561] The Star Online. (2007, June 8). *Exchange rate*, [Online]. Available: http://biz.thestar.com.my/business/exchange.asp [2007, June 8].

[562] The Unicode Consortium, Allen, J. D., et al. (Eds.). (2006, November 19). *The Unicode Standard 5.0*. Boston, Massachusetts, USA: Addison-Wesley Professional.

[563] *The United Nations Convention against transnational organized crime and its protocols*, [Online]. (2004). UNODC (United Nations Office on Drugs and Crime). Available: http://www.unodc.org/unodc/en/treaties/CTOC/index.html;

http://www.unodc.org/documents/treaties/UNTOC/Publications/TOC%20Convention/TOCebook-e.pdf [2008, July 21].

[564] Thomas, P., and Breitzman, A. (2006a, January 12). *The influence of IEEE on key patents*, [Online]. IEEE. Available: http://www.ieee.org/portal/cms_docs_iportals/discover/sub_pages/IEEE_Key_Patents_2006.pdf [2008, September 3].

[565] Thomas, P., and Breitzman, A. (2006b, August 1). A method for identifying hot patents and linking them to government-funded scientific research. *Research Evaluation, 15*(2), 145-152.

[566] Thorpe, J., and van Oorschot, P. C. (2007, August 6-10). Human-seeded attacks and exploiting hot-spots in graphical passwords. *Proceedings of the 16th USENIX Security Symposium 2007 (Security 2007)*, Boston, MA, USA, 103-118.

[567] Tomasi, W. (1998). *Electronic communications systems: Fundamentals through advanced* (3rd ed.). Upper Saddle River, NJ: Prentice-Hall.

[568] Tracy, M., Jansen, W., Scarfone, K., and Butterfield, J. (2007, February). *Guidelines on electronic mail security* (NIST Special Publication 800-45 Version 2). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[569] Tracy, M., Jansen, W., Scarfone, K., and Winograd, T. (2007, September). *Guidelines on securing public web servers* (NIST Special Publication 800-44 Version 2). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[570] Tsou, B. K., Lai, T. B. Y., and Chow, K.-P. (2004, March 22-24). Comparing entropies within the Chinese language. *Proceedings of 1st International Joint Conference on Natural Language Processing (IJCNLP 2004)*, LNCS 3248, Hainan Island, China, 466-475.

[571] Turk, S. A. (2005, Spring). The most important 21st century intellectual property issue. *Chicago-Kent Journal of Intellectual Property, 4*(2), 153-155.

[572] *Twelve steps to developing an effective first draft of your manuscript*, [Online]. (No date). San Francisco Edit. Available: http://www.cis.nctu.edu.tw/~tzeng/First%20Draft.pdf [2008, August 2].

[573] UKCS. (No date a). *Commonly quoted 'alternatives' to registration*, [Online]. The UK Copyright Service (UKCS). Available: http://www.copyrightservice.co.uk/services/alternatives [2007, May 20].

[574] UKCS. (No date b). *The registration service*, [Online]. The UK Copyright Service (UKCS). Available: http://www.copyrightservice.co.uk/register/ [2007, June 7].

[575] United Publishing House［联营出版有限公司］. (2001). *最新汉语大词典［修订版］* [The newest Chinese language big phrase dictionary (revised ed.)]. Seri Kembangan, Selangor, Malaysia: United Publishing House. (in Chinese language).

[576] United Publishing House [联营出版有限公司]. (2002). *新汉语字典* [New Chinese language word/character dictionary]. Seri Kembangan, Selangor, Malaysia: United Publishing House. (in Chinese language).

[577] *UNODC and organized crime*, [Online]. (No date). UNODC (United Nations Office on Drugs and Crime). Available: http://www.unodc.org/unodc/en/organized-crime/index.html [2008, July 23].

[578] U.S. Department of Defense. (1985). *Password management guideline* (Report No. CSC-STD-002-85) Fort George G. Meade, Maryland, USA: DoD Computer Center.

[579] USCO. (No date). *Current fees (July 2007 -)*, [Online]. US Copyright Office (USCO). Available: http://www.copyright.gov/docs/fees.html [2008, May 20].

[580] *Recommendations for consideration by the incoming administration regarding the US Patent and Trademark Office*, [Online]. (No date). US Chamber of Commerce (USCOC). Available: http://dotank.nyls.edu/communitypatent/docs/USCOCRecommendations.pdf [2008, November 10].

[581] Vacca, J. R. (2007, March 16). *Biometric technologies and verification systems*. Oxford, Oxfordshire, UK: Butterworth-Heinemann, p. 280.

[582] Vines, P., and Zobel, J. (1998, October). Compression techniques for Chinese text. *Software: Practice and Experience, 28*(12), 1299-1314.

[583] Wagstaff Jr., S. S. (2003). *Cryptanalysis of number theoretic ciphers*. Boca Raton, FL, USA: Chapman & Hall/CRC Press, 115-117.

[584] Wailgum, T. (2008, September 8). *Password brain teaser: Too many passwords or not enough brain power?*, [Online]. CIO.com. Available: http://www.cio.com/article/print/448241 [2008, September 10].

[585] Walker, P. B. (2007, November 8). *Intangible property transaction and leaseback business method.* USPTO Published Patent Application US2007/0260549, Filing Date: 4 May 2006.

[586] Wang, L. [汪力] (Ed.). (2007, August). *有组织犯罪专题研究* [Research on the topic of organized crime]. Beijing [北京], China [中国]: People's Publishing House [人民出版社]. (in Chinese language).

[587] Wang, S. C., and Kao, S. H. (2001, January 31 – February 2). A new approach for Byzantine agreement. *Proceedings of the 15th International Conference on Information Networking 2001 (ICOIN 2001)*, Beppu City, Oita, Japan, 518-524.

[588] Wang, S. C., and Yan, K. Q. (2000, July 4-7). Reaching fault diagnosis agreement on dual link failure mode. *Proceedings of the 7th International Conference on Parallel and Distributed Systems 2000 (ICPADS 2000)*, Iwate, Japan, 291-298.

[589] Wang, X., Yin, Y. L., and Yu, H. (2005, August 14-18). Finding collisions in the full SHA-1. *Proceedings of the 25th Annual International Cryptology Conference on Advances in Cryptology (Crypto 2005)*, LNCS 3621, Santa Barbara, CA, USA, 17-36.

[590] Wang, X. L. [王学理], and Pei, D, Y. [裴定一]. (2006). *椭圆与超椭圆曲线公钥密码的理论与实现* [Theory and implementation of elliptic curve and hyperelliptic curve cryptography]. Beijing [北京], China [中国]: Science Press [科学出版社]. (in Chinese language).

[591] Wangerin, W. Jr. [沃爾特。溫傑林] (1999a). *The book of God: New Testament* [小說聖經：新約篇]. (Song, B.-Y. [宋碧雲]， Trans.). Taipei City [臺北市], Taiwan (ROC) [中華民國臺灣]: Taiwan Wisdom Publishing Co., Ltd. [臺灣先智出版事業股份有限公司]. (in Chinese language).

[592] Wangerin, W. Jr. [沃爾特。溫傑林] (1999b). *The book of God: Old Testament* [小說聖經：舊約篇]. (Gao, Z.-R. [高志仁]， Trans.). Taipei City [臺北市], Taiwan (ROC) [中華民國臺灣]: Taiwan Wisdom Publishing Co., Ltd. [臺灣先智出版事業股份有限公司]. (in Chinese language).

[593] Weinshall, D., and Kirkpatrick, S. (2004, April 24-29). Passwords you'll never forget, but can't recall. *Proceedings of the Conference on Human Factors in Computing Systems 2004 (CHI 2004)*, Vienna, Austria, 1399-1402.

[594] Weiss, R. P., and South, N. (Eds.). (1998). *Comparing prison systems: Toward a comparative and international penology*. Amsterdam, The Netherlands: OPA (Overseas Publishers Association) under the license from GIB (Gordon and Breach Publishers).

[595] Wiener, M. J. (1990, May). Cryptanalysis of short RSA secret exponents. *IEEE Transactions on Information Theory, 36*(3), 553-558.

[596] Wikipedia Contributors. (2006, March 14). *Statutory damages*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Statutory_damages&oldid=43705331 [2007, May 23].

[597] Wikipedia Contributors. (2007a, January 13). *Four corner method*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Four_corner_method&oldid=100425004 [2007, June 2].

[598] Wikipedia Contributors. (2007b, February 10). *Password policy*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Password_policy&oldid=106996972 [2007, June 2].

[599] Wikipedia Contributors. (2007c, February 26). *Chinese input methods for computers*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Chinese_input_methods_for_computers&oldid=111110951 [2007, June 2].

[600] Wikipedia Contributors. (2007d, March 17). *Chinese character encoding*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Chinese_character_encoding&oldid=115862258 [2007, June 2].

[601] Wikipedia Contributors. (2007e, March 31a). *Statutory damages for copyright infringement*, [Online]. Wikipedia The Free Encyclopedia. Available:

http://en.wikipedia.org/w/index.php?title=Statutory_damages_for_copyright_infringement&oldid=119234996 [2007, May 23].

[602] Wikipedia Contributors. (2007f, March 31b). *Typeface*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Typeface&oldid=119171254 [2007, June 2].

[603] Wikipedia Contributors. (2007g, April 5). *ASCII art*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=ASCII_art&oldid=120415207 [2007, June 2].

[604] Wikipedia Contributors. (2007h, April 5). *Chinese character*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Chinese_character&oldid=120519979 [2007, June 2].

[605] Wikipedia Contributors. (2007i, June 6). *Berne Convention for the Protection of Literary and Artistic Works*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Berne_Convention_for_the_Protection_of_Literary_and_Artistic_Works&oldid=136381222 [2007, June 7].

[606] Wikipedia Contributors. (2007j, September 13). *First to file and first to invent*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=First_to_file_and_first_to_invent&oldid=157568486 [2007, October 22].

[607] Wikipedia Contributors. (2007k, September 20). *TIA/EIA-568-B*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=TIA/EIA-568-B&oldid=159241053 [2008, January 24].

[608] Wikipedia Contributors. (2007l, September 26). *Registered jack*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Registered_jack&oldid=160450633 [2008, January 24].

[609] Wikipedia Contributors. (2007m, September 27). *Modular connector*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Modular_connector&oldid=160734170 [2008, January 24].

[610] Wikipedia Contributors. (2007n, October 4). *Gordon Gould*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Gordon_Gould&oldid=162298481 [2007, October 10].

[611] Wikipedia Contributors. (2007o, October 6). *Power over Ethernet*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Power_over_Ethernet&oldid=162586392 [2008, January 24].

[612] Wikipedia Contributors. (2007p, October 7). *Category 5 cable*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Category_5_cable&oldid=162908384 [2008, January 24].

[613] Wikipedia Contributors. (2007q, October 7). *Hacker (computer security)*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Hacker_%28computer_security%29&oldid=162953161 [2008, January 24].

[614] Wikipedia Contributors. (2007r, December 2). *Penology*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Penology&oldid=175222217 [2008, July 23].

[615] Wikipedia Contributors. (2008a, March 6). *Monolingualism*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Monolingualism&oldid=196315794 [2008, July 23].

[616] Wikipedia Contributors. (2008b, March 8). *Patent Reform Act of 2005*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Patent_Reform_Act_of_2005&oldid=197359847 [2008, May 20].

[617] Wikipedia Contributors. (2008c, March 10). *Linguistic demography*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Linguistic_demography&oldid=197188475 [2008, July 23].

[618] Wikipedia Contributors. (2008d, April 26). *Ocean Tomo LLC*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Ocean_Tomo_LLC&oldid=208361904 [2008, August 21].

[619] Wikipedia Contributors. (2008e, May 5). *Patent Reform Act of 2007*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Patent_Reform_Act_of_2007&oldid=210421848 [2008, May 20].

[620] Wikipedia Contributors. (2008f, May 7). *Redundancy (information theory)*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Redundancy_%28information_theory%29&oldid=210871144 [2008, May 20].

[621] Wikipedia Contributors. (2008g, May 15). *CamelCase*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=CamelCase&oldid=212680140 [2008, May 20].

[622] Wikipedia Contributors. (2008h, May 30). *Theoretical linguistics*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Theoretical_linguistics&oldid=216055905 [2008, July 23].

[623] Wikipedia Contributors. (2008i, June 21). *Coefficient of inefficiency*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Coefficient_of_Inefficiency&oldid=220969534 [2008, July 23].

[624] Wikipedia Contributors. (2008j, June 21). *Language policy*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Language_policy&oldid=220714548 [2008, July 23].

[625] Wikipedia Contributors. (2008k, June 22). *Petname*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Petname&oldid=220857363 [2008, July 16].

[626] Wikipedia Contributors. (2008l, July 3). *Short-term memory*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Short-term_memory&oldid=223406253 [2008, July 23].

[627] Wikipedia Contributors. (2008m, July 7). *List of writing systems*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_writing_systems&oldid=224239367 [2008, September 1].

[628] Wikipedia Contributors. (2008n, July 8). *Timeline of computer security hacker history*, [Online]. Wikipedia The Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Timeline_of_computer_security_hacker_history&oldid=224341103 [2008, July 16].

[629] Wikipedia Contributors. (2008o, July 9). *Cryptographic hash function*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Cryptographic_hash_function&oldid=224604390 [2008, July 17].

[630] Wikipedia Contributors. (2008p, July 9). *List of multilingual countries and regions*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_multilingual_countries_and_regions&oldid=224607350 [2008, July 23].

[631] Wikipedia Contributors. (2008q, July 11). *Cronyism*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Cronyism&oldid=225028349 [2008, July 23].

[632] Wikipedia Contributors. (2008r, July 11). *MD5*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=MD5&oldid=225041244 [2008, July 16].

[633] Wikipedia Contributors. (2008s, July 12). *Criminology*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Criminology&oldid=225212774 [2008, July 23].

[634] Wikipedia Contributors. (2008t, July 13). *Dissertation*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Dissertation&oldid=225330900 [2008, July 14].

[635] Wikipedia Contributors. (2008u, July 15). *Applied linguistics*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Applied_linguistics&oldid=225753837 [2008, July 23].

[636] Wikipedia Contributors. (2008v, July 15). *Moore's Law*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Moore%27s_law&oldid=225829043 [2008, July 16].

[637] Wikipedia Contributors. (2008w, July 15). *SHA hash functions*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=SHA_hash_functions&oldid=225763377 [2008, July 16].

[638] Wikipedia Contributors. (2008x, July 17). *Parkinson's Law*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Parkinson%27s_law&oldid=226138504 [2008, July 23].

[639] Wikipedia Contributors. (2008y, July 19). *Corpus linguistics*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Corpus_linguistics&oldid=226567048 [2008, July 23].

[640] Wikipedia Contributors. (2008z, July 19). *Text corpus*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Text_corpus&oldid=226566944 [2008, July 23].

[641] Wikipedia Contributors. (2008aa, July 21). *First language*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=First_language&oldid=227072714 [2008, July 23].

[642] Wikipedia Contributors. (2008ab, July 21). *Second language*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Second_language&oldid=226939389 [2008, July 23].

[643] Wikipedia Contributors. (2008ac, July 22). *Ethnologue list of most spoken languages*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Ethnologue_list_of_most_spoken_languages&oldid=227286397 [2008, July 23].

[644] Wikipedia Contributors. (2008ad, July 22). *List of languages by number of native speakers*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_languages_by_number_of_native_speakers&oldid=227300820 [2008, July 23].

[645] Wikipedia Contributors. (2008ae, July 22). *Multilingualism*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Multilingualism&oldid=227240877 [2008, July 23].

[646] Wikipedia Contributors. (2008af, July 22). *Organized crime*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Organized_crime&oldid=227261189 [2008, July 23].

[647] Wikipedia Contributors. (2008ag, July 22). *SMS language*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=SMS_language&oldid=227293265 [2008, July 23].

[648] Wikipedia Contributors. (2008ah, July 22). *Text messaging*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Text_messaging&oldid=227309098 [2008, July 23].

[649] Wikipedia Contributors. (2008ai, July 23). *Language*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Language&oldid=227357180 [2008, July 23].

[650] Wikipedia Contributors. (2008aj, July 23). *Sans-serif*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Sans-serif&oldid=227325014 [2008, July 23].

[651] Wikipedia Contributors. (2008ak, July 23). *Serif*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Serif&oldid=227341547 [2008, July 23].

[652] Wikipedia Contributors. (2008al, August 9). *Digital identity*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Digital_identity&oldid=230770248 [2008, August 25].

[653] Wikipedia Contributors. (2008am, August 11). *OpenID*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=OpenID&oldid=231207285 [2008, August 16].

[654] Wikipedia Contributors. (2008an, August 12). *Alcatel-Lucent v. Microsoft*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Alcatel-Lucent_v._Microsoft&oldid=231503895 [2008, September 17].

[655] Wikipedia Contributors. (2008ao, August 15). *Inventor (patent)*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Inventor_(patent)&oldid=232136205 [2008, August 16].

[656] Wikipedia Contributors. (2008ap, August 19). *Information Card*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Information_Card&oldid=233004113 [2008, August 23].

[657] Wikipedia Contributors. (2008aq, August 27). *Writing system*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Writing_system&oldid=234534887 [2008, September 1].

[658] Wikipedia Contributors. (2008ar, September 8). *List of languages by total number of speakers*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_languages_by_total_number_of_speakers&oldid=237048814 [2008, October 4].

[659] Wikipedia Contributors. (2008as, September 18). *Long-term memory*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Long-term_memory&oldid=239249968 [2008, October 4].

[660] Wikipedia Contributors. (2008at, September 23). *CJK Unified Ideographs*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=CJK_Unified_Ideographs&oldid=240390843 [2008, October 16].

[661] Wikipedia Contributors. (2008au, September 27). *Languages by speakers*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Languages_by_speakers&oldid=241385108 [2008, October 4].

[662] Wikipedia Contributors. (2008av, October 3). *Multilingual education*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Multilingual_Education&oldid=242699632 [2008, October 4].

[663] Wikipedia Contributors. (2008aw, October 4). *Working memory*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Working_memory&oldid=242892307 [2008, October 4].

[664] Wikipedia Contributors. (2008ax, October 9). *Spy satellite*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Spy_satellite&oldid=244048302 [2008, October 11].

[665] Wikipedia Contributors. (2008ay, October 10). *GeoEye*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=GeoEye&oldid=244443527 [2008, October 11].

[666] Wikipedia Contributors. (2008az, October 10). *MIFARE*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=MIFARE&oldid=244386929 [2008, October 16].

[667] Wikipedia Contributors. (2008ba, October 12). *Unity of invention*, [Online]. Wikipedia the Free Encyclopedia. Available:

http://en.wikipedia.org/w/index.php?title=Unity_of_invention&oldid=244841923 [2008, October 15].

[668] Wikipedia Contributors. (2008bb, October 14). *Age of Discovery*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Age_of_Discovery&oldid=245322423 [2008, October 16].

[669] Wikipedia Contributors. (2008bc, October 15). *British Empire*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=British_Empire&oldid=245420209 [2008, October 16].

[670] Wikipedia Contributors. (2008bd, October 23). *Botnet*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Botnet&oldid=247128093 [2008, October 23].

[671] Wikipedia Contributors. (2008be, October 23). *Disk cloning*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Disk_cloning&oldid=247128872 [2008, October 23].

[672] Wikipedia Contributors. (2008bf, October 23). *Disk image*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Disk_image&oldid=247128569 [2008, October 23].

[673] Wikipedia Contributors. (2008bg, October 23). *H-index*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=H-index&oldid=247127081 [2008, October 23].

[674] Wikipedia Contributors. (2008bh, October 23). *Identity theft*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Identity_theft&oldid=246961579 [2008, October 23].

[675] Wikipedia Contributors. (2008bi, October 23). *Internet bot*, [Online]. Wikipedia the Free Encyclopedia. Available:

http://en.wikipedia.org/w/index.php?title=Internet_bot&oldid=247128048 [2008, October 23].

[676] Wikipedia Contributors. (2008bj, October 23). *Zombie computer*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Zombie_computer&oldid=247127637 [2008, October 23].

[677] Wikipedia Contributors. (2008bk, October 25). *Extrasensory perception*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Extrasensory_perception&oldid=247523584 [2008, October 25].

[678] Wikipedia Contributors. (2008bl, October 25). *Psi (parapsychology)*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Psi_(parapsychology)&oldid=247524054 [2008, October 25].

[679] Wikipedia Contributors. (2008bm, October 25). *Parapsychology*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Parapsychology&oldid=247525788 [2008, October 25].

[680] Wikipedia Contributors. (2008bn, October 25). *Parapsychological Association*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Parapsychological_Association&oldid=247524424 [2008, October 25].

[681] Wikipedia Contributors. (2008bo, October 25). *Joseph Banks Rhine*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=Joseph_Banks_Rhine&oldid=247524655 [2008, October 25].

[682] Wikipedia Contributors. (2008bp, October 25). *List of psychic abilities*, [Online]. Wikipedia the Free Encyclopedia. Available: http://en.wikipedia.org/w/index.php?title=List_of_psychic_abilities&oldid=247524868 [2008, October 25].

[683] Williams, A. (2006, February 16). *System and method for patent evaluation using artificial intelligence*. USPTO Published Patent Application US2006/0036632, Filing Date: 11 August 2004.

[684] Williams, L. C. (2002). *A discussion of the importance of key length in symmetric and asymmetric cryptography*, [Online]. Bethesda, Maryland, USA: SANS Institute. Available: http://www.giac.org/practical/gsec/Lorraine_Williams_GSEC.pdf [2008, May 17].

[685] Wilson, A. L. (2008, June 11). *Microsoft's CardSpace attacked by researchers*, [Online]. CIO.com. Available: http://www.cio.com/article/print/391813 [2008, September 10].

[686] Wilson, C., Grother, P., and Chandramouli, R. (2007, January). *Biometric data specification for personal identity verification* (NIST Special Publication 800-76-1). Gaithersburg, MD, USA: CSRC (Computer Security Resource Center), NIST.

[687] *Windows Live ID* (aka Microsoft Passport Network), [Online]. (No date). Available: http://www.passport.net [2008, July 16].

[688] Wingate, P., and Sinden, D. (Eds.). (2002, September). *Totltxt: The big Book of little text messages*. New York, NY, USA: Book Sales, Inc. (URL: http://www.booksalesusa.com).

[689] WIPO. (No date). Pre-established damages / statutory damages. In *Which kind of damages are available in IP disputes?*, [Online]. WIPO. Available: http://www.wipo.int/enforcement/en/faq/judiciary/faq08.html#pre [2007, May 23].

[690] WIPO. (1979, September 28). *Berne Convention for the Protection of Literary and Artistic Works*, [Online]. WIPO. Available: http://www.wipo.int/treaties/en/ip/berne/trtdocs_wo001.html [2007, May 20].

[691] Witty, R. J. (2001, January 30). *Best practices for managing PINs and passwords* (Tech. Rep. No. Gartner QA-12-8664). Stamford, CT, USA: Gartner, Inc.

[692] Wolfgang, R. B., Podilchuk, C. I., and Delp, E. J. (1999, July). Perceptual watermarks for digital images and video. *Proceedings of the IEEE, 87*(7), 1108-1126.

[693] Wrixon, F. B. (1998). *Codes ciphers & other cryptic & clandestine communication: Making & breaking secret messages from hieroglyphs to the Internet.* New York City, NY, USA: Könemann, 298-303, 671-672.

[694] Wu, T. J. (2003, March 25). *System and method for securely logging onto a remotely located computer*. USPTO Issued Patent US6539479, Filing Date: 14 July 1998, Issue Date: 25 March 2003.

[695] Wu, X., Cheng, S., and Xiong, Z. (2001, March). On packetization of embedded multimedia bitstreams. *IEEE Transactions on Multimedia, 3*(1), 132-140.

[696] Xu, F. C. [徐梵澄]. (1984, January). *五十奥义书* [Fifty Upanishads]. Beijing, China: China Social Sciences Press [中国社会科学出版社]. (in Chinese language).

[697] Xu, S. [许慎]. (2001). *说文解字* [Talking on language and explaining the words]. Hong Kong SAR, China: Chung Hwa Book [中华书局]. (in Chinese language).

[698] Yan, J., Blackwell, A., Anderson, R., and Grant, A. (2004, September-October). Password memorability and security: Empirical results. *IEEE Security and Privacy Magazine, 2*(5), 25-31.

[699] Yan, K. Q., and Chin, Y. H. (1988, August). An optimal solution for consensus problem in an unreliable communication system. *Proceedings of the International Conference on Parallel Processing (ICPP 1988)*, The Pennsylvania State University, University Park, PA, USA, Vol. 1: Architecture, 388-391.

[700] Yan, K. Q., Chin, C. H., and Wang, S. C. (1992, June). Optimal agreement protocol in malicious faulty processors and faulty links. *IEEE Transactions on Knowledge and Data Engineering, 4*(3), 266-280.

[701] Yan, K. Q., and Wang, S. C. (2005a, February 25). Reaching fault diagnosis agreement on an unreliable general network. *Information Sciences, 170*(2-4), 397-407.

[702] Yan, K. Q., and Wang, S. C. (2005b, July). Grouping Byzantine agreement. *Computer Standards & Interfaces, 28*(1), 75-92.

[703] Yan, K. Q., Wang, S. C., and Chin, Y. H. (1999, March 12). Consensus under unreliable transmission. *Information Processing Letters, 69*(5), 243-248.

[704] Yannakoudakis, E. J., and Angelidakis, G. (1988, November). An insight into the entropy and redundancy of the English dictionary. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 10*(6), 960-970.

[705] Yee, K. P., and Sitaker, K. (2006, July 12-14). Passpet: Convenient password management and phishing protection. *Proceedings of Symposium on Usable, Privacy and Security (SOUPS2006)*, Pittsburgh, PA, USA, 32-43.

[706] Zhang, H., Xu, B., and Huang, T. (2000, October 14-16). Statistical analysis of chinese language and language modeling based on huge text corpora. *Proceedings of the 3rd International Conference on Advances in Multimodal Interfaces (ICMI 2000)*, LNCS 1948, Beijing, China, 279-286.

[707] Zheng, Y. (1997, August 17-21). Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption). *Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO '97)*, LNCS 1294, Santa Barbara, CA, USA, 165-179.

[708] Zhu, Y. F. [祝跃飞], and Zhang, Y. J. [张亚娟]. (2006, October). *椭圆曲线公钥密码导引* [Guide to elliptic curve cryptography]. Beijing [北京], China [中国]: Science Press [科学出版社]. (in Chinese language).

ACRONYMS

2TDEA	2-Key Triple Data Encryption Algorithm
3TDEA	3-Key Triple Data Encryption Algorithm
2TDES	2-Key Triple Data Encryption Standard
3TDES	3-Key Triple Data Encryption Standard
ACM	Association for Computing Machinery
AES	Advanced Encryption Standard
AIPO/OAPI	African Intellectual Property Organization (Organisation Africaine de la Propriété Intellectuelle)
ANN	Artificial Neural Network
ANN Based BAP	Artificial Neural Network Based Byzantine Agreement Protocol
APA	American Psychological Association
APWG	Anti-Phishing Working Group
ARIPO	African Regional Industrial Property Organization
AUTM	Association of University Technology Managers
BAP	Byzantine Agreement Protocol
BAP-ANN	Byzantine Agreement Protocol with Artificial Neural Network
BGP	Byzantine Generals Problem
BTIRDM	Budapest Treaty on the International Recognition of the Deposit of Microorganisms for the Purposes of Patent Procedure
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
CII	Computer-Implemented Invention
CIS	Cryptography & Information Security
CLJ	Crime, Law, and Justice
CLPP	Chinese Language Passphrase
CLPW	Chinese Language Password
CM	Communication Management
CO	Central Office
CSPRNG	Cryptographically Secure Pseudo-Random Number Generator
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DIL/DIP	Dual In-Line Package
DNS	Domain Name System
DNSSEC	Domain Name System Security Extensions
DSA	Digital Signature Algorithm
DSS	Digital Signature Standard
EAPO	Eurasian Patent Organization

ECC	Elliptic Curve Cryptography
EMAIL	Electronic Mail
EPO	European Patent Office
ESP	Extra-Sensory Perception
EU	European Union
FAR	False Acceptance Rate
FCN	Fully Connected Network
FFC	Finite Field Cryptography
FOREX	Foreign Exchange
FRR	False Rejection Rate
FTP	File Transfer Protocol
FTPS	FTP over SSL
GCC	Gulf Cooperation Council
GCCPO	Gulf Cooperation Council Patent Office
HDD	Hard Disk Drive
HMAC	Keyed-Hash Message Authentication Code
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP over SSL
IACR	International Association for Cryptologic Research
IATUL	International Association of Technological University Libraries
IDC	Identity-Based Cryptography
IEEE	Institute of Electrical and Electronics Engineers, Inc.
IEICE	The Institute of Electronics, Information and Communication Engineers （電子情報通信学会）
IETF	Internet Engineering Task Force
IFC	Integer Factorization Cryptography
IIPA	International Intellectual Property Alliance
ILBS	International Law Book Services
IM	Instant Messaging
IMAP4	Internet Message Access Protocol version 4
IP	Intellectual Property
IPOS	Intellectual Property Office of Singapore
IPR	Intellectual Property Right
IRC	Internet Relay Chat
ITRC	Identity Theft Resource Center
JPO	Japan Patent Office （日本経済産業省特許庁）
MAC	Message Authentication Code
MCMC	Malaysian Communications and Multimedia Commission
MDC	Multimedia Development Corporation Sdn Bhd

MDeC	Multimedia Development Corporation Sdn Bhd
MePKC	Memorizable Public-Key Cryptography / Memorizable Public-Key Cryptosystem
MIME	Multipurpose Internet Mail Extensions
MITM	Man In The Middle
MoPKC	Mobile Public-Key Cryptography
MTSO	Mobile Telephone Switching Office
MY	Malaysia
MyIPO	Intellectual Property Corporation of Malaysia (Perbadanan Harta Intelek Malaysia)
NBER	National Bureau of Economic Research
NIST	National Institute of Standards and Technology
OAPI/AIPO	Organisation Africaine de la Propriété Intellectuelle (African Intellectual Property Organization)
OSCAR	Open System for CommunicAtion in Realtime (AOL Instant Messenger Protocol for ICQ and AIM)
OTP	One-Time Password
PAKE	Password-Authenitcated Key Exchange
PCPIP	Paris Convention for the Protection of Industrial Property
PCT	Patent Cooperation Treaty
PGP	Pretty Good Privacy
Ph.D.	Doctor of Philosophy
PKC	Public-Key Cryptography
PKC	Public-Key Cryptosystem
PLT	Patent Law Treaty
PNAS	Proceedings of the National Academy of Sciences
POP3	Post Office Protocol version 3
P.R.C.	People’s Republic of China （中华人民共和国）
PRNG	Pseudo-Random Number Generator
PSTN	Public Switched Telephone Network
RFC	Request for Comments
RFID	Radio Frequency Identification
Rlogin	Remote Login in UNIX Systems
RNG	Random Number Generator
R.O.C.	Republic of China （中華民國）
RSA	Rivest-Shamir-Adleman Public-Key Cryptography
S/MIME	Secure / Multipurpose Internet Mail Extensions
SATA	Serial Advanced Technology Attachment
SD	Statutory Declaration
SFTP	Secure FTP over SSH
SHA	Secure Hash Algorithm

SHS	Secure Hash Standard
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
SIPO	State Intellectual Property of the P.R.C. （中华人民共和国国家知识产权局）
SMS	Short Message Service
SNMP	Simple Network Management Protocol
SPC	Strasbourg Patent Convention
SPEKE	Simple Password Exponential Key Exchange
SPLT	Substantive Patent Law Treaty
SRP-6	Secure Remote Password Protocol version 6
SSH	Secure Shell
SSL	Secure Sockets Layer
TAC	Transaction Authorisation Code or Transaction Authentication Code
TAP	Transaction Authorization Pin
TELNET	Telecommunication Network
TIPO	The Intellectual Property Office of Ministry of Economic Affairs, R.O.C. （中華民國經濟部智慧財產局）
TLS	Transport Layer Security
TRIPS	Agreement on Trade Related Aspects of Intellectual Property Rights
TSIG	Transaction SIGnature Protocol
TSA	Timestamping Authority
TSP	Time-Stamp Protocol
TTP	Trusted Third Party
UI	Utility Innovation
UK	United Kingdom
UKCS	UK Copyright Service
UN	United Nations
UNESCOBKK	UNESCO Bangkok
UNODC	United Nations Office on Drugs and Crime
US	United States
USA	United States of America
USB	Universal Serial Bus
USCO	US Copyright Office
USCOC	US Chamber of Commerce
USPTO	US Patent and Trademark Office
MSVS	Microsoft Visual Studio
WIPO	World Intellectual Property Organization
WTO	World Trade Organization