Proceedings of the 4th International Cryptology and Information Security Conference 2014

24th – 26th June 2014
Putrajaya, Malaysia

Published by
Institute for Mathematical Research (INSPEM)
Universiti Putra Malaysia
43400 UPM Serdang
Selangor Darul Ehsan

First Print 2014

Perpustakaan Negara Malaysia Cataloguing – in – Publication Data

Proceedings of the 4th International Cryptology and Information Security
(4th.: 2014: Putrajaya)
CONFERENCE PROCEEDINGS CRYPTOLOGY 2014: Proceedings of The 4th
International Cryptology and Information Security Conference 2014 24th-26th June
2014, Putrajaya, MALAYSIA / Editors: Ainuddin Wahid Abdul Wahab,
Ang Min Huey, Chin Ji Jian, Faridah Yunos, Goi Bok Min, Hailiza Kamarul Haili,
Heng Swee Huay, Moesfa Soeheila Mohamad, Mohamad Rushdan Md. Said,
Muhammad Reza Z'aba, Muhammad Rezal Kamel Ariffin, Rabiah Ahmad,
Shekh Faisal Abdul Latip, Yanbin Pan.
ISBN 9789834406943
1. Cryptography- - Malaysia - - Congresses.
2. Computer security - - Malaysia- - Congresses.
I. Ainuddin Wahid Abdul Wahab. II. Ang, Min Huey. III. Chin, Ji-Jian.
IV. Faridah Yunos. V. Goi, Bok Min. VI. Hailiza Kamarul Haili
VII. Heng, Swee-Huay. VIII. Moesfa Soeheila Mohamad.
IX. Mohamad Rushdan Md. Said. X. Muhammad Reza Za'ba.
XI. Muhammad Rezal Kamel Ariffin. XII. Rabiah Ahmad.
XIII. Shekh Faisal Abdul Latip. XIV. Pan, Yanbin. XV. Title.
005.8072

OPENING REMARKS

First and foremost, I would like to thank the Malaysian Society for Cryptology Research (MSCR) in collaboration with CyberSecurity Malaysia together with universities: UPM, USM, MMU and UiTM for its continuous efforts and commitment to host this premier event, the International Cryptology and Information Security Conference for the fourth time. This bi – annual conference series which started in 2008 have been organized and hosted at several locations in Malaysia, beginning from Kuala Lumpur to Melaka, and then on to Langkawi. This year Putrajaya has been chosen as the venue.

Cryptology is an area of study and research which has numerous applications especially in the area of information and communication technology. The **4th International Cryptology and Information Security Conference 2014 (Cryptology2014)** is one of the many steps towards enhancing and realizing research and applications of cryptology in Malaysia through research collaboration and discussion with research counterparts from the international arena. This is an open forum, provided for contributions which cover research on the theoretical foundations, applications and any related issues in cryptology, information security and other underlying technologies. Pertaining to these challenges MSCR was formed in 2007. Since then, collaborative efforts between Malaysian cryptography centers of excellence has been galvanised and are no longer working in silo. The ministry greatly appreciates these efforts, where research activities do not only generate significant impact towards the improvement of the body of cryptographic knowledge but also related to society needs, especially in the area of information security.

Once again MSCR, CyberSecurity Malaysia and the above mentioned local institutions of higher learning has put in a huge effort to bring together distinguished researchers from various countries as speakers and participants to share knowledge and novel ideas. We congratulate MSCR and its partners for tirelessly organizing beneficial and meaningful events that contribute to further strengthen Research and Development (R&D) in Malaysia. I would also like to take this opportunity to give special thanks to the conference corporate supporters namely Digicert and TM Applied Business in supporting this event. As we know, both are Certificate Authorities (CA) in Malaysia and have direct application of cryptography in their business products. I urge researchers attending this event to take this opportunity to enquire on various industrial related issues regarding cryptography. It is hoped that both will continue to support this conference in the future.

Our keynote and invited speakers are highly distinguished researchers and are from different parts of the world. They bring with them invaluable knowledge and experience. We hope that participants will take this opportunity to interact and benefit from this conference. We would like to express our sincere appreciation and gratitude to the speakers for accepting our invitation despite their busy schedule and contributing to this meaningful event.

I also note that, the participation of researchers from various cryptographic disciplines is impressive and this signifies the interdisciplinary nature of the topic of the conference. It is hoped that the network among researchers and institutions will grow and new research collaboration can be built. We hope that through these sorts of events and activities we can overcome the shortage of expertise in this area of research.

I am confident that this Conference would serve as a platform to further discuss the research collaboration and afford brilliant ideas in the cryptology and information security. To all our participants, I wish you a fruitful and productive Conference. To our international participants, I wish you would take some time to travel in this country, especially in Putrajaya to experience and enjoy the rich Malaysian environment and warm hospitality.

Thank you.

DATUK DR. ABU BAKAR BIN MOHAMAD DIAH
Deputy Minister,
Ministry of Science Technology and Innovation, MALAYSIA

Welcoming Notes

I am most pleased to welcome speakers from various countries across the world to the 4th International Cryptology and Information Security Conference 2014 (Cryptology2014). It is our hope that participants will grab the advantage and experience in this intellectual discussion and social interaction during Cryptology2014 in order to further understand cryptology and its applications.

Cryptography is an area of study and research that has numerous applications especially in the area technology and communication technology. In this respect, the conference will provide a speech, which will restrict cover several current topics related to the local research in the cryptology. It is also aimed at promoting and encouraging the exchange of ideas and at the same time identifying areas of collaborative research between local and foreign researchers.

Information security has never become as important in daily lives as we are experiencing today. We are now on the brink of experiencing cryptography and its deployment in every corner of our day to day experiences. Thus, research in this area has become so important – that without continuous research in the area one would not be for certain the capabilities of ever growing adversaries globally.

In this conference, we have organized 5 keynote speeches to be delivered by renowned researchers on their respective areas for the benefit of the participants. Also, 27 papers are scheduled to be presented encompassing various areas of cryptology such as theoretical foundations, applications, network security and other underlying technologies in this interesting mathematical field. I hope this conference will be one more step closer towards realizing research in theory and applications of cryptology in Malaysia.

It goes without saying that a conference of this kind could not have been held without the committed efforts of various individuals. I would like to take this opportunity to congratulate and thank everyone involved for their excellent work and in particular to Universiti Sains Malaysia (USM), Malaysian Multimedia University (MMU), Universti Teknologi MARA (UiTM), Institute for Mathematical Research (INSPEM) of Universiti Putra Malaysia (UPM) and CyberSecurity Malaysia for taking up the challenge of organizing this conference. I would also like to thank our corporate partners – Digicert and TM Applied Business, who have helped to realize this event. I wish all participants to have an enjoyable and beneficial event. With that, I once again thank all presenters and participants in making this conference possible and a successful event.

Thank you.

PROF. DATO' DR HJ KAMEL ARIFFIN BIN HJ MOHD ATAN
Director,
Institute for Mathematical Research, UPM
President,
Malaysian Society for Cryptology Research

Editorial Preface

Since the time of Julius Caesar and possibly up until the Greek era, cryptography (a word that is derived from the Greek term "cryptos") has been an integral tool for organizations (and indeed for individuals too) to ensure information that is intended only for authorized recipients remain confidential only to this set of people. Cryptography had far reaching implications for organizations in the event information leakage occurred. Often referred to as the "last bastion of defence" – after all other mechanisms had been overcome by an adversary, encrypted information would still remain useless to the attacker (i.e. that is, under the usual security assumptions). Nevertheless, this simple fact has remained oblivious to the practitioners of information security – omitting cryptographic mechanism for data being transferred and also during storage.

Fast forward to World War 2, – the war between cryptographic and cryptanalytic techniques. While the Germans were efficiently transferring information via the Enigma encryption machine, the Allies in Bletchley Park, England were busy intercepting these ciphered information being transmitted via telegraph by the Germans. Leading mathematicians, linguists, engineers etc. were all working to cryptanalyze these ciphers in the most information way. It is here that the first electrical machine (i.e. the "bomba") was born – and revolutionized computing. Post World War 2 saw the emergence of the "computer". Every organization that had to process data had to acquire a computer so as not to be left behind by their competitor. The banking sector advanced on a global scale due to the invention of the computer. Techniques to secure information among the headquarters of these banks had to be developed. Encryption procedures using the same key (i.e. symmetric encryption) played this role in the early days. Then came the unthinkable problem – computers were being deployed almost everywhere. How is it possible to deploy cryptographic keys in secure manner so that symmetric encryption could take place? Thus, leading to the so-called "key distribution" problem. It was not until 1975, when Diffie and Hellman provided us with a secure key exchange method – and in 1976 when Rivest, Shamir and Adleman with the "asymmetric encryption" scheme (i.e. to encrypt using key e and decrypt uing key d, where $e \neq d$). Since then, cryptographic procedures evolved, not only playing the role of ensuring confidentiality of data, but also to ensure integrity and authenticity of data. It is also able to ensure that non-repudiating of data does not occur.

Mechanisms to transfer and store data has changed of the centuries and more so every 5 years (in this modern age). Cryptography that has long existed before mechanisms changed from manual – telegraphic – electrical – electronic (WAN/LAN/internet) – wired until wireless procedures, has to be properly deployed in order to maintain a high level of security confidence among the stakeholders of a certain organization. The concept of securing information via encryption procedures has to be properly understood in order to avoid a null intersection to occur between cryptography and computer security practitioners. This scenario would not be to the best interest for stakeholders. As a "friendly" reminder, this scenario could already been seen in other discipline of knowledge where the "minuting" ("minute-ting") of knowledge has forced the original body of knowledge to look as though it is independent and disassociated. Ever since mass usage of computers became a reality, computer security issues have never been this complicated. However, as the human race advances so will ingenious ideas emerge to overcome challenges.

It is hoped that Cryptology2014 will not only provide a platform for every participant to exchange ideas in their respective fields, but also to exchange new ideas on a broader scale for the advancement of the field of cryptology and computer security. The organizing committee hopes every participant will have an enjoyable and beneficial conference.

Thank you.

Editorial Board,
Cryptology2014

CONTENTS

Board of Editors

Wan Zariman Omar
Zahari Mahad

Associate Editors Amir Hamzah Abd Ghafar
Nor Azlida Aminudin

Cover Design Zahari Mahad

Recent Attacks on the RSA Cryptosystem

Abderrahmane Nitaj
LMNO, Université de Caen Basse Normandie, France
Email: abderrahmane.nitaj@unicaen.fr

EXTENDED ABSTRACT.

The RSA public-key cryptosystem was invented by Rivest, Shamir, and Adleman in 1978. Since then, the RSA cryptosystem has been the best known and most widely accepted public key cryptosystem and, consequently, has been analyzed for vulnerability by many researchers. We review the latest attacks on the RSA cryptosystem including algebraic and side-channel attacks.

The RSA cryptosystem invented by (Rivest, Shamir and Adleman, 1978) is the most widely used public key cryptosystem. It is deployed in many popular commercial systems and applications for providing privacy and ensuring authenticity of digital data.

In the RSA cryptosystem, the main parameters are the public key (N, e) and the private key (N, d) where $N = pq$ is the product of two large primes of the same bit-size, and e and d are two positive integers satisfying $ed \equiv 1(\text{mod } \phi(N))$ where $\phi(N) = (p-1)(q-1)$ is Euler's totient function. The value N is called the RSA modulus, the value e is called the public exponent and d is called the private exponent. To encrypt a message using an RSA public key (N, e), one first transforms the message to obtain a positive integer m with $m < N$. The encrypted text is then computed as $c \equiv m^e(\text{mod } N)$. To decrypt an encrypted message c using the private key (N, d), one simply compute $m \equiv c^d(\text{mod } N)$. An encrypted message c can be digitally signed by applying the decryption operation $s \equiv c^d(\text{mod } N)$. The digital signature can then be verified by applying the encryption operation $c \equiv s^e(\text{mod } N)$.

Since its invention an due to its popularity, the RSA was subject to an extensive cryptanalysis. The RSA cryptosystem has been analyzed for vulnerability by many researchers. There are hundreds of attacks on RSA in the literature, but none of them is devastating. The most popular attack on RSA was found by (Wiener, 1990) using algebraic tools. Wiener's attack was slightly improved by (Boneh and Durfee, 1999) using more sophisticated tools based on Coppersmith's method. Very recently, (Genkin, Shamir and Tromer, 2013) described a side channel attack that enables to factor a 4096-bit RSA modulus by listening to the processor sounds, which makes the attack a very academic approach.

Our goal is to survey the recent attacks on RSA. This includes integer factorization attacks, chosen ciphertext attacks, timing attacks, fault injection attacks, acoustic attacks and algebraic attacks.

- **Integer factorization attacks**

The obvious way to attack RSA is factoring the modulus N. Using the factorization $N = pq$, it is easy to find $\phi(N)$ and then to break the system. Currently, the best factoring algorithm is the general number field sieve (GNFS). This algorithm was used in 2009 to factor a 768-bit RSA modulus. Its running time is $\exp\left((c + o(1))\log N)^{1/3} (\log\log N)^{2/3}\right)$ for some $c < 2$. However, even if GNFS is known as the fastest method for factoring large integers, as long as quantum computers are not concretely invented, it is not a real threat for RSA when the modulus is sufficiently large.

- **Chosen ciphertext attacks**

A chosen ciphertext attack on RSA works as follows. Suppose that the attacker wants to decrypt a ciphertext c with a public key (N, e) and a corresponding private key (N, d). He chooses a random x and asks for the decryption of $y \equiv cx^e(\text{mod } N)$ and receives $\hat{y} \equiv y^d(\text{mod } N)$. Since $\hat{y} \equiv y^d \equiv c^d(x^e)^d \equiv c^d x(\text{mod } N)$, then he retrieves the plaintext by computing $c^d \equiv \hat{y}x^{-1}(\text{mod } N)$.

- **Side channel attacks**

Since the introduction of side-channel attacks, the RSA cryptosystem has been a privileged target, but a wide variety of countermeasures have been proposed making these attacks ineffective. For example, timing attacks are used to analyze the execution time that result from input parameters. In RSA, the timing attacks concern the implementation phase when computing exponentiation $c^d \pmod N$ in the decryption process. The main countermeasure in RSA consists in making sure that the implementation always takes the same amount of time.

A very recent side channel attack was mounted by Genkin, Shamir and Tromer by using acoustic technique. This attack can factor up to 4096-bit RSA modulus if the attacker is physically close to your data.

- **Algebraic attacks**

The prominent attacks on the RSA cryptosystem are the algebraic attacks and are mostly based on two classes of techniques. The first class uses the continued fraction algorithm such as Wiener's attack (Wiener, 1990), and the second class uses Coppersmith's method (Coppersmith, 1997) for solving polynomial equations. Wiener's attack is very efficient when the decryption exponent is small enough while Coppersmith's method can be applied to various sizes of encryption or decryption exponents. Coppersmith's method is very efficient and is widely used to attack the RSA cryptosystem. In 2013, Bernstein *et al.*(Smartfact's website, 2013) used Coppersmith's method to factor at least 81 distinct 1024-bit RSA keys from Taiwan's national "Citizen Digital Certificate" database. Also, Coppersmith's method is very useful to launch new algebraic attacks on RSA. Very recently, Nitaj, Arrifin, Diaa and Bahig presented a new attack on RSA at Africacrypt 2014. The new attack uses Coppersmith's method and leads to the factorization of the RSA modulus N under certain conditions.

Keywords: RSA, integer factorization problem, Coppersmith method

REFERENCES

Bernstein, D. J., Chang, Y. A., Cheng, C. M., Chou, L. P., Heninger, N., Lange, T., and van Someren, N. 2013. Factoring RSA keys from certified smart cards: Coppersmith in the wild. In *Advances in Cryptology-ASIACRYPT 2013* (pp. 341-360). Springer Berlin Heidelberg.

Boneh, D., and Durfee, G. 2000. Cryptanalysis of RSA with private key d less than N 0.292. *Information Theory, IEEE Transactions on*, *46*(4), 1339-1349.

Coppersmith, D. 1997. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of Cryptology*, *10*(4), 233-260.

Genkin, D., Shamir, A., and Tromer, E. 2013. *RSA Key Extraction via Low-Bandwidth Acoustic Cryptanalysis*. Cryptology ePrint Archive, Report 2013/857, 2013. http://eprint. iacr. org.

Nitaj, A., Ariffin, M. R. K., Nassr, D. I., and Bahig, H. M. 2014. New Attacks on the RSA Cryptosystem. In *Progress in Cryptology–AFRICACRYPT 2014* (pp. 178-198). Springer International Publishing.

Rivest, R. L., Shamir, A., and Adleman, L. 1978. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, *21*(2), 120-126.

Wiener, M. J. 1990. Cryptanalysis of short RSA secret exponents. *Information Theory, IEEE Transactions on*, *36*(3), 553-558.

Revisiting Identity-Based Encryption towards No Escrow Application and Analysis of Embedding Secret Key Information in RSA Moduli

Kouichi Sakurai [*],
Department of Informatics, Kyushu University
Fukuoka, Japan
sakurai@inf.kyushu-u.ac.jp

EXTENDED ABSTRACT

This talk first surveys existing approaches to assure public keys in a public-key cryptosystem. One approach is that a certificate issued by a reliable certification authority (CA) guarantees validity of a public key. The certificate works to prevent impersonations on the public key. The infrastructure to realize this function is usually called a public-key infrastructure (PKI). Another approach is identity-based encryption (IBE), which does not need those certificates. In IBE system, a participant's ID can be used as his public key. The corresponding secret key, on the other hand, is obtained from a key distribution center. Hence there is a key-escrow problem that user's secret keys are known to the key distribution center. We discuss the key-escrow problem associated with IBE systems.

Next, we report analysis of the Lenstra's algorithm to embed information into RSA moduli. Then we modify the Lenstra's algorithm and consider how to tackle the key-escrow problem. The modified algorithm enables us to propose an IBE-like system without the key-escrow problem. In our system, a candidate-participant can create his secret key and public key, where he can embed his characteristic information into the public key. The characteristic information can be taken as his ID, a guarantor's ID and a string which proves that he certainly made his secret key. Here the guarantor can be taken as any participant of the underlying network of our IBE-like system; we do not need any certificate issued by CA, and simultaneously, we can avoid the key-escrow problem.

Our system has a useful application to digital rights management (DRM) system. When a participant tries to send a file to a receiver who is also a participant, the sender can verify whether the receiver joined legitimately into the network or not by checking guarantor's ID and receiver's string. Here, if we want, we can further verify whether the guarantor joined legitimately into the network or not by checking in the same way, like checking the block-chain of the Bitcoin. Our DRM system can be called a flat model because it is P2P-model and a guarantor can be taken as any participant.

Keywords: PKI, IBE, key escrow, information embedding

REFERENCES:

Shamir, A. 1985. Identity-based cryptosystems and signature schemes. In *Advances in cryptology* (pp. 47-53). Springer Berlin Heidelberg.

Girault, M. 1991. Self-certified public keys. In *Advances in Cryptology—EUROCRYPT'91* (pp. 490-497). Springer Berlin Heidelberg.

[*] This is joint work with KITAHARA, YASUDA, NISHIDE, ANADA and KAWAMOTO which is presented partially in [M. Kitahara, T. Nishide and K. Sakurai, A method for embedding secret key information in RSA public key and its application, Sixth International Workshop on Advances in Information Security, pp.665-670, (2012)], [M. Kitahara, T. Yasuda, T. Nishide and K. Sakurai, Upper Bound of the Length of Efficiently Embedding Information in RSA Modulus, Proc. AsiaPKC2013, pp33-39 (2013)] and [M. Kitahara, H. Anada, J. Kawamoto and K. Sakurai, An Embedding Method of Owner's Information into Public Key of RSA Encryption and Its Application to Digital Rights Management System, IPSJ SIG Technical Report, Vol.2014-CSEC-65 No.3 (2014)].

Vanstone, S. A., and Zuccherato, R. J. 1995. Short RSA keys and their generation. *Journal of Cryptology*, *8*(2), 101-114.

Lenstra, A. K. 1998. Generating RSA moduli with a predetermined portion. In *Advances in Cryptology—Asiacrypt'98* (pp. 1-10). Springer Berlin Heidelberg.

Boneh, D., and Franklin, M. 2001. Identity-based encryption from the Weil pairing. In *Advances in Cryptology—CRYPTO 2001* (pp. 213-229). Springer Berlin Heidelberg.

Cocks, C. 2001. An identity based encryption scheme based on quadratic residues. In *Cryptography and Coding* (pp. 360-363). Springer Berlin Heidelberg.

Al-Riyami, S. S., and Paterson, K. G. 2003. Certificateless public key cryptography. In *Advances in Cryptology-ASIACRYPT 2003* (pp. 452-473). Springer Berlin Heidelberg.

Laih, C. S., and Chen, K. Y. 2004. Generating visible RSA public keys for PKI.*International Journal of Information Security*, *2*(2), 103-109.

Boneh, D., Gentry, C., and Hamburg, M. 2007. Space-efficient identity based encryptionwithout pairings. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on* (pp. 647-657). IEEE.

Joye, M. 2008. RSA moduli with a predetermined portion: Techniques and applications. In Information Security Practice and Experience (pp. 116-130). Springer Berlin Heidelberg.

Abelian varieties and theta functions for cryptography

Andreas Enge
University of Bordeau, IMB, UMR 5251,
F-33400 Talence, France
Email: andreas.enge@inria.fr

ABSTRACT

Abelian varieties have imposed themselves as the structure of choice for the implementation of Diffie–Hellman based cryptosystems, due to their favourable ratio of quality (high security) and price (efficient implementability). They come in two flavours, as elliptic curves of the form $Y^2 = X^3 + aX + b$ and as Jacobians of genus 2 hyperelliptic curves of the form $Y^2 = X^5 + \cdots$. Hyperelliptic curves of higher genus (with a right hand side of higher degree) and curves of a different shape have turned out to be less secure. It has been shown that in the presence of a suitable auxiliary group (which, remarkably, is again an abelian variety and which can heuristically be found in subexponential time) the Diffie–Hellman problem and the discrete logarithm problem are polynomially equivalent (Maurer and Wolf, 1999). Except for special cases, for instance curves over finite fields of composite degree that can sometimes be embedded into Jacobians of higher genus curves via Weil descent (Frey, 2001; Gaudry *et al.*, 2002), only attacks of exponential complexity are known for elliptic and genus 2 curves.

While the curves and their associated groups are defined over finite fields in the cryptographic context, studying them over the complex numbers leads to a better understanding. Analytic properties observed there often carry over to the algebraic setting that is valid over finite fields. Precisely, complex abelian varieties are given by $\mathbb{C}^g$ modulo a lattice of dimension 2g, where the group law is the usual addition of complex numbers. Theta functions can be used to embed them into projective space, and the rich structure of theta functions makes it possible to formulate an equivalent algebraic group law. A simpler embedding by theta functions yields the Kummer variety, containing the points of the abelian variety up to sign, which is enough for Diffie–Hellman based cryptography. The resulting group law, suggested in (Gaudry, 2007) following (Chudnovsky and Chudnovsky, 1986), shows that a priori more complicated varieties of dimension 2 can compete with elliptic curves concerning their efficient implementation, and Kummer arithmetic is also among the fastest possibilities for elliptic curve cryptosystems.

The theory of theta functions is closely linked to that of complex multiplication (CM), which describes the endomorphism rings of abelian varieties and which can be used to taylor varieties to one's needs; especially in the very constrained context of pairing-based cryptography, complex multiplication is the only way of obtaining secure instances. Floating point computations over the complex numbers can be used to obtain algebraic information on the occurring number fields. Ultimately, an algorithm of quasi-linear complexity (that is, linear up to logarithmic factors) for evaluating theta functions (Dupont, 2011) leads to a quasi-linear algorithm for computing class polynomials, the algebraic main ingredient of the CM approach for elliptic curves (Enge, 2009). In this setting, purely algebraic algorithms based on Chinese remaindering have turned out to be faster in practice, see (Enge and Sutherland, 2010) and the references given there. In genus 2, the floating point approach has a better complexity (Streng, 2010) and is very fast in practice, as shown by an implementation that is made available under a free, copyleft licence (Enge and Thomé, 2013; Enge and Thomé, 2014).

Keywords: elliptic curves, hyperelliptic curves, theta functions, complex multiplication

REFERENCES

Chudnovsky, D. V., and Chudnovsky, G. V. 1986. Sequences of numbers generated by addition in formal groups and new primality and factorization tests. *Advances in Applied Mathematics*, *7*(4), 385-434.

Dupont, R. 2011. Fast evaluation of modular functions using Newton iterations and the AGM. *Mathematics of Computation*, *80*(275), 1823-1847.

Enge, A. 2009. The complexity of class polynomial computation via floating point approximations. *Mathematics of Computation*, *78*(266), 1089-1107.

Enge, A., and Sutherland, A. V. 2010. Class invariants by the CRT method. In*Algorithmic Number Theory* (pp. 142-156). Springer Berlin Heidelberg.

Frey, G. 2001. Applications of arithmetical geometry to cryptographic constructions. In Jungnickel, D. and Niederreiter, H., editors, Finite *Fields and Applications — Proceedings of The Fifth International Conference on Finite Fields and Applications Fq5* , held at the University of Augsburg, Germany, August 2–6, 1999, pages 128–161, Berlin. Springer-Verlag.

Gaudry, P. 2007. Fast genus 2 arithmetic based on Theta functions. *Journal of Mathematical Cryptology JMC*, *1*(3), 243-265.

Gaudry, P., Hess, F., and Smart, N. P. 2002. Constructive and destructive facets of Weil descent on elliptic curves. *Journal of Cryptology*, *15*(1), 19-46.

Maurer, U. M., and Wolf, S. 1999. The Relationship Between Breaking the Diffie--Hellman Protocol and Computing Discrete Logarithms. *SIAM Journal on Computing*, *28*(5), 1689-1721.

Streng, T. C. 2010. *Complex multiplication of abelian surfaces*. Mathematical Institute, Faculty of Science, Leiden University.

Accountable Privacy in Online Communication

Sherman S. M. Chow

ABSTRACT

Advances in communication technologies and cyber-physical systems change the way people communicate. For example, Internet brings together people who do not know each other before hand to engage in discussion of a certain topic. People can make seamless payment using stored-value smartcard or auto-toll devices. One indispensable feature in these systems is privacy. However, perfect privacy may be abused, e.g., vandalism in discussion forum, money laundering, etc.

Anonymous credential offers a cryptographic authentication mechanism that aims to provide accountability and privacy at the same time. They are useful in applications which just require a proof of certain attributes such as membership, instead of positive identification of the user. Normally, the users remain anonymous, yet they are still accountable for their actions after authentication, by the means of anonymity revocation. Years of research efforts have been made on balancing between privacy and accountability, which is also the main theme of this talk.

Firstly, we aim to improve real-world anonymous credential realizations, such as Microsoft's U-Prove, which does not support revocation without compromising untraceability. We aim to develop a new, efficient, and modular mechanism for revocation. Techniques developed will also benefit other anonymous credential systems.

Secondly, we are particularly interested in application of anonymous credentials technologies in supporting privacy-preserving electronic payment. Realizing privacy-preserving payment systems is closely related to the problem of striking a balance in privacy and accountability. We will briefly talk about they key cryptographic primitives involved.

The final focus of this talk is on anonymous blacklistable credentials for the Internet, which a trusted party for ensuring privacy is often absent. Apart from blacklisting bad behaviour, rewarding user contribution via reputation score is equally important. Existing mechanisms either require costly authentication process or a timely evaluation of all authenticated sessions to see if blacklisting is required. We aim to propose a new framework to resolve all these problems simultaneously.

Improving any of these areas will make an impact on potentially millions of users worldwide. Part of the results described in this talk are joint work with Microsoft Research, Intel, and NTT R&D. We hope this talk can stimulate research in new cryptographic building blocks and new paradigms of constructing cryptographic anonymous credential systems, which may make a broad impact on cryptography, distributed systems, and privacy research communities.

Keywords: anonymity, identification, authentication

REFERENCES

Abe, M., Chow, S. S., Haralambiev, K., and Ohkubo, M. 2013. Double-trapdoor anonymous tags for traceable signatures. *International journal of information security, 12*(1), 19-31.

Acar, T., Chow, S. S., and Nguyen, L. 2013. Accumulators and U-prove revocation. In *Financial Cryptography and Data Security* (pp. 189-196). Springer Berlin Heidelberg.

Chow, S. S., He, Y. J., Hui, L. C., and Yiu, S. M. 2012. Spice–simple privacy-preserving identity-management for cloud environment. In *Applied Cryptography and Network Security* (pp. 526-543). Springer Berlin Heidelberg.

Yu, K. Y., Yuen, T. H., Chow, S. S., Yiu, S. M., and Hui, L. C. 2012. Pe (ar) 2: Privacy-enhanced anonymous authentication with reputation and revocation. In*Computer Security–ESORICS 2012* (pp. 679-696). Springer Berlin Heidelberg.

Chow, S. S. 2009 Real traceable signatures. In *Selected Areas in Cryptography* (pp. 92-107). Springer Berlin Heidelberg.

Reverse Engineering Cryptography and Obfuscation Routine Inside Android Malware

Mahmud Ab. Rahman

ABSTRACT

The ubiquity of the mobile platform and growing threats to mobile applications calls for increased vigilance on the part of organizations analyzing malicious mobile applications. Malware author are aiming for users of mobile operating system with a malicious application that harvests personal information, controlling the system and sends it to a remote server. Smartphone malware has become more sophisticated in nature, and most of them will try to add layer of obfuscation to minimize of being detected.

Mobile malware is likely not going to decrease in attempts because the rapid growth of mobile is projected to continue. As increased use of mobile to connect to the Internet occurs, users can probably expect to see increased levels of exploits. Malware infection on Android platform is going to be interesting in future. Thus, reversing Android malware is interesting challenge to address. Malware analysis can be performed using two approaches, which are Dynamic Analysis and Dead-Listing Analysis (Reverse Engineering). Many methods of obfuscation have been applied in Android malware to prevent analysis and to hide information such as command and control URL, information and many more. By abusing cryptography implementation, malware author is capable of securing and obfuscated his/her data.

The speaker will discuss on technical analysis on reversing encryption routine inside malicious Android malware. The analysis parts will focus on dissecting obfuscation abusing commonly used cryptography algorithm such as AES, XOR and few obfuscation techniques applied inside the Android malware. The challenges when dealing with reversing encryption implementation inside Android malware will be addressed as well.

Keywords: malware, obfuscation, reversing encryption routine

Malaysian Electronic ID (eID), Challenges and Moving Forward

Nik Khairul Raja Abdullah
Business Architect
Digicert Sdn Bhd, Subsidiary of POS Malaysia

ABSTRACT

Malaysia started its PKI implementation in 1998 with the enactment of the Digital Signature Act in 1997 (DSA 1997). Since then, the licensed Certificate Authorities (CA) specifically Digicert Sdn Bhd have gone through numerous challenges and also successes in adopting PKI in Malaysia. There are so many lessons to be learned in order to implement prudent electronic ID (eID) in Malaysia. Moving forward, Digicert shall share our visions, beliefs and initiatives for pushing eID to a greater scale nation-wide. That is, by building eID digital ecosystem by leveraging on our strengths, expertise and experiences.

Keywords: PKI, Certificate Authority, Electronic ID

Certificateless Public Key Encryption (CL-PKE) Scheme Using Extended Chebyshev Polynomial Over the Finite Field $\mathbb{Z}_p$

[1]Mohammed Benasser Algehawi and [2]Azman Samsudin
[1]Libyan Defense Ministry, Tripoli, Libya, [2]School of Computer Sciences, Universiti Sains Malaysia,
Penang, 11800, Malaysia
Email: [1]malgehawi@yahoo.com, [2]azman@cs.usm.my

ABSTRACT

In this paper, we introduce a new alternative model for a Certificateless Public Key Encryption (CL-PKE) scheme. Our proposed CL-PKE scheme uses the Identity Based Encryption (IBE) scheme extended over the finite field $\mathbb{Z}_p$ to generate its encryption and decryption keys. The proposed system is applicable, secure, and reliable.

Keywords: Certificateless Public Key Encryption; Chebyshev polynomial extended over $\mathbb{Z}_p$; Public key cryptography; Chaos cryptography.

1. INTRODUCTION

Certificateless Public Key Encryption is a scheme in which there is no intervention by the Key Generation Center (KGC) during the generation of the encryption and decryption keys. The CL-PKE scheme was introduced by Al-Riyami and Paterson in 2003.

The CL-PKE scheme uses the Identity Based Encryption (IBE) scheme introduced by for the generation of the users' encryption and decryption keys. The decryptor receives a partial-private-key from the KGC through a secure channel. Unlike the Private Key Generator (PKG) in the scheme introduced by Boneh and Franklin in 2003, the KGC has no access to the users' secret information and the only role of the KGC is to generate the partial-private-key. The received partial-private-key will be used by the decryptor to generate its actual private-key.

The security of the CL-PKE scheme against the Indistinguishable Chosen Ciphertext Attack (IND-CCA2) has been proved (Al-Riyami and Paterson, 2003). The strength of the scheme is inherited from the bilinear Diffie-Hellman hard problem (BDHP), the hard problem that is used in the parameters generation of this scheme. The formal definition of the CL-PKE concept and its aspects are briefly explained in the following subsection based on the CL-PKE description (Al-Riyami and Paterson, 2003).

2. BASIC AND FULLY SECURE CL-PKE SCHEME BASED ON PAIRINGS

The execution of the basic CL-PKE scheme starts with the introduction of the security parameter k and the BDHP parameter generator $\mathcal{G}$ to the setup algorithm to generate the system parameters. The basic CL-PKE scheme is performed as follows:

A. Setup: This algorithm generates the master-key s and the public parameters;
$g = \langle G_1, G_2, \hat{e}, n, P, P_0, H_1, H_2 \rangle$

B. This algorithm runs as follows:
1. Generate two groups G_1 and G_2 of prime order q and an admissible map $\hat{e}: G_1 \times G_1 \to G_2$.
2. Choose a random generator $P \in G_1$.
3. Choose a random master-key $s \in \mathbb{Z}_q^*$ and calculate the public key $P_0 = sP$.
4. Select two hash functions $H_1: \{0,1\}^* \to G_1^*$ and $H_2: G_2 \to \{0,1\}^n$ for some bit-length n.
5. Choose the message space $\mathcal{M} = \{0,1\}^n$ and the ciphertext space $C = G_1 \times \{0,1\}^n$.

C. Partial-Private-Key-Extract: This algorithm generates the partial private key for party A as follows:
1. Takes as an input the master-secret key s and the identity of party A, $ID_A \in \{0,1\}^*$.

2. Map the identity ID_A to the group G_1^* such that $Q_A = H_1(ID_A) \in G_1^*$.
3. Finally, generate party A's partial private key, $D_A = sQ_A \in G_1^*$.

D. Set-Secret-Value: This algorithm outputs the randomly selected value $x_A \in \mathbb{Z}_q^*$ as the secret value of party A by taking the public parameters g and A's identity ID_A as the inputs.

E. Set-Private-Key: This algorithm generates party A's private key as follows:

1. Takes as input the public parameters g, A's partial private key D_A, and A's secret value x_A.
2. Generates A's private key $S_A \in G_1^*$, such that $S_A = x_A D_A = x_A s Q_A$.

F. Set-Public-Key: This algorithm generates A's public key as follows:

1. Takes as input the public parameters g and A's secret value x_A.
2. Generate A's public key $P_A = \langle X_A, Y_A \rangle$, such that $X_A = x_A P$ and $Y_A = x_A P_0 = x_A s P$.

G. Encryption: This algorithm encrypts the message M as follows:

1. First checks that $X_A, Y_A \in G_1^*$ and make sure that $\hat{e}(X_A, P_0) = \hat{e}(Y_A, P)$. If so then encrypt the message M, otherwise output $\perp$ and abort the encryption.
2. Set $Q_A = H_1(ID_A) \in G_1^*$.
3. Select a random value $r \in G_1^*$.
4. Compute the ciphertext $C = (U, V)$, where, $U = rP$ and $V = M \oplus H_2(\hat{e}(Q_A, Y_A)^r)$.

H. Decryption: This algorithm decrypts the ciphertext. Upon receiving the ciphertext $C = (U, V)$, the ciphertext will be decrypted using A's private key S_A as follows:

$$\begin{aligned} M &= V \oplus H_2\big(\hat{e}(S_A, U)\big) \\ &= V \oplus H_2\big(\hat{e}(x_A s Q_A, rP)\big) = V \oplus H_2(\hat{e}(Q_A, x_A s P)^r) \\ &= V \oplus H_2(\hat{e}(Q_A, Y_A)^r) \\ &= M. \end{aligned}$$

The adaptation of the CL-PKE scheme that is fully secure against IND-CCA2 is obtained by incorporating the technique proposed by Fujisaki and Okamoto in 1999. After adding the two extra hash functions H_3 and H_4 as random oracles, this fully secure CL-PKE scheme will be as the following (Al-Riyami and Paterson, 2003):

The setup algorithm will be as in the basic CL-PKE scheme except that two extra random oracles, $H_3: \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_q^*$ and $H_4: \{0,1\}^n \to \{0,1\}^n$ will be added. The new public parameters will be $g = \langle G_1, G_2, \hat{e}, n, P, P_0, H_1, H_2, H_3, H_4 \rangle$, the message space will be the same as in the basic scheme, and the ciphertext space will be defined as $C = G_1 \times \{0,1\}^{2n}$. The encryption and the decryption algorithms will be executed as follows:

A. Encryption:

1. First checks that $X_A, Y_A \in G_1^*$ such that $\hat{e}(X_A, P_0) = \hat{e}(Y_A, P)$. If the equivalence holds then encrypt the message otherwise output $\perp$ and abort the encryption.
2. Set $Q_A = H_1(ID_A) \in G_1^*$.
3. Select $\sigma \in \{0,1\}^n$.
4. Set $r = H_3(\sigma, M)$.
5. Send the ciphertext $C = [U, V, W]$ where $U = rP$, $V = H_2(\hat{e}(Q_A, Y_A)^r)$, and $W = M \oplus H_4(\sigma)$.

B. Decryption: Party B will decrypt C as follows:

1. Calculate $\sigma = V \oplus H_2\big(\hat{e}(U, S_A)\big)$.
2. Calculate $M = W \oplus H_4(\sigma)$.
3. Set $r = H_3(\sigma, M)$. If $U \neq rP$ then output $\perp$ and reject the ciphertext, otherwise accept the decrypted message, M.

In this paper, we will present a new CL-PKE scheme that uses the same method used by Al-Riyami and Paterson in 2003. The proposed CL-PKE uses the mechanism of the IBE scheme as introduced by Algehawi and Samsudin in 2010 for the generation of the users' encryption and decryption keys. The strength of the IBE scheme (Algehawi and Samsudin, 2010) is based on Chebyshev map bilinearity and the discrete Chebyshev hard problem (DCP) which arises after extending the map over the finite field $\mathbb{Z}_p$. To demonstrate its validity for CL-PKE purposes, the characteristics of the Chebyshev polynomial extended over the finite field $\mathbb{Z}_p$ are explained in the following section.

3. CHEBYSHEV POLYNOMIAL

In Sections 3.1 and 3.2, the Chebyshev polynomial will be briefly explained, both in the real domain $\mathbb{R}$ and the extended finite field $\mathbb{Z}_p$. The pertinent definitions and properties of the Chebyshev polynomial extended over the finite field $\mathbb{Z}_p$ will be given. These properties strengthen the proposed scheme security.

3.1 Chebyshev Polynomial in the Real Domain

The Chebyshev polynomial in the real domain has some properties that make it usable for cryptography purposes. The Chebyshev polynomial in the real domain has been defined in many publications (Amig *et al.*, 2008; Xiao et al, 2007; Yoon and Yoo, 2008). The definition of the n^{th} term T_n of the Chebyshev polynomial can be written as follows:

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), \tag{1}$$

where $n \in \mathbb{N},\ n \geq 2, x \in \mathbb{R}$, and the initial terms are $T_0(x) = 1$ and $T_1(x) = x$.

The Chebyshev polynomial in the real domain has been proven to be weak for the purpose of cryptography ((Bergamo *et al.*, 2004), (Han, 2008), (Xiang *et al.*, 2009)). The small range of the real domain, which is [-1,1], results in weakening the DCP hard problem obtained from the one-way function of the Chebyshev polynomial.

3.2 Chebyshev Polynomial Extended Over the Finite Fields $\mathbb{Z}_p$

The extension of the Chebyshev polynomial over the finite field $\mathbb{Z}_p$ has been discussed in several places in the literature ((Algehawi and Samsudin, 2010), (Bi and Wang, 2009), (Maze, 2003), (Wang *et al.*, 2008), (Wang and Zhao, 2010)). This extension does not alter the bilinear property of the Chebyshev polynomial, but it does strengthen the DCP hard problem. This extension and its details, including the one way function of the extended Chebyshev polynomial, have been explained extensively ((Algehawi and Samsudin, 2010), (Bi and Wang, 2009), (Maze, 2003), (Wang *et al.*, 2008), (Wang and Zhao, 2010)). The DCP of the extended Chebyshev polynomial has been proven to be as hard as the Discrete Logarithmic Problem (DLP), such that can be used safely for cryptographic purposes ((Bi and Wang, 2009), (Wang *et al.*, 2008), (Maze, 2003). The extended Chebyshev polynomial equation is defined as follows:

$$T_n(x) = \left(2xT_{n-1}(x) - T_{n-2}(x)\right) \pmod p, \tag{2}$$

where the initial terms are $T_0(x) = 1 \pmod p$ and $T_1(x) = x \pmod p$. The semi group property (bilinearity) of the extended Chebyshev polynomial can be derived as follows:

$$\begin{aligned} T_r\left(T_s(x)\right) \bmod p &= T_r(T_s(x) (\bmod p)) \bmod p \\ &= T_{rs}(x) \bmod p \\ &= T_s(T_r(x) (\bmod p)) \bmod p \end{aligned}$$

$$\begin{aligned} &= T_{rs}(x) \bmod p \\ &= T_s\big(T_r(x)\big) \bmod p, \end{aligned} \tag{3}$$

where $r, s, x \in \mathbb{Z}_p^*$, and $r, s \geq 2$.

From Eq. 2, the representation of the secret information n can be written as a product of primes, $n = S_1^{k_1} \times S_2^{k_2} \times \cdots \times S_m^{k_m}$, where $S_1, S_2, \cdots, S_m$ are prime numbers and $k_1, k_2, \cdots, k_m, m \in \mathbb{Z}^+$. Based on this expression, Eq. 2 can be represented as the following:

$$\begin{aligned} T_n(x) &= T_{s_1^{k_1} \times s_2^{k_2} \times \cdots \times s_m^{k_m}}(x) (\bmod p) \\ &= T_{s_1^{k_1}}\left(T_{s_2^{k_2}}\left(\cdots T_{s_m^{k_m}}(x) \cdots\right)\right) (\bmod p). \end{aligned} \tag{4}$$

If $T_n(x)$ and x are known, to find n, one has to compute $T_r(x)$ for all $r = S_1^{k_1} \times S_2^{k_2} \times \cdots \times S_l^{k_l}$, $l \in \mathbb{Z}^+$ and find the r for which $T_n(x) = T_r(x)$ process is infeasible for large n .

4. IBE SCHEME USING THE CHEBYSHEV POLYNOMIAL EXTENDED OVER FINITE FIELD $\mathbb{Z}_p$

The IBE scheme using the Chebyshev polynomial extended over finite field $\mathbb{Z}_p$ consists of four main algorithms, Setup, Extraction, Encryption, and Decryption. The basic IBE scheme is explained below:

Based on Eq. 4, the basic IBE scheme is executed as follows (Algehawi and Samsudin, 2010):

A. Setup (by PKG):

1. Choose a large prime number p, a large (≥ 2) secret number $s \in \mathbb{Z}_p^*$ as the secret key, and a global parameter $G \in \mathbb{Z}_p^*$.
2. Use Eq. 4 to calculate P_{Pub} as follows:

$$P_{Pub} = T_s(G) \ (\bmod p), \tag{5}$$

where $\{p, G, P_{Pub}\}$ are the public parameters of the PKG.

3. Publish the public parameters $(p, G, P_{Pub}, H_1, H_2)$ where $H_1\colon \{0,1\}^* \to \mathbb{Z}_p^*$, $H_2\colon \mathbb{Z}_p^* \to \{0,1\}^n$.

B. Extraction (by PKG): The PKG extracts the shared key, K_{shared}, for party B based on Eq. 4 as follows:

$$K_{shared} = T_s\big(T_{B_{Pub}}(A_{Pub}) \bmod p\big) \bmod p, \tag{6}$$

where $B_{Pub} = H_1(ID_B)$ is party's B identity.

C. Encryption (by Party A):

1. Choose a large integer $A_{Pri} \in \mathbb{Z}_p^*$ where $A_{Pri} \geq 2$ as its private key.
2. Calculate the public key, A_{Pub}, using Eq. 4 as follows:

$$U = A_{Pub} = T_{A_{Pri}}(G) \ (\bmod p). \tag{7}$$

3. Calculate the shared secret key, K_{shared}, as follows:

$$K_{shared} = T_{A_{Pri}}\left(T_{B_{Pub}}(P_{Pub}) \ (\bmod p)\right) \ (\bmod p), \tag{8}$$

where $B_{Pub} = H_1(ID_B)$ is Party B's identity.

4. Encrypt message M using the shared secret key, K_{shared}, and produce the ciphertext component $V = M \oplus H_2(K_{shared})$. Send the ciphertext $C = (U, V)$ to party B.

D. Decryption algorithm (by Party B):

1. Receive the shared key, K_{shared}, from the PKG through a secure channel.
2. Decrypt message M using the shared secret key, K_{shared}, as follows:

$$M = V \oplus H_2(K_{shared})$$

To obtain the fully secure version of this scheme, another two hash functions, H_3 and H_4, will be added to the scheme as random oracles. The details of the fully secure version, its security proofs against IND-CCA2, and its practicality have been explained (Algehawi and Samsudin, 2010).

5. THE PROPOSED CL-PKE SCHEME USING THE CHEBYSHEV POLYNOMIAL EXTENDED OVER FINITE FIELD $\mathbb{Z}_p$

The new CL-PKE scheme using the Chebyshev polynomial extended over $\mathbb{Z}_p$ is discussed in this section. Subsection 5.1 explains the proposed basic CL-PKE scheme which uses the IBE concept introduced by Algehawi and Samsudin in 2010). Subsection 5.2 shows the proposed fully secure CL-PKE scheme.

5.1 The Proposed Basic CL-PKE Scheme

The basic version of the proposed CL-PKE scheme consists of seven algorithms:

A. Setup Algorithm:

1. Randomly choose a large prime p.
2. Choose a random master-key $s \in \mathbb{Z}_p^*$, where $s \geq 2$ and a global parameter $G \in \mathbb{Z}_p^*$.
3. Using Eq. 4, calculate the public key P_0 as follows:

$$P_0 = T_s(G) \bmod p. \quad (9)$$

4. Select two hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ and $H_2 : \mathbb{Z}_p^* \rightarrow \{0,1\}^n$.
5. Choose the message space $M = \{0,1\}^n$, and ciphertext space $C = \mathbb{Z}_p^* \times \{0,1\}^n$, for some integer n.
6. Publish the public parameters, $g = \langle p, P_0, G, n, H_1, H_2 \rangle$.

B. Partial-Private-Key-Extract Algorithm: Generates the partial private key for party A as follows:

1. Takes as input, the identity of party A, $ID_A \in \mathbb{Z}_p^*$, and party's B public key B_{pub}.
2. Set $Q_A = H_1(ID_A) \in \mathbb{Z}_p^*$.
3. Use Eq. 4 to extract party A's partial private key as follows :

$$\begin{aligned} D_A &= T_s\left(T_{Q_A}(B_{pub}) \bmod p\right) \bmod p, \\ &= T_s\left(T_{Q_A}\left(T_{B_{pri}}(G)\right)\right) \pmod p. \end{aligned} \quad (10)$$

C. Set-Secret-Value Algorithm: Select a random value $x_A \in \mathbb{Z}_p^*$ as the secret value for party A.

D. Set-Shared-Secret-Key Algorithm: Using Eq. 4, generate the shared secret key K_{shared} by taking as input the public parameters g, A's partial private key D_A, and A's secret value x_A as follows:

$$\begin{aligned} K_{shared} &= T_{x_A}(D_A) \bmod p,, \\ &= T_{x_A}\left(T_s\left(T_{Q_A}\left(T_{B_{pri}}(G) \bmod p\right)\right)\right) \bmod p. \end{aligned} \quad (11)$$

E. Set-Public-Key Algorithm: Use Eq. 4 to generate A's public key P_A by using Eq. 4 as follows:

6. Take as input the public parameters g and A's secret value x_A.
7. Calculates the tuple, $P_A = \langle X_A, Y_A, Z_A \rangle$, where:

$$
\begin{aligned}
X_A &= T_{x_A}(G) \bmod p., && (12)\\
Y_A &= T_{x_A}(U) \bmod p \\
&= T_{x_A}\big(T_{B_{Pri}}(G) \bmod p\big) \bmod p. && (13)\\
Z_A &= T_{x_A}(P_0) \bmod p \\
&= T_{x_A}(T_s(G) \bmod p) \bmod p. && (14)
\end{aligned}
$$

F. Encryption Algorithm: The encryption is executed as follows:

1. First check that $X_A, Y_A, Z_A \in \mathbb{Z}_p^*$, and make sure that $T_{B_{Pri}}(X_A)\ mod\ p = Y_A$. If so, then encrypt the message; otherwise, output $\perp$, and abort the encryption.
2. Set $Q_A = H_1(ID_A) \in \mathbb{Z}_p^*$.
3. Choose a large integer $B_{Pri} \in \mathbb{Z}_p^*$ where $B_{Pri} \geq 2$ as the private key.
4. Generate the public key,

$$U = B_{Pub} = T_{B_{Pri}}(G) \bmod p. \qquad (15)$$

5. Compute the ciphertext $C = (U, V)$, where, $V = M \oplus H_2(\,K_{shared})$ and the shared key K_{shared} is computed as follows:

$$
\begin{aligned}
K_{shared} &= T_{B_{Pri}}\big(T_{Q_A}(Z_A) \bmod p\big) \bmod p\ , \\
&= T_{B_{Pri}}\Big(T_{Q_A}\big(T_{x_A}(P_0) \bmod p\big) \bmod p\Big) \bmod p \\
&= T_{B_{Pri}}\Big(T_{Q_A}\big(T_{x_A}(T_s(G)\,)\,\big)\Big)\,(\mathrm{mod}\ p). && (16)
\end{aligned}
$$

G. Decryption Algorithm: Upon receiving the ciphertext $C = (U, V)$, using the shared secret key K_{shared} generated earlier by Eq. 11, the ciphertext is decrypted as follows: $M = V \oplus H_2(K_{shared})$.

Working example

Table 1 shows a working example of the basic CL-PKE scheme. The results show that the communicating parties produced the same shared key $\mathrm{K}_{\mathrm{shared}}$.

Algorithm	Step[c]	Description	Generation method	Key value
Setup	A.1	Large prime p	Chosen	15485863
	A.2	Secret key $s \in \mathbb{Z}_p^*,\ s \geq 2$	Chosen	877
		Global parameter $G \in \mathbb{Z}_p^*$	Chosen	673
	A.3	Public key P_0	Eq. 9	14014563
Partial-Private-Key Extraction	B.2	Set the party A public value Q_A	$Q_A = H_1(ID_A)$	305
	B.3	Extracts the partial private key D_A	Eq. 10	3274740
Set-Secret-Value	C	Selected a random value $x_A \in \mathbb{Z}_q^*$	Chosen	859
Set-Shared-Secret-Key	D	Generates the shared secret key K_{shared}	Eq. 11	26560
Set-Public-Key P_A	E.2	Calculates X_A	Eq. 12	5787979
	E.2	Calculates Y_A	Eq. 13	9146288
	E.2	Calculates Z_A	Eq. 14	10741254
Encryption	F.1	checks that $X_A, Y_A, Z_A \in \mathbb{Z}_p^*$ and $T_{B_{Pri}}(X_A) \bmod p = Y_A$	Calculation	9146288 = Y_A
	F.2	Set the party A public value Q_A	$Q_A = H_1(ID_A)$	305
	F.3	Private key $B_{Pri} \in \mathbb{Z}_p^*,\ B_{Pri} \geq 2$	Chosen	587
	F.4	Public key B_{Pub}	Eq. 15	6118415
	F.5	Shared key K_{shared}	Eq. 16	26560
Decryption	G	Decrypt	No generation	N/A

Table 1: Practical example of the proposed CL-PKE scheme (basic)

5.2 Fully Secure CL-PKE Scheme

Because the proposed CL-PKE scheme is a one-way encryption scheme, its fully secure version can be obtained by applying the same transformations used by Al-Riyami and Paterson , and Boneh and Franklin. These transformations have been proven to provide security strength against IND-CCA2, which is considered to be the highest security test for such a scheme. The fully secure version of the proposed CL-PKE scheme is similar to the basic version, except that in the execution of the Setup algorithm, two additional hash functions, $H_3 \colon \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_p^* \text{ and } H_4 \colon \{0,1\}^n \times \{0,1\}^n$ are used. With the additional hash functions, the encryption and decryption algorithms are as follows:

A. Encryption Algorithm:

1. First checks that $X_A, Y_A, Z_A \in \mathbb{Z}_p^*$ and make sure that $T_{B_{Pri}}(X_A) \bmod p = Y_A$. If so then encrypt the message otherwise output ⊥ and abort the encryption.
2. Set $Q_A = H_1(ID_A)$.
3. Choose random value $\sigma \in \{0,1\}^n$.
4. Set party B's private key, $r = B_{Pri} = H_3(\sigma, M)$.
5. Compute the ciphertext $C = (U, V, W)$, where, $U = B_{Pub} = T_r(G) \bmod p$, $V = \sigma \oplus H_2(K_{shared})$, and $W = M \oplus H_4(\sigma)$. The shared key K_{shared} is generated by Eq. 16.

B. Decryption Algorithm:

1. Compute $\sigma = V \oplus H_2(K_{shared})$. The shared key is generated by Eq. 11.
2. Compute message, $M = W \oplus H_4(\sigma)$.
3. Set $r = B_{Pri} = H_3(\sigma, M)$, and test whether $U = T_r(G) \bmod p$. Accept the message M if equal, otherwise, reject the ciphertext.

6. SECURITY ANALYSIS

The security analysis of the proposed CL-PKE scheme is presented in two parts. The first part shows the strength of the shared secret key and its intractability against attacks. The second part shows the security analysis against the strongest security threat, the IND-CCA2. The analysis is performed by comparing the proposed CL-PKE scheme against the Al-Riyami and Paterson CL-PKE scheme (Al-Riyami and Paterson, 2003).

6.1 The Strength of the Shared Key

The proposed CL-PKE scheme inherits its security strength from the NP-hard Discrete Chebyshev Problem (DCP). Based on the NP-hard problem, we make the following claim:

Claim1: Given all the public parameters $g = \langle p, P_0, G, n, H_1, H_2, H_3, H_4 \rangle$ and party A's public key $P_A = \langle X_A, Y_A, Z_A \rangle$, it is very easy for parties A and B to generate the shared secret key, K_{shared}. However, it is intractable for an adversary and the KGC to generate the shared secret key.

Proof:

1. **Party A (Decryptor):**
Upon request, Party A will receive its partial private key D_A from the KGC through a secure channel and subsequently will generate its secret value $x_A \in \mathbb{Z}_q^*$. It is feasible to generate the shared secret key K_{shared} as shown by Eq. 11. Party A will then fuse its secret value x_A into the partial private key by using Eq. 4.

2. **Party B (Encryptor):**
Party B has its private key B_{Pri}, Party A's public key Z_A and the sender's identity. It is feasible for Part B to generate the shared secret key K_{shared} as shown by Eq. 16. Party B will then fuse its private key B_{Pri} with party A's public key Z_A by using Eq. 4.

3. **The KGC:**
Given all the public parameters $g = \langle p, P_0, G, n, H_1, H_2, H_3, H_4 \rangle$, party A's public key Z_A, and partial private key D_A, and party B's public key B_{Pub}, it is infeasible to generate the shared secret key K_{shared}, due to the DCP hard problem. To generate K_{shared}, the KGC needs the secret value of party A, x_A, and the private key of the party B, B_{Pri}. But both of the values x_A and B_{Pri} are already fused with the public values through the DCP hard problem. x_A is fused with Z_A as shown by Eq. 14, and B_{Pri} is fused with B_{Pub} as indicated by Eq. 17. Therefore, KGC will not be able to generate K_{shared}.

4. **The adversary:**
The information the adversary can acquire is limited to the public values. These public values are the parameters g, party A's public key Y_A, and party B's public key B_{Pub}. Again, due to the fact that all of the secret and private values that needed for the generation of K_{shared} are fused with the public values by the DCP hard problem, it is impossible for the adversary to generate the shared secret key K_{shared}.

6.2 Security Against IND-CCA2

First, consider the comparison between the proposed CL-PKE schemes with Al-Riyami-Paterson CL-PKE scheme as shown in Table 2. IND-CCA is a type of strong security threat that is used to measure the security strength of ID-based schemes. Formally, the IND-CCA can be defined as the ability of an adversary $\mathcal{A}$ to successfully decrypt an intercepted ciphertext with probability $\Pr \geq 1/2$ given that, he has the ability to observe and intercept any ciphertext sent from the encryptor to the decryptor as well as the ability to chose the decryptions of any number of plaintexts associated with their public keys (Al-Riyami and Paterson, 2003; Bellare and Desai, 1998; Dolev *et al.*, 2000; Rackoff and Simon, 1991).

It has been proven that the security proof against IND-CCA for IBE scheme (Algehawi and Samsudin, 2010) is the same as the security proofs of the IBE scheme introduced by Boneh and Franklin in 2003, but each of them relies on a different hard problem. The security proof against IND-CCA for the CL-PKE scheme presented by Al-Riyami and Paterson is an extended version of the security proof of the Boneh-Franklin IBE scheme; both of them use IND-CCA as the measure against which to evaluate security strength.

The proposed CL-PKE scheme and the CL-PKE scheme by Al-Riyami and Paterson follow the same steps in terms of their algorithms. Thus, the same extended version of the game played in Al-Riyami-Paterson CL-PKE scheme to prove its security strength against adversaries of types, $\mathcal{A}_{\text{I}}$ and $\mathcal{A}_{\text{II}}$ can also be played to prove the security strength of the proposed CL-PKE scheme. Therefore, the probability assumption for the proposed CL-PKE scheme can be devised from Theorem 1 (Al-Riyami and Paterson, 2003) as the following:

The new proposed CL-PKE scheme is IND-CCA secure against the two types of adversaries, $\mathcal{A}_{\text{I}}$ and $\mathcal{A}_{\text{II}}$, as explained by Al-Riyami and Paterson.

If there is no polynomially bounded adversary $\mathcal{A}$ of either types I or II with a non-negligible advantage against the challenger in the games played, $\mathcal{A}$ advantage in this game is $\text{Adv}(\mathcal{A}) = 2(\Pr[b = b'] - \frac{1}{2})$, where, $b, b' \in \{0,1\}$, and the adversary wins the game if $b = b'$. From the analysis, we summarize the following:

1. The security proof of Theorem 1 by Al-Riyami and Paterson is devised based on the layout and the steps of the algorithms composing the CL-PKE scheme, regardless of the mathematical foundation which is used to relate the encryption and decryption keys.
2. By comparing our proposed CL-PKE scheme with Al-Riyami-Paterson CL-PKE scheme as shown by Table 2, we find that both schemes have the same algorithms with the same steps

involved but that each of them relies on a different hard problem; that is, our proposed CL-PKE scheme is based on the DCP hard problem as, explained in Subsection 2.2, while the CL-PKE scheme relies on the BDHP hard problem.

3. The probability assumption of the security proof of the CL-PKE scheme is based on its application steps, which are the same as those of the proposed CL-PKE scheme. The difference between these two schemes is only in the underlying cryptography technique. This difference in the underlying technique does not affect the probability ass
4. umptions of the security proof used for the CL-PKE scheme; therefore, the same probability assumption can be used for the proposed CL-PKE scheme.
5. Finally, based on previous findings (Algehawi and Samsudin, 2010; Al-Riyami and Paterson, 2003; Boneh and Franklin, 2003; Maze, 2003), we can conclude that the new proposed CL-PKE scheme is IND-CCA secure against the two types of adversaries, $\mathcal{A}_I$ and $\mathcal{A}_{II}$ as defined by Al-Riyami and Paterson.

Algorithm	Al-Riyami-Paterson CL-PKE scheme	The proposed fully secure CL-PKE scheme based on Chebyshev polynomial
Setup	1. Generate two groups G_1 and G_2 of prime order q and an admissible map $\hat{e}: G_1 \times G_1 \to G_2$. 2. Choose a random generator $P \in G_1$. 3. Choose a random master-key $s \in \mathbb{Z}_q^*$ and calculate the public key $P_0 = sP$. 4. Select the hash functions :$H_3: \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_q^*$, $H_4: \{0,1\}^n \to \{0,1\}^n$, $H_1: \{0,1\}^* \to G_1$, and $H_2: G_2 \to \{0,1\}^n$ for some bit-length n. 5. Choose the message space $\mathcal{M} = \{0,1\}^n$ and the ciphertext space $C = G_1 \times \{0,1\}^n$.	1. Randomly choose a large prime p. 2. Choose a random master-key $s \in \mathbb{Z}_p^*$, where $s \geq 2$ and a global parameter $G \in \mathbb{Z}_p^*$. 3. By using Eq. 4, calculate the public key P_0 as the following: $P_0 = T_s(G) \bmod p$. 4. Select two hash functions $H_1: \{0,1\}^* \to \mathbb{Z}_p^*$, $H_2: \mathbb{Z}_p^* \to \{0,1\}^n$, $H_3: \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_p^*$, and $H_4: \{0,1\}^n \times \{0,1\}^n$ for some bit-length n. 5. Choose the message space $M = \{0,1\}^n$, and ciphertext space $C = \mathbb{Z}_p^* \times \{0,1\}^n$.
Partial-Private-Key-Extract	Generate $D_A = sQ_A \in G_1^*$.	Generate $D_A = T_s\left(T_{Q_A}\left(T_{B_{pri}}(G)\right)\right) (\bmod p)$.
Set-Secret-Value	Selected a random value $x_A \in \mathbb{Z}_q^*$.	Selected a random value $x_A \in \mathbb{Z}_p^*$.
Set-private-Key	Generates A's private key $S_A \in G_1^*$, $S_A = x_A D_A = x_A s Q_A$.	Generate the shared secret key $K_{shared} \in \mathbb{Z}_p^*$, $K_{shared} = T_{x_A}\left(T_s\left(T_{Q_A}\left(T_{B_{pri}}(G)\right)\right)\right) (\bmod p)$.
Set-Public-Key	Generates the A's public key $P_A = \langle X_A, Y_A \rangle$, such that $X_A = x_A P$ and $Y_A = x_A P_0 = x_A s P$.	Generate the tuple, $P_A = \langle X_A, Y_A, Z_A \rangle$, such that: $X_A = T_{x_A}(G) \bmod p$. $Y_A = T_{x_A}(T_{B_{Pri}}(G))(\bmod p)$. $Z_A = T_{x_A}(T_s(G))(\bmod p)$.
Encryption	1. Checks that $X_A, Y_A \in G_1^*$ and $\hat{e}(X_A, P_0) = \hat{e}(Y_A, P)$. 2. Set $Q_A = H_1(ID_A) \in G_1^*$. 3. Chose $\sigma \in \{0,1\}^n$. 4. Set $r = H_3(\sigma, M)$. 5. Set the ciphertext as $C = (U, V, W)$ where $U = rP$, $V = H_2(\hat{e}(Q_A, Y_A)^r)$, and $W = M \oplus H_4(\sigma)$.	1. Checks that $X_A, Y_A, Z_A \in \mathbb{Z}_p^*$ and that $T_{B_{Pri}}(X_A) \bmod p = Y_A$. 2. Set $Q_A = H_1(ID_A) \in \mathbb{Z}_p^*$. 3. Choose $\sigma \in \{0,1\}^n$. 4. Set $r = B_{Pri} = H_3(\sigma, M)$. 5. Set the ciphertext as $C = (U, V, W)$, where, $U = T_r(G) \bmod p$, $V = \sigma \oplus H_2(K_{shared})$, and $W = M \oplus H_4(\sigma)$.
Decryption	1. Calculate $\sigma = V \oplus H_2\left(\hat{e}(U, S_A)\right)$. 2. Calculate $M = W \oplus H_4(\sigma)$. 3. Set $r = H_3(\sigma, M)$. If $U \neq rP$ reject the ciphertext, otherwise accept the decrypted message, M.	1. Compute $\sigma = V \oplus H_2(K_{shared})$. 2. Compute message, $M = W \oplus H_4(\sigma)$. 3. Set $r = B_{Pri} = H_3(\sigma, M)$, if $U \neq T_r(G) \bmod p$ reject the ciphertext, otherwise accept the decrypted message, M.

Table 2: Comparison between the proposed CL-PKE scheme and Al-Riyami-Paterson CL-PKE scheme.

7. CONCLUSION

In this paper, we have proposed a new CL-PKE scheme based on the Chebyshev polynomial extended over $\mathbb{Z}_p$. Our scheme is built to have the same properties as the well-known CL-PKE scheme of Al-Riyami and Paterson. The Discrete Chebyshev Problem (DCP) over finite field $\mathbb{Z}_p$ and the bilinearity property of the extended Chebyshev polynomial have been used to implement the CL-PKE scheme. Our scheme is well-tested and found to be secure, applicable and reliable.

REFERENCES

Algehawi M., and Samsudin A. 2010. A new Identity Based encryption scheme (IBE) Using Chebyshev Polynomial Extended over the Finite Field Z_p, *Phys. Lett. A*, (374), 4670–4674.

Al-Riyami, S. S., and Paterson, K. G. 2003. Certificateless public key cryptography. In *Advances in Cryptology-ASIACRYPT 2003* (pp. 452-473). Springer Berlin Heidelberg..

Amigó, J. M., Kocarev, L., and Szczepanski, J. 2007. Theory and practice of chaotic cryptography. *Physics Letters A*, *366*(3), 211-216.

Bellare, M., Desai, A., Pointcheval, D., and Rogaway, P. 1998. Relations among notions of security for public-key encryption schemes. In*Advances in Cryptology—CRYPTO'98* (pp. 26-45). Springer Berlin Heidelberg.

Bergamo, P., D'Arco, P., De Santis, A., and Kocarev, L. 2005. Security of public-key cryptosystems based on Chebyshev polynomials. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, *52*(7), 1382-1393.

Dayuan, B., and Dahu, W. 2009. A Chaos Public-Key Cryptosystem Based on Semi-Group Features. In *Biomedical Engineering and Informatics, 2009. BMEI'09. 2nd International Conference on* (pp. 1-3). IEEE.

Boneh, D., and Franklin, M. 2001. Identity-based encryption from the Weil pairing. In *Advances in Cryptology—CRYPTO 2001* (pp. 213-229). Springer Berlin Heidelberg.

Dolev D., Dwork C., and Naor M. 2000. Non-malleable cryptography. *SIAM J. of Comp.*. (30): 391-437.

Fujisaki, E., and Okamoto, T. 1999. Secure integration of asymmetric and symmetric encryption schemes. In *Advances in Cryptology—CRYPTO'99*(pp. 537-554). Springer Berlin Heidelberg.

Han, S. 2008. Security of a key agreement protocol based on chaotic maps.*Chaos, Solitons & Fractals*, *38*(3), 764-768.

Maze, G. 2003. *Algebraic methods for constructing one-way trapdoor functions* (Doctoral dissertation, University of Notre Dame).

Rackoff, C., and Simon, D. R. 1992. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Advances in Cryptology—CRYPTO'91* (pp. 433-444). Springer Berlin Heidelberg.

Dahu, W., Haizhu, Y., Fashan, Y., and Xudong, W. 2008. A new key exchange scheme based on Chebyshev polynomials.

Wang, X., and Zhao, J. 2010. An improved key agreement protocol based on chaos. *Communications in Nonlinear Science & Numerical Simulation,15*(12), 4052-4057.

Xiang, T., Wong, K. W., and Liao, X. 2009. On the security of a novel key agreement protocol based on chaotic maps. *Chaos, Solitons & Fractals, 40*(2), 672-675.

Xiao, D., Liao, X., and Deng, S. 2007. A novel key agreement protocol based on chaotic maps. *Information Sciences*, *177*(4), 1136-1142.

Yoon, E. J., and Yoo, K. Y. 2008. A new key agreement protocol based on chaotic maps. In *Agent and Multi-Agent Systems: Technologies & Applications* (pp. 897-906). Springer Berlin Heidelberg.

On the Security of a Strongly Secure Efficient Certificateless Signature Scheme

Rouzbeh Behnia, Swee-Huay Heng, Syh-Yuan Tan
Faculty of Information Science and Technology,
Multimedia University
Email: {rouzbeh.behnia, shheng, sytan}@mmu.edu.my

ABSTRACT

Certificateless cryptography overcomes the private key escrow problem in identity-based cryptography while addressing the costly issues in traditional public key cryptography. Due to their applications, particularly in mobile devices, efficient signature schemes with short signature length have attracted much attention within the cryptography community. In 2012, Tso, Huang and Susilo proposed an efficient certificateless signature scheme. While all the existing short certificateless signature schemes in the literature are only secure against the weakest type of adversary, namely, normal adversary, the new scheme claims to be secure against the strongest type of adversary, namely, super adversary. In this paper, we falsify Tso *et al.*'s claim by mounting an attack on their scheme. We show that upon a public key replacement attack, a Type I strong adversary (which is weaker than a super adversary) is able to forge signatures on behalf of the signer.

Keywords: Certificateless, signatures, bilinear pairing, unforgeability, public key replacement.

1. INTRODUCTION

In order to address the costly issues inherited in traditional public key cryptography, Shamir (1985) proposed the idea of identity-based cryptography. In identity-based systems, the public key of the user is computed from her identifying information that is publicly known (e.g. IP address, email, ID number, etc.). Hence, the need to issue and manage certificates on the authenticity of public keys is completely eliminated. The private key of the user, on the other hand, is computed by a fully trusted third party called the *Private Key Generator* (PKG).

The fact that the PKG has complete control over the users' private keys, introduced the private key escrow problem. In other words, the PKG is able to impersonate all the users in the system. This issue limits the applications of identity-based cryptography only in highly trusted environments where the PKG is completely trusted by the system users (e.g. the PKG is the organisation manager which has ownership over all the information being transferred).

In 2003, Al-Riyami and Paterson introduced the concept of certificateless cryptography. The new system bridges between traditional public key cryptography and identity-based cryptography by tackling the costly issues in the former and the private key escrow problem in the latter at the same time. Certificateless cryptography relies on a semi-trusted third party called the *Key Generation Centre* (KGC). In certificateless paradigms, the KGC only supplies one half of the user's private key (i.e. partial private key) which is computed from her publicly available information. The other half of the user's private key (i.e. secret value) is computed and kept secret by the user herself. Such systems obligate the users to be in charge of computing and publishing (e.g. on a public bulletin) their own public keys.

Efficient signature schemes with short signature length are always the best option to be employed in hand-held or light weight devices with low-computation power which are operating in low-bandwidth communication environments. After the work of Boneh *et al.* (2001), there has been a wide range of research covering a variety of different efficient schemes with additional features (Cha and Cheon, 2003; Hess, 2003; Huang *et al.*, 2007; Katz and Wang, 2003; Yap *et al.*, 2006). Huang *et al.* (2007) proposed a certificateless signature scheme as efficient as the one proffered by Boneh *et al.* (2001). Following Huang *et al.*'s (2007) work, other schemes with different levels of security were proposed to the literature (Du and Wen, 2009; Fan *et al.*, 2009; Tso *et al.*, 2011). Recently, Tso *et al.* (2012) proposed a new efficient certificateless signature scheme and proved its security under a relatively weak assumption. The proposed scheme is claimed to be secure against the strongest adversary type in certificateless systems, namely, super adversary. To the best of our knowledge, Tso *et al.*'s (2012) scheme is the only efficient certificateless signature scheme which is claimed to be secure against such a powerful adversary.

In this paper, we mount a public key replacement attack on Tso *et al.*'s (2012) scheme and show that a strong adversary which is a weaker adversary than the super adversarial model adopted by Tso *et al.* (2012) is able to break the unforgeability of the scheme. Our attack demonstrates that even though the proposed scheme is less efficient than the existing ones in the literature (Huang *et al.*, 2007; Du and Wen, 2009; Fan *et al.*, 2009; Tso *et al.*, 2011), it does not offer a better security.

The rest of this paper is organised as follows. In Section 2, we describe some useful definitions and discuss the adversarial models of certificateless systems. In Section 3, we recall the structure and the security model of Tso *et al.*'s scheme. In Section 4, we present our public key replacement attack. Finally, we conclude our paper in Section 5.

2. PRELIMINARIES

2.1 Bilinear Pairing

We let $\mathbb{G}_1$ be a cyclic group of prime order q with g as its generator, and $\mathbb{G}_2$ be another cyclic group of the same order. Let $a, b \in \mathbb{G}_q^*$ be the randomly selected scalar multiplier. An admissible bilinear pairing $e: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is given which is to satisfy the following properties:

1. Bilinearlity: $e(g^a, g^b) = e(g^b, g^a) = e(g, g)^{ab}$
2. Non-degeneracy: $e(g, g) \neq 1$
3. Easily Computable.

2.2 Certificateless Signature Scheme

Typically, a certificateless undeniable signature scheme consists of five algorithms and two protocols (Du and Wen, 2009; Fan *et al.*, 2009; Tso *et al.*, 2011; Tso *et al.*, 2012) as follows.

Setup: Upon inputting a security parameter k, it produces the KGC's key pair (s, P_{Pub}) where s is the master secret key and P_{Pub} is the corresponding public key. Next, the KGC publishes the public parameters $params$ in the system. For the sake of brevity, we omit $params$ as the input of the rest of the algorithms/protocols.

Partial-private-key-extraction: Upon submitting the user's identity ID, the KGC uses its master secret key s to compute the user's partial private key D_{ID}.

Set-secret-value: Using this algorithm, the user with identity ID computes her secret value x_{ID}.

Set-private-key: This algorithm computes the private key of the user SK_{ID}, given her secret value x_{ID} and partial private key D_{ID}.

Set-public-key: Using this algorithm, the user with identity ID computes her public key PK_{ID}.

Sign: Given a message $m \in \{0,1\}^*$, the user with identity ID uses her private key SK_{ID} to generate a signature σ which is valid for the tuple (m, ID, PK_{ID}).

Verify: Given a message-signature pair (m, σ) and the identity of the signer ID with public key PK_{ID}, this algorithm outputs a decision bit $d \in \{valid, invalid\}$ on the validity or invalidity of the signature.

Adversary Types in Certificateless Signature Schemes

As we mentioned in the previous section, there is no certificate involved in certificateless systems to deliver the authenticity of the users' public keys. Due to this property and following the original proposal of certificateless cryptography by Al-Riyami and Paterson (2003), we always consider two types of adversaries for the security models of certificateless schemes.

- **Type I Adversary:** Type I adversary A_I simulates a normal adversary which has no knowledge on the master secret key. However, due to the aforementioned characteristic of certificateless systems, A_I is allowed to replace the public keys of the users with public keys of his choice.
- **Type II Adversary:** Type II adversary A_{II} simulates a malicious KGC. A_{II} is in possession of the system wide master secret key; therefore, it is able to compute the users' partial private keys. Nonetheless, A_{II} is not permitted to replace the public key of the target user.

Similar to the security models of most of the cryptographic paradigms, in the security model of certificateless signature schemes, the adversary is able to obtain signatures on messages of his choice which are valid for the target user. This is to simulate the fact that in the real world, the adversary may be able to acquire valid signatures from the user by either eavesdropping or acting as a rightful user. However, this assumption may become fairly complicated in the context of certificateless cryptography, considering the fact that the adversary is able to replace the users' public keys with public keys of his choice. Consequently, signatures that the adversary obtains from the user could be valid under the user's original public key or the replaced public key.

In order to clarify the peculiar situation in the adversarial model of certificateless signature schemes, Huang *et al.* (2007) proposed a taxonomy of the potential adversaries based on their capabilities as follows.

- **Normal Type I/II adversary:** Normal adversary that is the weakest type of adversary can query the user's signing oracle for signatures which are valid under the user's original public key.
- **Strong Type I/II adversary:** Strong adversary is able to query the user's signing oracle to obtain valid signatures under the public key that was replaced by him. However, the adversary has to provide the signing oracle with the secret value corresponding to the replaced public key.
- **Super Type I/II adversary:** Super adversary that is the strongest type of adversary is assumed to be able to receive valid signatures under the replaced public key without providing the signing oracle with the corresponding secret value

More precisely, the relationships among various adversary types can be shown as follows.

$$\text{Super} \longrightarrow \text{Strong} \longrightarrow \text{Normal}$$

Therefore, if a cryptosystem is secure against a super adversary, then it is definitely secure against strong and normal adversaries.

2.3 Tso et al.'s (2012) Certificateless Signature Scheme

In this section, we review Tso *et al.*'s (2012) scheme and discuss about its security features. In order to avoid confusion, we use the same notations as in Tso *et al.* (2012).

Setup: The setup algorithm is initiated by the KGC; it takes a security parameter k, and generates groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order $q \geq 2^k$, a generator g of $\mathbb{G}_1$ and an admissible bilinear map$e$: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. It also chooses two cryptographic hash functions: H_0: $\{0,1\}^* \rightarrow \mathbb{G}_1$ and H_1: $\{0,1\}^* \rightarrow \mathbb{G}_1$. Then, it picks $s \in \mathbb{Z}_q$ at random as the master secret key and calculates $P_{Pub} = g^s$ as the corresponding public key. The KGC's public key and the system's public parameters $params$ will be made available to all system users.

$$params = (q, \mathbb{G}_1, \mathbb{G}_2, g, P_{Pub}, H_0, H_1)$$

Partial-private-key-extraction: Given the user's identity ID, the KGC computes her partial private key as $D_{ID} = Q_{ID}^{s} = H_0(ID)^s$, and delivers it to the user in a secure manner.

Set-secret-value: The user with identity ID picks $x_{ID} \in \mathbb{Z}_q$ randomly as her secret value.

Set-private-key: After the user received her partial private key and computed her secret value, she forms her private key as $SK_{ID} = (x_{ID}, D_{ID})$.

Set-public-key: After computing her private key SK_{ID}, the user computes her public key as $PK_{ID} = \left(PK_{(ID,1)}, PK_{(ID,2)}\right) = \left(D_{ID}^{x_{ID}}, Q_{ID}^{x}\right)$.

Sign: In order to issue a signature on message $m \in \{0,1\}^*$, the signer with identity ID computes $\sigma = D_{ID} \cdot H_1(m \parallel ID \parallel PK_{ID})^{x_{ID}^{-1}}$.

Verify: Provided a message-signature pair (m, σ) and the signer's identity and public key pair (ID, PK_{ID}), the verifier checks if $e\left(P_{Pub}, PK_{(ID,2)}\right) = e(g, PK_{(ID,2)})$ and $e\left(\sigma, PK_{(ID,2)}\right) = e(PK_{(ID,1)}H_1(m \parallel ID \parallel PK_{ID}), Q_{ID}))$ hold, if so he outputs valid. Otherwise, he outputs invalid.

In the very first certificateless signature scheme proposed in the literature (Al-Riyami and Paterson, 2003), the authors made an assumption that the adversary is able to receive valid signatures under the replaced public keys (i.e. super adversary). This assumption, however, is too strong and claimed to provide better security assurances. Before Tso *et al.*'s (2012) work, all the efficient certificateless signature schemes were only secure against the normal adversary, i.e. the adversary is not provided with the signatures that are valid under the replaced public keys. More precisely, the security of the schemes would be compromised if the adversary is able to receive valid signatures for the replaced public keys. In Tso *et al.*'s (2012) paper, the authors proposed the first short certificateless signature scheme which is secure under the super adversary attacks and related the security of the proposed scheme against the Type I and Type II adversaries to the hardness of the Computational Diffie-Hellman (CDH) and Inverse Computational Diffie-Hellman (InvCDH) problems, respectively. However, in order to achieve such level of security, the signature verification algorithm of the proposed scheme has two additional expensive pairing evaluations comparing to the existing schemes (Huang *et al.*, 2007; Du and Wen, 2009; Fan *et al.*, 2009; Tso *et al.*, 2011).

3. PUBLIC KEY REPLACEMENT ATTACK

As it was highlighted above, if a certificateless scheme is secure against super adversary, then it is secure against strong and normal adversaries as well. Strong adversary is known to have much less power than the super adversary and the assumption of having such adversary is somehow more realistic. In the proposed scheme by Tso *et al.* (2012), the authors claimed that their scheme is secure against super adversary. In this section, we mount a public key replacement attack to show that the proposed scheme is not even secure against strong adversary.

In order to mount the attack, a strong Type I adversary A_I performs the following steps:

- First, A_I picks $x'_{ID} \in \mathbb{Z}_{q^*}$ at random and forms $PK'_{ID} = \left(PK'_{(ID,1)}, PK'_{(ID,2)}\right) = (P_{Pub}^{x'_{ID}}, g^{x'_{ID}})$.
- Next, it replaces the public key of the signer PK_{ID} with PK'_{ID}, provides x'_{ID} to the signer and queries for a signature on message $m \in \{0,1\}^*$.

- Upon receiving such request, the signer outputs the signature as $\sigma^* = D_{ID} \cdot H_1(m \parallel ID \parallel PK'_{ID})^{x'^{-1}_{ID}}$.

Finally, the adversary $\mathrm{A_I}$ computes $H_1(m \parallel ID \parallel PK'_{ID})^{x'^{-1}_{ID}}$ and extracts the signer's partial private key by computing:

$$D_{ID} = \frac{\sigma^*}{H_1(m \parallel ID \parallel PK'_{ID})^{x'^{-1}_{ID}}}$$

- Consequently, by having knowledge on the signer's partial private key D_{ID}, the adversary can generate a new public key as $PK''_{ID} = \left(PK''_{(ID,1)}, PK''_{(ID,2)}\right) = (D_{ID}^{x'_{ID}}, Q_{ID}^{x'_{ID}})$ and forge signatures on any arbitrary message on behalf of the signer at will.

In Step 1, it is easy to see that the replaced public key $PK'_{ID} = \left(PK'_{(ID,1)}, PK'_{(ID,2)}\right) = (P_{Pub}^{x'_{ID}}, g^{x'_{ID}})$ can pass through the public key verification test since $e\left(P_{Pub}, PK'_{(ID,2)}\right) = e(g, PK'_{(ID,2)})$.

In Step 2, when the adversary replaces the public key of the target signer (with identity ID), then based on the definition of the strong Type I adversary (see Section 2.2), the corresponding secret value should also be presented to the signer's signing oracle. Thus, the output of the oracle would be computed using the secret value that is provided by the adversary.

While the proposed scheme by Tso *et al.* (2012) was shown to be secure against super adversary, the above attack is mounted by a strong adversary which is considered to have much less power than the super adversary (Huang *et al.*, 2007). Therefore, if the scheme is not secure against strong adversary then it definitely cannot be secure against super adversary.

4. CONCLUSION

In this paper, we illustrated that the efficient certificateless signature scheme proposed by Tso *et al.* (2012) is not secure against the strong Type I adversary. Our attack falsifies the authors' claim on the security of their scheme against super adversary. This is due to the fact that strong adversary is known to be much weaker than the super adversary and evidently, if the scheme is not secure against a strong adversary, then, it cannot be secure against a super adversary. Due to the attack presented in this paper, it is worth mentioning that proposing efficient certificateless signature schemes with short signature length which are secure against either strong or super adversary is still an open problem.

5. ACKNOWLEDGEMENT

This research was supported by the FRGS grant (FRGS/2/2013/ICT04/MMU/01/1) and Multimedia University Graduate Research Assistant Scheme.

REFERENCES

Al-Riyami, S. S., and Paterson, K. G. 2003. Certificateless public key cryptography. In *Advances in Cryptology-ASIACRYPT 2003* (pp. 452-473). Springer Berlin Heidelberg.

Boneh, D., Lynn, B., and Shacham, H. 2001. Short signatures from the Weil pairing. In *Advances in Cryptology—ASIACRYPT 2001* (pp. 514-532). Springer Berlin Heidelberg.

Choon, J. C., and Cheon, J. H. 2002. An identity-based signature from gap Diffie-Hellman groups. In *Public key cryptography—PKC 2003* (pp. 18-30). Springer Berlin Heidelberg.

Du, H., and Wen, Q. 2009. Efficient and provably-secure certificateless short signature scheme from bilinear pairings. *Computer Standards & Interfaces,31*(2), 390-394.

Fan, C. I., Hsu, R. H., and Ho, P. H. 2009. Cryptanalysis on Du-Wen certificateless short signature scheme. *Proceedings of JWIS09. Available at http://jwis2009. nsysu. edu. tw/location/paper/Cryptanalysis*.

Hess, F. 2003. Efficient identity based signature schemes based on pairings. In *Selected Areas in Cryptography* (pp. 310-324). Springer Berlin Heidelberg.

Huang, X., Mu, Y., Susilo, W., Wong, D. S., and Wu, W. 2007. Certificateless signature revisited. In *Information Security and Privacy* (pp. 308-322). Springer Berlin Heidelberg.

Katz, J., and Wang, N. 2003. Efficiency improvements for signature schemes with tight security reductions. In *Proceedings of the 10th ACM conference on Computer and communications security* (pp. 155-164). ACM.

Shamir, A. 1985. Identity-based cryptosystems and signature schemes. In *Advances in cryptology* (pp. 47-53). Springer Berlin Heidelberg.

Tso, R., Huang, X., and Susilo, W. 2012. Strongly secure certificateless short signatures. *Journal of Systems and Software*, *85*(6), 1409-1417.

Tso, R., Yi, X., and Huang, X. 2011. Efficient and short certificateless signatures secure against realistic adversaries. *The Journal of Supercomputing*, *55*(2), 173-191.

Yap, W.-S., Heng, S.-H., Goi, B.-M. 2006. An efficient certificateless signature scheme, in: X. Zhou, O. Sokolsky, L. Yan, E.-S. Jung, Z. Shao, Y. Mu, D. Lee, D. Kim, Y.-S. Jeong, C.-Z. Xu (Eds.), *Emerging Directions in Embedded and Ubiquitous Computing*, Vol. 4097 of Lecture Notes in Computer Science, (pp. 322-331). Springer Berlin / Heidelberg.

Group Ring Codes over a Dihedral Group

[1]Zi Shyuan Tan, [2]Miin Huey Ang and [3]Wen Chean Teh
[1,2,3]Pusat Pengajian Sains Matematik, Universiti Sains Malaysia, Minden 11800, Penang, Malaysia
Email: [1]tzs13_mah007@student.usm.my, [2]mathamh@cs.usm.my,
[3]dasmenteh@usm.my

ABSTRACT

A group ring code is a code that can be constructed using group rings. Linear codes have been associated to group rings since 1967. Many existing codes such as cyclic codes and abelian codes are specific examples of group ring codes. This study aims to answer whether there exists a group ring code that can never be a group ring code over a cyclic group. It is conceivable that it has a positive answer. However, our result on group ring codes over the dihedral group D_6 of 6 elements does not support our belief. We found that every binary group ring code over D_6 is a binary group ring code over the cyclic group Z_6 up to equivalent.

Keywords: Group ring code, dihedral group, equivalent code

1. INTRODUCTION

Suppose G is a finite group and R is a ring. Then $RG = \left\{ \sum_{g \in G} a_g g \,\middle|\, a_g \in R \right\}$ is called a group ring over G which has a ring structure as well as a free module structure. Codes that can be constructed using group rings are known as group ring codes. Group ring codes were first discussed by Berman (1967) by associating every cyclic code to a group algebra over a cyclic group and by associating every Reed-Muller code to a group algebra over an elementary abelian 2-group. Two years later, MacWilliams (1969) examined the class of codes associated to group rings over dihedral groups. Charpin (1983) discovered that every extended Reed-Solomon code can be considered as an ideal of some modular group algebras. Well-known classical codes, such as the extended binary Golay code, have been shown to be group ring codes (Landrock and Manz, 1992, McLoughlin and Hurley, 2008).

Hughes (2000) defined a group ring code as an ideal in a group ring. Since then, various studies on group ring codes, such as self-orthogonal group ring codes, checkable group ring codes and etcetera, have also been done in the literature (Fu and Feng, 2009, Jitman *et al.*, 2010, Wong and Ang, 2013, Hurley, 2014). Hurley (2006) discovered the isomorphism between a group ring and a ring of matrices. This result leads to a group ring encoding method for codes which was introduced by Hurley (2009). The group ring codes introduced by Hurley are generally submodules of their corresponding group rings and are only ideals in certain restrictive cases.

Our work is triggered by the curiosity on the existence of a group ring code which can never be a group ring code over a cyclic group. We found that each binary group ring code over the dihedral group D_6 of 6 elements, is always equivalent to a binary group ring code over the cyclic group Z_6. This paper is organized as follows. We give some basic definitions in preliminary section. Next section contains our main result and conclusion is given in the last section.

2. PRELIMINARY

In this paper, our focus is on binary group ring codes, that is, F_2G-codes, where F_2 is the finite field of order 2 and G is a group. Therefore, all the definitions and results given are restricted to the finite field F_2, although some of them are applicable for arbitrary ring R.

Let W be a submodule of F_2G and $\alpha \in F_2G$. A function $f: W \rightarrow F_2G$ such that $f(x) = \alpha x$ is called a *group ring encoding function* (Hurley, 2009). The image of f, denoted as $C(\alpha, W)$, is called an F_2G-*code* with generator α relative to the submodule W. Thus $C(\alpha, W)$ is the set $\{\alpha x | x \in W\}$.

It is pointed out by Hurley (2009) that if a submodule of F_2G has a basis that consists of only group elements, then the corresponding generating matrix and parity check matrix can be constructed easily. Hence, following Hurley's approach, we concern only on F_2G-codes such that the corresponding submodule W are generated by a subset N of G. By abuse of notation, we denote the group ring code by $C(\alpha, N)$ instead of $C(\alpha, W)$.

From now on, for the remaining of this section, fix a group G. Suppose $\{g_1, g_2, \dots, g_n\}$ is a fixed ordering of the elements of G. Every F_2G-code can be identified with a linear code of length n by associating $\sum_{i=1}^{n} a_i g_i$ with a binary string $a_1 a_2 \dots a_n$. Note that if a linear code of length n is identified with a group ring code, then the associated group must be of order n. Recall that two binary linear codes C_1 and C_2 are called *equivalent* if there exist a permutation of coordinates which sends C_1 to C_2 (Huffman and Pless, 2003). Two group ring codes are said to be equivalent if they are associated with two equivalent linear codes.

Definition 2.1. (Hurley, 2009) The F_2G-*matrix* of $\alpha = \sum_{g_i \in G} a_{g_i} g_i \in F_2G$ is defined as the matrix $\left[a_{g_i^{-1} g_j}\right]_{n \times n}$. The *rank* of α is defined as the rank of the F_2G-matrix for α.

Remark 2.2.

(i) The rank of a matrix can be determined by counting the number of nonzero-rows in its reduced row echelon form.

(ii) Suppose $u \in F_2G$. All elements of the form $ux \in F_2G$ where $x \in G$ has the same rank as u.

Definition 2.3. (Hurley, 2009) Suppose $N \subseteq G$. If α is a zero-divisor of F_2G, then $C(\alpha, N)$ is called an F_2G-*zero divisor code*. Similarly, $C(\alpha, N)$ is called an F_2G-*unit derived code* when α is a unit of F_2G.

Note that the elements in F_2G are either zero-divisors or units. For any element $\alpha = \sum_{g \in G} a_g g \in F_2G$, the *support* of α is defined to be the set

$$supp(\alpha) = \{g \in G | a_g \neq 0\}$$

and the *weight* of α is defined as

$$wt(\alpha) = |supp(\alpha)|$$

Lemma 2.4. Suppose $\alpha \in F_2G$ and $x \in G$. For arbitrary $N \subseteq G$, there exists $N' \subseteq G$ such that $C(\alpha x, N') = C(\alpha, N)$. Particularly, an F_2G-code $C(\alpha x, G)$ is the same as the code $C(\alpha, G)$.

Proof. Suppose $N = \{g_{k_1}, g_{k_2}, \dots, g_{k_t}\} \subseteq G$. Then the code

$$C(\alpha, N) = \mathcal{L}_{F_2}\{\alpha g_{k_1}, \alpha g_{k_2}, \dots, \alpha g_{k_t}\}.$$

For each $i = 1,2, \dots, t$, there exists unique $g_{h_i} \in G$ such that $x g_{h_i} = g_{k_i}$. Hence,

$$\begin{aligned} C(\alpha, N) &= \mathcal{L}_{F_2}\{\alpha(x g_{h_1}), \alpha(x g_{h_2}), \dots, \alpha(x g_{h_t})\} \\ &= \mathcal{L}_{F_2}\{\alpha x(g_{h_1}), \alpha x(g_{h_2}), \dots, \alpha x(g_{h_t})\} \\ &= C(\alpha x, N') \end{aligned}$$

where $N' = \{g_{h_1}, g_{h_2}, \dots, g_{h_t}\} \subseteq G$.

Particularly, we have

$$\begin{aligned} C(\alpha x, G) &= \mathcal{L}_{F_2}\{\alpha(xg_1), \alpha(xg_2), \ldots, \alpha(xg_n)\} \\ &= \mathcal{L}_{F_2}\{\alpha g_1, \alpha g_2, \ldots, \alpha g_n\} \\ &= C(\alpha, G). \end{aligned}$$

3. MAIN RESULT

Suppose that $\{1, a, a^2, b, ba, ba^2\}$ is a fixed listing of the elements for the dihedral group $D_6 = \langle a, b | a^3 = b^2 = 1, ba = a^{-1}b \rangle$ and suppose that $\{1, g, g^2, g^3, g^4, g^5\}$ is a fixed listing of the elements for the cyclic group $Z_6 = \langle g | g^6 = 1 \rangle$. First, we look at some examples.

Example 3.1. The element $1 + a + b + ba \in F_2D_6$ is of rank 2 and $1 + g + g^2 + g^4 \in F_2Z_6$ is of rank 5. The code $C(1 + a + b + ba, \{1, a\}) = \mathcal{L}_{F_2}\{1 + a + b + ba, a + a^2 + ba + ba^2\}$ can be identified with the code $C_1 = \mathcal{L}_{F_2}\{110110, 011011\}$ whereas the code$C(1 + g + g^2 + g^4, \{1, g\}) = \mathcal{L}_{F_2}\{1 + g + g^2 + g^4, g + g^2 + g^3 + g^5\}$ can be identified with the code $C_2 = \mathcal{L}_{F_2}\{111010, 011101\}$. We can verify that the codes C_1 and C_2 are equivalent, by some permutation of the digits, which implies the code $C(1 + a + b + ba, \{1, a\})$ is equivalent to the code $C(1 + g + g^2 + g^4, \{1, g\})$.

Example 3.2. Both the elements $1 + a + a^2 + b \in F_2D_6$ and $1 + g \in F_2Z_6$ are of rank 5 but of different weight. It can be shown that $C(1 + a + a^2 + b, D_6)$ and $C(1 + g, Z_6)$ are the same code, which consist of all the F_2D_6 elements of even weight. Hence, they are immediately equivalent.

Given an F_2D_6-code $C(u, N)$ where $N \subseteq D_6$, our aim is to look for a suitable $v \in F_2Z_6$ and $M \subseteq Z_6$ such that the F_2Z_6-code $C(v, M)$ is equivalent to $C(u, N)$. From the examples, we see that $C(u, N)$ can be equivalent to $C(v, M)$ although u and v are of different rank or weight. Says u is of weight w and rank k. In general, any F_2Z_6 element of weight w and of rank greater than or equal to k can act as v. Nevertheless, v with smaller rank may not work; this is due partially to the fact that every group ring code with generator of rank k has dimension at most k (Hurley, 2009). However, for the sake of simplicity, we choose an F_2Z_6 element of weight w and rank k as our v.

We categorise all the elements in F_2D_6 and F_2Z_6 according to their weight and rank in the following table. Let $u \in F_2D_6$ and $v \in F_2Z_6$. By Remark 2.2 (ii), all elements of the form $ux \in F_2D_6$ ($vy \in F_2Z_6$) where $x \in D_6$ ($y \in Z_6$) is of the same rank as u (v).

Wt	F_2D_6 element	F_2Z_6 element	rank
1	$1x, x \in D_6$	$1y, y \in Z_6$	6
2	-	$(1+g)y, y \in Z_6$	5
	$(1+a)x, x \in D_6$	$(1+g^2)y, y \in Z_6$	4
	$(1+b)x, x \in D_6$ $(1+ba)x, x \in D_6$ $(1+ba^2)x, x \in D_6$	$(1+g^3)y, y \in Z_6$	3
3	-	$(1+g+g^3), y, y \in Z_6$ $(1+g+g^4)y, y \in Z_6$	6
	$(1+a+b)x, x \in D_6$ $(1+a+ba)x, x \in D_6$ $(1+a+ba^2)x, x \in D_6$	$(1+g+g^2)y, y \in Z_6$	4
	$(1+a+a^2)x, x \in D_6$	$(1+g^2+g^4)y, y \in Z_6$	2
4	$(1+a+a^2+b)x, x \in D_6$	$(1+g+g^2+g^4)y, y \in Z_6$	5
	-	$(1+g+g^2+g^3)y, y \in Z_6$	4
	$(1+a+b+ba)x, x \in D_6$ $(1+a+b+ba^2)x, x \in D_6$	$(1+g+g^3+g^4)y, y \in Z_6$	2

	$(1+a+ba+ba^2)x, x \in D_6$		
5	$(1+a+a^2+b+ba)x, x \in D_6$	$(1+g+g^2+g^3+g^4)y, y \in Z_6$	6

Table 1: Categorization of F_2D_6 elements and F_2Z_6 elements

From table 1, we see that for every element $u \in F_2D_6$, there exists an F_2Z_6 element v such that $wt(v) = wt(u)$ and $rank\ (v) = rank\ (u)$. Now, we are ready to show that every F_2D_6-code is a F_2Z_6-code up to equivalent. Our next example illustrate that for a special example.

Example 3.3. Consider the F_2D_6-codes $C(1+ba, N)$ where N is an arbitrary subset of D_6. Note that $wt(1+ba) = 2$ and $rank(1+ba) = 3$. From table 1, the element $1+g^3 \in F_2Z_6$ is of the same weight and same rank as $1+ba$. Hence, this will be our choice of generator for the equivalent group ring code over Z_6.

Recall that a codeword of the form $\sum_{i=0}^{2} \alpha_i a^i + \beta_i ba^i \in F_2D_6$ is identified with the binary codeword $\alpha_0\alpha_1\alpha_2\beta_0\beta_1\beta_2$, whereas a codeword of the form $\sum_{i=0}^{5} \omega_i g^i \in F_2Z_6$ is identified with the binary codeword $\omega_0\omega_1\omega_2\omega_3\omega_4\omega_5$. Additionally, $C(1+ba, N)$ has a spanning set $\{(1+ba)x | x \in N\}$. Similar is true for $C(1+g^3, M)$.

By comparing the two tables below, we can choose M easily such that $C(1+ba, N)$ and $C(1+g^3, M)$ are equivalent, to be described below.

$x \in D_6$	$(1+ba)x$	Binary representation
1	$1+ba$	100010
a	$a+ba^2$	010001
a^2	a^2+b	001100
b	a^2+b	001100
ba	$1+ba$	100010
ba^2	$a+ba^2$	010001

$y \in Z_6$	$(1+g^3)y$	Binary representation
1	$1+g^3$	100100
g^2	g^2+g^5	001001
g^4	$g+g^4$	010010
g^3	$1+g^3$	100100
g	$g+g^4$	010010
g^5	g^2+g^5	001001

By the permutation $\begin{pmatrix}1&2&3&4&5&6\\1&2&3&6&4&5\end{pmatrix}$ on the digits, notice that the above two sets of binary codewords are the same. Hence, for every N subset of D_6, we can easily find M such that $C(1+g^3, M)$ is equivalent to $C(1+ba, N)$. For example, let ϕ be the bijection defined by $\phi(1) = 1, \phi(a) = g^4, \phi(a^2) = g^2, \phi(b) = g^5, \phi(ba) = g^3, \phi(ba^2) = g$, then we can take M to be $\phi(N)$. Note that the permutation $\begin{pmatrix}1&2&3&4&5&6\\1&2&3&6&4&5\end{pmatrix}$ can be abbreviated as $(4\ 6\ 5)$, as it is commonly done in group theory.

Table 2 shows the corresponding permutation that works in the way as described in Example 3.3 for other key values of u.

$u \in F_2D_6$	$v \in F_2Z_6$	Permutation on coordinates
1	1	e
$1+a$	$1+g^2$	$(2\ 3\ 5)(4\ 6)$
$1+b$	$1+g^3$	e
$1+ba$	$1+g^3$	$(4\ 6\ 5)$
$1+ba^2$	$1+g^3$	$(4\ 6\ 5)$
$1+a+a^2$	$1+g^2+g^4$	$(2\ 3\ 5\ 4)$
$1+a+b$	$1+g+g^2$	$(2\ 3\ 5\ 4)$
$1+a+ba$	$1+g+g^2$	$(2\ 3\ 5)(4\ 6)$
$1+a+ba^2$	$1+g+g^2$	$(2\ 3\ 5\ 6)$
$1+a+a^2+b$	$1+g+g^2+g^4$	$(2\ 3\ 5\ 4)$
$1+a+b+ba$	$1+g+g^2+g^4$	e

$1+a+ba+ba^2$	$1+g+g^2+g^4$	$(4\ 6\ 5)$
$1+a+b+ba^2$	$1+g+g^2+g^4$	$(4\ 5\ 6)$
$1+a+a^2+b+ba$	$1+g+g^2+g^3+g^4$	e

Table 2: Permutations that sends $C(u, D_6)$ to $C(v, Z_6)$

Theorem 3.4. Every F_2D_6-code is a F_2Z_6-code up to equivalent.

Proof. Suppose $u' \in F_2D_6$ and $N' \subseteq D_6$. Consider the F_2D_6-code $C(u', N')$. If u' is one of the listed element in table 2, then we choose the F_2Z_6 element in the same row as our v. By the corresponding permutation, we can always find an appropriate $M \subseteq Z_6$ (as in example 3.3) such that the code $C(v, M)$ is equivalent to the code $C(u', N')$.

Otherwise, $u' = ux$ for some u, an element listed in table 2, and $x \in D_6$. By lemma 2.4, $C(u', N')$ is equivalent to a code $C(u, N)$ where $N \subseteq D_6$. Since $C(u, N)$ is equivalent to some code $\mathrm{C}(v, M)$, by transitivity of equivalence relation, $C(u', N')$ is equivalent to $C(v, M)$ too.

4. CONCLUSION

In this paper, we have seen that every F_2D_6-code can always be expressed as an F_2Z_6-code up to equivalent by using a suitable generator and an appropriate submodule. However we do not know yet whether the converse is true. We do know the existence of an F_2Z_4-code that can never be an F_2D_4-code. The existence of a group ring code that can never be a group ring code over a cyclic group remains as an open problem. To address this problem, we have been working on group ring codes over groups of order 8 as well but so far we have not reach a conclusion yet. On the other hand, our work led us to believe that every group ring code over the dihedral group D_{2n} is equivalent to a group ring code over the abelian group $Z_2 \times Z_n$.

5. ACKNOWLEDGEMENTS

The authors wish to thank the School of Mathematical Science of Universiti Sains Malaysia, particularly the dean, *Prof. Ahmad Izani Md. Ismail*, for providing necessary financial assistance for attending this conference. This work is part of the master research of the first author, Tan, under the supervision of the other authors.

REFERENCES

Berman, S. D. 1967. On the Theory of Group Codes. *Kibernetika*. **3**(1): 31-39.

Charpin, P. 1983. The Extended Reed-Solomon Codes Considered as Ideals of a Modular Group Algebra. *Annals of Discrete Math*. **17**: 171–176.

Fu, W. and Feng, T. 2009. On Self-orthogonal Group Ring Codes. *Designs, Codes and Cryptography*. **50**(2): 203-214.

Huffman, W. C. and Pless, V. 2003. *Fundamental of Error Correcting codes.* Cambridge: Cambridge University Press.

Hughes, G. 2000. Constacyclic Codes, Cocycles and a u+v|u-v Construction. *IEEE Transactions in Information Theory*. **46**(2): 674-680.

Hurley, B. and Hurley, T. 2014. Paraunitary Matrices and Group Rings. *International Journal of Group Theory*. **3**(1): 31-56.

Hurley, P. and Hurley, T. 2009. Codes from Zero-divisors and Units in Group Rings. *Int. J. Information and Coding Theory*. **1**(1): 57-87.

Hurley, T. 2006. Group Rings and Rings of Matrices. *International Journal of Pure and Applied Mathematics*. **31**(3): 319-335.

Jitman, S., Ling, S., Liu, H. and Xie, X. 2010. Checkable Codes from Group Rings. arXiv:1012.5498.

MacWilliams, F. J. 1969. Codes and Ideals in Group Algebras. *Combinatorial Mathematics and its Applications*. 312-328.

McLoughlin, I. and Hurley, T. 2008. A Group Ring Construction of the Extended Binary Golay Code. *IEEE Transactions in Information Theory*. **54**(9): 4381-4383.

Wong, Denis C.K. and Ang, M.H. 2013. Group Algebra Codes Defined over Extra Special p-group. *JP Journal of Algebra, Number Theory and Applications*. **78**(1): 19-27.

Harmonic Analysis and A Bentness-Like Notion in Certain Finite Abelian Groups over Some Finite Fields

[1]Laurent Poinsot and [2*]Nadia El Mrabet
[1]*University Paris 13, Sorbonne Paris Cité, LIPN, CNRS (UMR 7030), France,*
[2]*University Paris 8, LIASD, France*
Email: [1] laurent.poinsot@lipn.univ-paris13.fr, [2]elmrabet@ai.univ-paris8.fr
Website: [1] http://lipn.univ-paris13.fr/poinsot/, [2]http://www.ai.univ-paris8.fr/elmrabet/

ABSTRACT

It is well-known that degree two finite field extensions can be equipped with a Hermitian-like structure similar to the extension of the complex field over the reals. In this contribution, using this structure, we develop a modular character theory and the appropriate Fourier transform for some particular kind of finite Abelian groups. Moreover we introduce the notion of bent functions for finite field valued functions rather than usual complex-valued functions, and we study several of their properties

Keywords: Finite Abelian groups, characters, Hermitian spaces, Fourier transform, bent functions.

1. INTRODUCTION

The most simple Hermitian structure is obtained from the degree two field extension of the complex numbers over the real numbers. It has many applications and in particular provides the usual theory of characters for finite Abelian groups and the existence of an associated Fourier transform. Given a degree two extension $\mathrm{GF}(p^{2n})$ of $\mathrm{GF}(p^n)$, the Galois field with p^n elements where p is a prime number, we can also define a "conjugate" and thus a Hermitian structure on $\mathrm{GF}(p^{2n})$ in a way similar to the relation $\mathbb{C}/\mathbb{R}$. In particular this makes possible the definition of a unit circle $\mathcal{S}(\mathrm{GF}(p^{2n}))$ which is a cyclic group of order p^n+1, subgroup of the multiplicative group $\mathrm{GF}(p^{2n})^*$ of invertible elements. The analogy with $\mathbb{C}/\mathbb{R}$ is extended in this paper by the definition of $\mathrm{GF}(p^{2n})$-valued characters of finite Abelian groups G as group homomorphisms from G to $\mathcal{S}(\mathrm{GF}(p^{2n}))$. But $\mathcal{S}(\mathrm{GF}(p^{2n}))$ does obviously not contain a copy of each cyclic group. Nevertheless if d divides p^n+1, then the cyclic group $\mathbb{Z}_d$ of modulo d integers embeds as a subgroup of this particular unit circle. It forces our modular theory of characters to be applied only to direct products of cyclic groups whose order d_i divides p^n+1. In addition we prove that these modular characters form an orthogonal basis (by respect to the Hermitian-like structure $\mathrm{GF}(p^{2n})$ over $\mathrm{GF}(p^n)$). This decisive property makes it possible the definition of an appropriate notion of Fourier transform for $\mathrm{GF}(p^{2n})$-valued functions, rather than $\mathbb{C}$-valued ones, defined on G, as their decompositions in the dual basis of characters. In this contribution we largely investigate several properties of this modular version of the Fourier transform similar to classical ones. As an illustration of our theory of modular characters one introduces and studies the corresponding cryptographic notion of bent functions in this setting.

2. CHARACTER THEORY: THE CLASSICAL APPROACH

In this paper G always denotes a finite Abelian group (in additive representation), 0_G is its identity element. Moreover for all groups H, H^* is the set obtained from H by removing its identity. As usual $\mathbb{N}^* = \mathbb{N}\backslash\{\,0\,\}$.

The *characters* are the group homomorphisms from a finite Abelian group G to the unit circle $\mathcal{S}(\mathbb{C})$ of the complex field. The set of all such characters of G together with point-wise multiplication is denoted by $\widehat{G}$ and called the *dual group of* G. A classical result claims that G and its dual are isomorphic (essentially because $\mathcal{S}(\mathbb{C})$ contains an isomorphic copy of all cyclic groups). The image in $\widehat{G}$ of $\alpha \in G$ by such an isomorphism is denoted by χ_α. The complex vector space $\mathbb{C}^G$ of complex-valued functions defined on G can be equipped with an inner product defined for $f, g \in \mathbb{C}^G$ by

$$\langle f, g\rangle = \sum_{x\in G} f(x)\overline{g(x)} \tag{1}$$

where $\overline{z}$ denotes the complex conjugate of $z \in \mathbb{C}$. With respect to this Hermitian structure, $\widehat{G}$ is an orthogonal basis, *i.e.*

$$\langle \chi_\alpha, \chi_\beta\rangle = \begin{cases} 0 & \text{if } \alpha \neq \beta, \\ |G| & \text{if } \alpha = \beta \end{cases} \tag{2}$$

for $\alpha, \beta \in G^2$. We observe that in particular (replacing β by 0_G),

$$\sum_{x\in G} \chi_\alpha(x) = \begin{cases} 0 & \text{if } \alpha \neq 0_G, \\ |G| & \text{if } \alpha = 0_G. \end{cases} \tag{3}$$

Definition 1. Let G be a finite Abelian group and $f: G \to \mathbb{C}$. The Fourier transform of f is defined as

$$\begin{array}{rccl} \hat{f}: & G & \to & \mathbb{C} \\ & \alpha & \mapsto & \sum_{x\in G} f(x)\chi_\alpha(x) \ . \end{array} \tag{4}$$

The Fourier transform of a function f is its decomposition in the basis $\widehat{G}$. This transform is invertible and one has an *inversion formula* for f,

$$f(x) = \frac{1}{|G|}\sum_{\alpha\in G} \hat{f}(\alpha)\overline{\chi_\alpha(x)} \tag{5}$$

for each $x \in G$. More precisely the Fourier transform is an algebra isomorphism from $(\mathbb{C}^G, *)$ to $(\mathbb{C}^G, .)$ where the symbol "." denotes the point-wise multiplication of functions, and $*$ is the convolution product defined by

$$\begin{array}{rccl} f * g: & G & \to & \mathbb{C} \\ & \alpha & \mapsto & \sum_{x\in G} f(x)g(-x+\alpha) \end{array} \tag{6}$$

Since the Fourier transform is an isomorphism between the two algebras, the *trivialization of the convolution product* holds for each $(f, g) \in (\mathbb{C}^G)^2$ and each $\alpha \in G$, *i.e.*,

$$(\widehat{f * g})(\alpha) = \hat{f}(\alpha)\hat{g}(\alpha) \ . \tag{7}$$

Proposition 1. Let G be a finite Abelian group and $f, g \in \mathbb{C}^G$. We have

$$\sum_{x\in G} f(x)\overline{g(x)} = \frac{1}{|G|}\sum_{\alpha\in G} \hat{f}(\alpha)\overline{\hat{g}(\alpha)} \quad \text{(Plancherel formula)}, \tag{8}$$

$$\sum_{x\in G} |f(x)|^2 = \frac{1}{|G|}\sum_{\alpha\in G} |\hat{f}(\alpha)|^2 \quad \text{(Parseval equation)} \tag{9}$$

where $|z|$ is the complex modulus of $z \in \mathbb{C}$.

3. HERMITIAN STRUCTURE OVER FINITE FIELDS

In this section we recall some results about an Hermitian structure in some kinds of finite fields. This section is directly inspired from (Dobbertin, et al., 2006) of which we follow the notations, and is generalized to any characteristic p.

Let p be a given prime number and q an even power of p, i.e., there is $n \in \mathbb{N}^*$ such that $q = p^{2n}$, and in particular q is a square.

Assumption 1. *From now on the parameters p, n, q are fixed as introduced above.*

As usual $\mathrm{GF}(q)$ is the finite field of characteristic p with q elements and by construction $\mathrm{GF}(\sqrt{q})$ is a subfield of $\mathrm{GF}(q)$. The field $\mathrm{GF}(q)$, as an extension of degree 2 of $\mathrm{GF}(\sqrt{q})$, is also a vector space of dimension 2 over $\mathrm{GF}(\sqrt{q})$. This situation is similar to the one of $\mathbb{C}$ and $\mathbb{R}$. As $\mathrm{GF}(q)$ plays the role of $\mathbb{C}$, the Hermitian structure should be provided for it. Again according to the analogy $\mathbb{C}/\mathbb{R}$, we then need to determine a corresponding conjugate. In order to do this we use the Frobenius automorphism Frob of $\mathrm{GF}(q)$

$$\begin{array}{llll} \mathrm{Frob}\colon & \mathrm{GF}(q) & \to & \mathrm{GF}(q) \\ & x & \mapsto & x^p \end{array} \tag{10}$$

and one of its powers

$$\begin{array}{llll} \mathrm{Frob}_k\colon & \mathrm{GF}(q) & \to & \mathrm{GF}(q) \\ & x & \mapsto & x^{p^k} \ . \end{array} \tag{11}$$

In particular $\mathrm{Frob}_1 = \mathrm{Frob}$.

Definition 2. *The conjugate of $x \in \mathrm{GF}(q)$ over $\mathrm{GF}(\sqrt{q})$ is denoted by $\overline{x}$ and defined as*

$$\overline{x} = \mathrm{Frob}_n(x) = x^{p^n} = x^{\sqrt{q}} \ . \tag{12}$$

In particular, for every $n \in \mathbb{Z}$, $\overline{n1} = n1$. The field extension $\mathrm{GF}(q)/\mathrm{GF}(\sqrt{q})$ has amazing similarities with the extension $\mathbb{C}$ over the real numbers in particular regarding the conjugate.

Proposition 2. *Let $x_1, x_2 \in \mathrm{GF}(q)^2$, then*

$$\overline{x_1 + x_2} = \overline{x_1} + \overline{x_2}, \qquad \overline{-x_1} = -\overline{x_1}, \overline{x_1 x_2} = \overline{x_1}\ \overline{x_2}, \qquad \overline{\overline{x_1}} = x_1$$

Proof. The three first equalities come from the fact that Frob_n is a field homomorphism of $\mathrm{GF}(q)$. The last point holds since for each $x \in \mathrm{GF}(q)$, $x^q = x$.

The *relative norm with respect to* $\mathrm{GF}(q)/\mathrm{GF}(\sqrt{q})$ is defined as

$$\mathrm{norm}(x) = x\overline{x} \tag{13}$$

for $x \in \mathrm{GF}(q)$, and it maps $\mathrm{GF}(q)$ to $\mathrm{GF}(\sqrt{q})$. We observe that $\mathrm{norm}(x) \in \mathrm{GF}(\sqrt{q})$ because $\sqrt{q} + 1$ divides $q - 1$, and $\mathrm{norm}(x) = 0$ if, and only if, $x = 0$. The *unit circle* of $\mathrm{GF}(q)$ is defined as the set

$$\mathcal{S}(\mathrm{GF}(q)) = \{\, x \in \mathrm{GF}(q) : x\overline{x} = 1 \,\} \tag{14}$$

of all elements having relative norm 1. By construction $\mathcal{S}(\mathrm{GF}(q))$ is the group of $(\sqrt{q}+1)$-th roots of unity, and therefore it is a (multiplicative) cyclic group of order $\sqrt{q}+1$ since $\mathrm{GF}(q)^*$ is cyclic and $\sqrt{q}+1$ divides $q-1$. In what follows, $\mathcal{S}(\mathrm{GF}(q))$ will play exactly the same role as $\mathcal{S}(\mathbb{C})$ in the classical theory of characters.

4. CHARACTERS OVER A FINITE FIELD

Before beginning some formal developments, one should warn the reader on the limitations of the expected character theory in finite fields. We claimed that $\mathcal{S}(\mathrm{GF}(q))$ is a cyclic group of order $\sqrt{q}+1$. Then for each nonzero integer d that divides $\sqrt{q}+1$, there is a (cyclic) subgroup of $\mathcal{S}(\mathrm{GF}(q))$ of order d, and this is the unique kind of subgroups. As a character theory is essentially used to faithfully represent an abstract group as an isomorphic group of functions, a copy of such group must be contained in the corresponding unit circle. Then our character theory in $\mathrm{GF}(q)$ will only apply on groups for which all their factors in a representation as a product direct group of cyclic subgroups have orders that divide $\sqrt{q}+1$.

Assumption 2. *From now on d always denotes an element of $\mathbb{N}^*$ that divides $\sqrt{q}+1$.*

Definiton 3. (*and proposition*) *The (cyclic) subgroup of $\mathcal{S}(\mathrm{GF}(q))$ of order d is denoted by $\mathcal{S}_d(\mathrm{GF}(q))$. In particular, $\mathcal{S}(\mathrm{GF}(q)) = \mathcal{S}_{\sqrt{q}+1}(\mathrm{GF}(q))$. If u is a generator of $\mathcal{S}(\mathrm{GF}(q))$ then $u^{\frac{\sqrt{q}+1}{d}}$ is a generator of $\mathcal{S}_d(\mathrm{GF}(q))$.*
A *character* of a finite Abelian group G *with respect to* $\mathrm{GF}(q)$ (or simply a *character*) is a group homomorphism from G to $\mathcal{S}(\mathrm{GF}(q))$. Since a character χ is $\mathcal{S}(\mathrm{GF}(q))$-valued, $\chi(-x) = (\chi(x))^{-1} = \overline{\chi(x)}$, $\mathrm{norm}(\chi(x)) = 1$ and $\chi(0_G) = 1$ for each $x \in G$. By analogy with the traditional version, we denote by $\hat{G}$ the set of all characters of G that we call its *dual*. When equipped with the point-wise multiplication, $\hat{G}$ is a finite Abelian group. One recall that this multiplication is defined as

$$\forall \chi, \chi' \in \hat{G},\ \chi\chi' : x \mapsto \chi(x)\chi'(x). \tag{15}$$

As already mentioned in introduction, we focus on a very special kind of finite Abelian groups: the additive group of modulo d integers $\mathbb{Z}_\mathrm{d}$ which is identified with the subset $\{0, \dots, d-1\}$ of $\mathbb{Z}$.

Theorem 1. *The groups $\mathbb{Z}_d$ and $\widehat{\mathbb{Z}_d}$ are isomorphic.*
Proof. The parameter d has been chosen so that it divides $\sqrt{q}+1$. Then there is a unique (cyclic) subgroup $\mathcal{S}_d(\mathrm{GF}(q))$ of $\mathcal{S}(\mathrm{GF}(q))$ of order d. Let u_d be a generator of this group. Then the elements of $\widehat{\mathbb{Z}_d}$ have the form, for $j \in \mathbb{Z}_d$,

$$\chi_j : \begin{cases} \mathbb{Z}_d & \rightarrow \quad \mathcal{S}_d(\mathrm{GF}(q)) \\ k & \mapsto \quad (u_d^j)^k = u_d^{jk}. \end{cases} \tag{16}$$

Actually the characters are $\mathcal{S}_d(\mathrm{GF}(q))$-valued since for each $x \in \mathbb{Z}_d$ and each character χ, $\chi(x) \in \mathcal{S}(\mathrm{GF}(q))$ by definition, and satisfies $1 = \chi(0) = \chi(dx) = (\chi(x))^d$ and then $\chi(x)$ is a d-th root of the unity. Then to determine a character $\chi \in \widehat{\mathbb{Z}_d}$, we need to compute the value of $\chi(k) = \chi(k1)$ for $k \in \{0, \dots, d-1\}$, which gives

$$\chi(k) = u_d^{jk}. \tag{17}$$

In this equality, we have denoted $\chi(1)$ by u_d^j for $j \in \{0, \dots, d-1\}$ since $\chi(1)$ is a d-th root of the unity in $\mathcal{S}(\mathrm{GF}(q))$. Then the character χ belongs to $\{\chi_0, \dots, \chi_{d-1}\}$. Conversely, we observe that for $j \in \{1, \dots, d-1\}$, the maps χ_j are group homomorphisms from $\mathbb{Z}_d$ to $\mathcal{S}(\mathrm{GF}(q))$ so they are elements of $\widehat{\mathbb{Z}_d}$. Let us define the following function.

$$\begin{array}{cccc} \Psi\colon & \mathbb{Z}_d & \to & \widehat{\mathbb{Z}_d} \\ & j & \mapsto & \chi_j \ . \end{array} \tag{18}$$

We have already seen that it is onto. Moreover, it is also one-to-one (it is sufficient to evaluate $\chi_j = \Psi(j)$ at 1) and it is obviously a group homomorphism. It is then an isomorphism, so that $\widehat{\mathbb{Z}_d}$ is isomorphic to $\mathbb{Z}_d$.

Proposition 3. $\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}$ and $(\widehat{\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}})$ *are isomorphic*.
Proof. The proof is easy since it is sufficient to remark that $(\widehat{\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}})$ and $\widehat{\mathbb{Z}_{d_1}} \times \widehat{\mathbb{Z}_{d_2}}$ are isomorphic. We recall that d_1 and d_2 are both assumed to divide $\sqrt{q}+1$, thus $\widehat{\mathbb{Z}_{d_1}}$ and $\widehat{\mathbb{Z}_{d_2}}$ exist and are isomorphic to $\mathbb{Z}_{d_1}$ and $\mathbb{Z}_{d_2}$ respectively. Let i_1 be the first canonical injection of $\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}$ and i_2 the second (when $\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}$ is seen as a direct sum). The following map

$$\Phi\colon \begin{cases} (\widehat{\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}}) & \to \quad \widehat{\mathbb{Z}_{d_1}} \times \widehat{\mathbb{Z}_{d_2}} \\ \chi & \mapsto \quad (\chi \circ i_1, \chi \circ i_2) \end{cases} \tag{19}$$

is a group isomorphism. It is obviously one-to-one and for $(\chi_1, \chi_2) \in \widehat{\mathbb{Z}_{d_1}} \times \widehat{\mathbb{Z}_{d_2}}$, the map $\chi\colon (x_1, x_2) \mapsto \chi_1(x_1)\chi_2(x_2)$ is an element of $(\widehat{\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}})$ and $\Phi(\chi) = (\chi_1, \chi_2)$. Then $(\widehat{\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}})$ is isomorphic to $\mathbb{Z}_{d_1} \times \mathbb{Z}_{d_2}$ since $\widehat{Z_{d_\iota}}$ and $\mathbb{Z}_{d_i}$ are isomorphic (for $i = 1{,}2$).

From proposition 3 it follows in particular that $\widehat{\mathbb{Z}_d^m}$ is isomorphic to $\mathbb{Z}_d^m$. This result also provides a specific form to the characters of $\mathbb{Z}_d^m$ as follows. We define a dot product, which is a $\mathbb{Z}_\mathrm{d}$-bilinear map from $(\mathbb{Z}_d^m)^2$ to $\mathbb{Z}_d$, by

$$x \cdot y = \sum_{i=1}^{m} x_i y_i \in \mathbb{Z}_d \tag{20}$$

for $x, y \in \mathbb{Z}_d^m$. Then the character that corresponds to $\alpha \in \mathbb{Z}_d^m$ can be defined by

$$\begin{array}{cccc} \chi_\alpha\colon & \mathbb{Z}_d^m & \to & \mathcal{S}_d(\mathrm{GF}(q)) \\ & x & \mapsto & u_d^{\alpha \cdot x} \end{array} \tag{21}$$

where u_d is a generator of $\mathcal{S}_d(\mathrm{GF}(q))$. In particular for each $\alpha, x \in \mathbb{Z}_d^m$, $\chi_\alpha(x) = \chi_x(\alpha)$. The following result is obvious.

Corollary 1. *Let* $G \cong \prod_{i=1}^{N} \mathbb{Z}_{d_i}^{m_i}$ *be a finite Abelian group for which each integer* d_i *divides* $\sqrt{q}+1$. *Then* G *and* $\hat{G}$ *are isomorphic*.
If $G = \prod_{i=1}^{N} \mathbb{Z}_{d_i}^{m_i}$ satisfies the assumption of the corollary 1, then for $\alpha = (\alpha_1, \dots, \alpha_N) \in G$ one has

$$\chi_\alpha\colon G \ \to \mathcal{S}(\mathrm{GF}(q))$$

$$x = (x_1, \dots, x_N) \quad \mapsto \prod_{i=1}^{N} u_{d_i}^{\alpha_i \cdot x_i} \tag{22}$$

where for each $i \in \{1, \dots, N\}$, u_{d_i} is a generator of $\mathcal{S}_{d_i}(\mathrm{GF}(q))$. In particular for each $\alpha, x \in G^2$, we also have $\chi_\alpha(x) = \chi_x(\alpha)$.

Assumption 3. *From now on, each finite Abelian group G considered is assumed to be of a specific form $\prod_{i=1}^{N} \mathbb{Z}_{d_i}^{m_i}$ where for each $i \in \{1, \dots, N\}$, d_i divides $\sqrt{q} + 1$, so that we have at our disposal a specific isomorphism given by the formula (22) between G and $\hat{G}$.*

The dual $\hat{G}$ of G is constructed and is shown to be isomorphic to G. We may also be interested into the bidual $\hat{\hat{G}}$ of G, namely the dual of $\hat{G}$. Similarly to the usual situation of complex-valued characters, we prove that G and its bidual are canonically isomorphic. It is already clear that $G \cong \hat{\hat{G}}$ (because $G \cong \hat{G}$ and $\hat{G} \cong \hat{\hat{G}}$). But this isomorphism is far from being canonical since it depends on a decomposition of G, and of $\hat{G}$, and choices for generators of each cyclic factor in the given decomposition. We observe that the map $e: G \to \hat{\hat{G}}$ defined by $e(x)(\chi) = \chi(x)$ for every $x \in G, \chi \in \hat{G}$ is a group homomorphism. To prove that it is an isomorphism it suffices to check that e is one-to-one (since G and $\hat{\hat{G}}$ have the same order). Let $x \in \ker(e)$. Then, for all $\chi \in \hat{G}, \chi(x) = 1$. Let us fix an isomorphism $\alpha \in G \to \chi_\alpha \in \hat{G}$ as in the formula (22). Then, for every $\alpha \in G, \chi_\alpha(x) = 1 = \chi_x(\alpha)$ so that $x = 0_G$. Thus we have obtained an appropriate version of Pontryagin-van Kampen duality (see (Hewitt & Ross, 1994)). Let us recall that according to the structure theorem of finite Abelian groups, for any finite Abelian group G, there is a unique finite sequence of positive integers, called the invariants of G, $d_1, \cdots, d_{\ell_G}$ such that d_i divides d_{i+1} for each $i < \ell_G$. Let us denote by $Ab_{\sqrt{q}+1}$ the category of all finite Abelian groups G such that d_{ℓ_G} divides $\sqrt{q} + 1$, with usual homomorphisms of groups as arrows. From the previous results, if G is an object of $Ab_{\sqrt{q}+1}$, then $G \cong \hat{G}$. Moreover, $\widehat{(\cdot)}$ defines a contravariant functor (see (McLane, 1998)) from $Ab_{\sqrt{q}+1}$ to itself. Indeed, if $\phi: G \to H$ is a homomorphism of groups (where G, H belongs to $Ab_{\sqrt{q}+1}$), then $\hat{\phi}: \hat{H} \to \hat{G}$ defined by $\hat{\phi}(\chi) = \chi \circ \phi$ is a homomorphism of groups. Then, we have the following duality theorem.

Theorem 2 (Duality). *The covariant (endo-)functor $\widehat{\widehat{(\cdot)}}: Ab_{\sqrt{q}+1} \to Ab_{\sqrt{q}+1}$ is a (functorial) isomorphism (this means in particular that $G \cong \hat{\hat{G}}$).*

5. ORTHOGONALITY RELATIONS

The characters satisfy a certain kind of orthogonality relation. In order to establish it we introduce the natural "action" of $\mathbb{Z}$ on any finite field $\mathrm{GF}(p^l)$ of characteristic p as $kx = \underbrace{x + \cdots + x}_{k \text{ times}}$ for $(k, x) \in \mathbb{Z} \times \mathrm{GF}(p^l)$. This is nothing else than the fact that the underlying Abelian group structure of $\mathrm{GF}(p^l)$ is a $\mathbb{Z}$-module. In particular one has for each $(k, k', x) \in \mathbb{Z} \times \mathbb{Z} \times \mathrm{GF}(p^l)$,

1. $0x = 0$, $1x = x$ and $k0 = 0$,
2. $(k + k')x = kx + k'x$ and then $nkx = n(kx)$,
3. $k1 \in \mathrm{GF}(p)$, $k1 = (k \bmod p)1$, $k^m 1 = (k1)^m$ and if $k \bmod p \not\equiv 0$, then $(k1)^{-1} = (k \bmod p)^{-1} 1$.

In the remainder we identify $k1$ with $k \bmod p$ or in other terms we make explicit the identification of $\mathrm{GF}(p)$ and $\mathbb{Z}_p$.

Lemma 1. *Let G be a finite Abelian group. For $\chi \in \widehat{G}$,*

$$\sum_{x \in G} \chi(x) = \begin{cases} 0 & \text{if } \chi \neq 1 \ , \\ (|G| \operatorname{mod} p) & \text{if } \chi = 1 \ . \end{cases} \tag{23}$$

Proof. If $\chi = 1$, then $\sum_{x \in G} 1 = (|G| \operatorname{mod} p)$ since the characteristic of $\mathrm{GF}(q)$ is equal to p. Let us suppose that $\chi \neq 1$. Let $x_0 \in G$ such that $\chi(x_0) \neq 1$. Then we have

$$\chi(x_0) \sum_{x \in G} \chi(x) = \sum_{x \in G} \chi(x_0 + x) = \sum_{y \in G} \chi(y), \tag{24}$$

so that $(\chi(x_0) - 1) \sum_{x \in G} \chi(x) = 0$ and thus $\sum_{x \in G} \chi(x) = 0$ (because $\chi(x_0) \neq 1$).

Definition 4. *Let G be a finite Abelian group. Let $f, g \in \mathrm{GF}(q)^G$. We define the "inner product" of f and g by*

$$\langle f, g \rangle = \sum_{x \in G} f(x) \overline{g(x)} \in \mathrm{GF}(q). \tag{25}$$

The above definition does not ensure that $\langle f, f \rangle = 0$ implies that $f \equiv 0$ as it holds for a true inner product. Indeed, take $q = 2^{2n}$, and let $f : \mathbb{Z}_2 \to \mathrm{GF}(2^{2n})$ be the constant map with value 1. Then, $\langle f, f \rangle = 0$. Thus, contrary to a usual Hermitian dot product, an orthogonal family (with respect to $\langle \cdot, \cdot \rangle$) of $\mathrm{GF}(q)^G$ is not necessarily $\mathrm{GF}(q)$-linearly independent. Let G be a finite Abelian group. For all $(\chi_1, \chi_2) \in \widehat{G}^2$ then the orthogonality relations holds

$$\langle \chi_1, \chi_2 \rangle = \begin{cases} 0 & \text{if } \chi_1 \neq \chi_2, \\ |G| \operatorname{mod} p & \text{if } \chi_1 = \chi_2. \end{cases} \tag{26}$$

Proof. Let us denote $\chi = \chi_1 \chi_2^{-1} = \chi_1 \overline{\chi_2}$. We have:

$$\langle \chi_1, \chi_2 \rangle = \sum_{x \in G} \chi(x). \tag{27}$$

If $\chi_1 = \chi_2$, then $\chi = 1$ and if $\chi_1 \neq \chi_2$, then $\chi \neq 1$. The proof is obtained by using the previous lemma 1.

Remark 1. The term *orthogonality* would be abusive if $|G| \operatorname{mod} p = 0$, because then $\sum_{x \in G} \chi(x) = 0$ for all $\chi \in \widehat{G}$. Nevertheless from the assumption 3 all the d_i's divide $\sqrt{q} + 1 = p^n + 1$. In particular, $d_i = 1 \operatorname{mod} p$ and therefore $|G| = \prod_i d_i^{m_i}$ is co-prime to p, and the above situation cannot occur, so $|G|$ is invertible modulo p.

6. FOURIER TRANSFORM OVER A FINITE FIELD

There is already a Fourier transform with values in some finite field called *Mattson-Solomon transform* (Blahut, 1983) but it is not useful in our setting. Let u be a generator of $\mathcal{S}(\mathrm{GF}(q))$. Let G be a finite Abelian group and $f : G \to \mathrm{GF}(q)$. We define the following function.

$$\begin{array}{cccl} \hat{f}: & \widehat{G} & \to & \mathrm{GF}(q) \\ & \chi & \mapsto & \sum_{x \in G} f(x) \chi(x) \ . \end{array} \tag{28}$$

Since $G = \prod_{i=1}^{N} \mathbb{Z}_{d_i}^{m_i}$, we define, by the isomorphism between G and its dual,

$$\begin{array}{rcl} \hat{f}\colon \quad G & \rightarrow & \mathrm{GF}(q) \\ \alpha & \mapsto & \sum_{x\in G} f(x)\chi_\alpha(x) = \sum_{x\in G} f(x) \prod_{i=1}^{N} u^{\frac{(\sqrt{q}+1)\alpha_i\cdot x_i}{d_i}} \end{array} \qquad (29)$$

Let us compute $\hat{\hat{f}}$. Let $\alpha \in G$. We have

$$\begin{array}{rcl} \hat{\hat{f}}(\alpha) & = & \sum_{x\in G} \hat{f}(x)\chi_\alpha(x) \\ & = & \sum_{x\in G} \sum_{y\in G} f(y)\chi_x(y)\chi_\alpha(x) \\ & = & \sum_{x\in G} \sum_{y\in G} f(y)\chi_y(x)\chi_\alpha(x) \\ & = & \sum_{y\in G} f(y) \sum_{x\in G} \chi_{\alpha+y}(x) \\ & = & (|G| \mathrm{mod}\, p) f(-\alpha) \end{array} \qquad (30)$$

The last equality holds since

$$\sum_{x\in G} \chi_{\alpha+y}(x) = \begin{cases} 0 & \text{if } \ y \neq -\alpha \ , \\ (|G| \mathrm{mod}\, p) & \text{if } \ y = -\alpha \ . \end{cases}$$

Now if we assume that $(|G| \mathrm{mod}\, p) = 0$, then it follows that the function $f \mapsto \hat{f}$ is non invertible but this situation cannot occur since from the assumption 3, $|G|$ is invertible modulo p. Therefore we can claim that the function $\widehat{(\cdot)}$ that maps $f \in \mathrm{GF}(q)^G$ to $\hat{f} \in \mathrm{GF}(q)^G$ is invertible. It is referred to as the *Fourier transform* of f (with respect to $\mathrm{GF}(q)$) and it admits an *inversion formula*: for $f \in \mathrm{GF}(q)^G$ and for each $x \in G$,

$$f(x) = (|G| \ \mathrm{mod}\, p)^{-1} \sum_{\alpha\in G} \hat{f}(\alpha)\overline{\chi_\alpha(x)} \qquad (31)$$

where $(|G| \mathrm{mod}\, p)^{-1}$ is the multiplicative inverse of $(|G| \mathrm{mod}\, p)$ in $\mathbb{Z}_p$ (this inverse exists according to the choice of G). This Fourier transform shares many properties with the classical discrete Fourier transform.

Definition 5. Let G be a finite Abelian group. Let $f, g \in \mathrm{GF}(q)^G$. For each $\alpha \in G$, we define the *convolution product* of f and g at α by

$$(f * g)(\alpha) = \sum_{x\in G} f(x)g(-x+\alpha). \qquad (32)$$

Proposition 5 (Trivialization of the convolution product). *Let $f, g \in \mathrm{GF}(q)^G$. For each $\alpha \in G$,*

$$(\widehat{f * g})(\alpha) = \hat{f}(\alpha)\hat{g}(\alpha) \ . \qquad (33)$$

Proof. Let $\alpha \in G$. We have

$$\begin{array}{rcl} (\widehat{f * g})(\alpha) & = & \sum_{x\in G} (f * g)(x)\chi_\alpha(x) \\ & = & \sum_{x\in G} \sum_{y\in G} f(y)g(-y+x)\chi_\alpha(x) \end{array}$$

$$\begin{array}{rl} = & \sum_{x\in G}\sum_{y\in G} f(y)g(-y+x)\chi_\alpha(y-y+x) \\ = & \sum_{x\in G}\sum_{y\in G} f(y)g(-y+x)\chi_\alpha(y)\chi_\alpha(-y+x) \\ = & \hat{f}(\alpha)\hat{g}(\alpha). \end{array} \tag{34}$$

The group-algebra $\mathrm{GF}(q)[G]$ of G over $\mathrm{GF}(q)$ is the $\mathrm{GF}(q)$-vector space $\mathrm{GF}(q)^G$ equipped with convolution. The Fourier transform $\widehat{(\cdot)}$ is an algebra isomorphism from the group-algebra $\mathrm{GF}(q)[G]$ to $\mathrm{GF}(q)[G]$ with the point-wise product. Moreover, let $(\delta_x)_{x\in G}$ be the canonical basis of $\mathrm{GF}(q)^G$ (as a $\mathrm{GF}(q)$-vector space). It is easy to see that $\hat{\delta}_x = \chi_x$. Because $\widehat{(\cdot)}$ is an isomorphism, this means that $(\chi_x)_{x\in G}$ is a basis of $\mathrm{GF}(q)^G$ over $\mathrm{GF}(q)$, and it turns that the Fourier transform $\hat{f}$ of $f \in \mathrm{GF}(q)^G$ is the decomposition of f into the basis of characters (even if a family of elements of $\mathrm{GF}(q)^G$ is orthogonal with respect to the inner-product $\langle\cdot,\cdot\rangle$ of $\mathrm{GF}(q)^G$ this does not ensure linear independence because $\langle\cdot,\cdot\rangle$ is not positive-definite).

Proposition 6 (Plancherel formula). *Let $f, g \in \mathrm{GF}(q)^G$. Then,*

$$\sum_{x\in G} f(x)\overline{g(x)} = (|G| \bmod p)^{-1}\sum_{\alpha\in G}\hat{f}(\alpha)\overline{\hat{g}(\alpha)}. \tag{35}$$

Proof. Let us define the following functions with $h\colon G \to \mathrm{GF}(q)$,

$$\begin{array}{rlll} I_G\colon & G & \to & G \\ & x & \mapsto & -x \\ & \text{and} & & \\ \bar{h}\colon & G & \to & \mathrm{GF}(q) \\ & x & \mapsto & \overline{h(x)}\ . \end{array} \tag{36}$$

Then $(f * \bar{g}\circ I_G)(0_G) = \sum_{x\in G} f(x)\overline{g(x)}$. By the inversion formula:

$$\begin{array}{rcl} (f * \bar{g}\circ I_G)(0_G) & = & (|G| \bmod p)^{-1}\sum_{\alpha\in G}(\widehat{f * \bar{g}\circ I_G})(\alpha) \\ & = & (|G| \bmod p)^{-1}\sum_{\alpha\in G}\hat{f}(\alpha)(\widehat{\bar{g}\circ I_G})(\alpha). \end{array} \tag{37}$$

Let us compute $(\widehat{\bar{g}\circ I_G})(\alpha)$ for $\alpha \in G$.

$$\begin{array}{rcl} (\widehat{\bar{g}\circ I_G})(\alpha) & = & \sum_{x\in G}(\bar{g}\circ I_G)(x)\chi_\alpha(x) \\ & = & \sum_{x\in G}\overline{g(-x)}\chi_\alpha(x) \\ & = & \sum_{x\in G}\overline{g(x)}\chi_\alpha(-x) \\ & = & \sum_{x\in G}\overline{g(x)}(\chi_\alpha(x))^{-1} \\ & = & \sum_{x\in G}\overline{g(x)\chi_\alpha(x)} \\ & = & \overline{\sum_{x\in G} g(x)\chi_\alpha(x)} \\ & = & \hat{g}(\alpha) \end{array} \tag{38}$$

Then we obtain the equality that ensures the correct result

$$(f * \overline{g} \circ I_G) = (|G| \bmod p)^{-1} \textstyle\sum_{\alpha\in G} \hat{f}(\alpha)\overline{\hat{g}(\alpha)} \tag{39}$$

Corollary 2 (Parseval equation). *Let $f, g \in \mathrm{GF}(q)^G$. Then*

$$\sum_{x\in G} \mathrm{norm}(f(x)) = (|G| \bmod p)^{-1} \sum_{\alpha\in G} \mathrm{norm}(\hat{f}(\alpha)) \ . \tag{40}$$

In particular, if f is $\mathcal{S}(\mathrm{GF}(q))$-valued, then

$$\sum_{\alpha\in G} \mathrm{norm}(\hat{f}(\alpha)) = (|G| \bmod p)^2 \tag{41}$$

7. BENT FUNCTIONS OVER A FINITE FIELD

In the traditional setting, i.e., for complex-valued functions defined on any finite Abelian group G, bent functions ((Carlet and Ding, 2004), (Dillon, 1974), (Logachev, Salnikov, and Yashchenko, 1997), (Nyberg, 1990) (Rothaus, 1976)) are those maps $f: G \to \mathcal{S}(\mathbb{C})$ such that for each $\alpha \in G$,

$$|\hat{f}(\alpha)|^2 = |G| \ . \tag{42}$$

This notion is closely related to some famous cryptanalysis namely the differential (Biham and Shamir, 1991) and linear (Matsui, 1994) attacks on secret-key cryptosystems. We translate this concept in the current finite-field setting as follows.

Definition 6. *The map $f: G \to \mathcal{S}(\mathrm{GF}(q))$ is called bent if for all $\alpha \in G$,*

$$\mathrm{norm}(\hat{f}(\alpha)) = (|G| \bmod p). \tag{43}$$

7.1 Derivative and bentness

Propositon 7. (Logachev, Salnikov, and Yashchenko, 1997) *Let $f: G \to \mathcal{S}(\mathbb{C})$. The function f is bent if, and only if, for all $\alpha \in G^*$,*

$$\sum_{x\in G} f(\alpha + x)\overline{f(x)} = 0. \tag{44}$$

Now let $f: G \to \mathrm{GF}(q)$. For each $\alpha \in G$, we define the derivative of f in direction α as

$$\begin{array}{rccc} d_\alpha f: & G & \to & \mathrm{GF}(q) \\ & x & \mapsto & f(\alpha + x)\overline{f(x)} \ . \end{array} \tag{45}$$

Lemma 2. *Let $f: G \to \mathrm{GF}(q)$. We have*

1. $\forall x \in G^*, f(x) = 0 \Leftrightarrow \forall \alpha \in G, \hat{f}(\alpha) = f(0_G)$.
2. $\forall \alpha \in G^*, \hat{f}(\alpha) = 0 \Leftrightarrow f$ is constant.

Proof.

1. $\Rightarrow \hat{f}(\alpha) = \sum_{x\in G} f(x)\chi_\alpha(x) = f(0_G)\chi_\alpha(0_G) = f(0_G)$,
   $\Leftarrow$ According to the inversion formula,

$$\begin{aligned} f(x) &= (|G| \bmod p)^{-1} \sum_{\alpha \in G} \hat{f}(\alpha)\overline{\chi_{\alpha}(x)} \\ &= f(0_G)(|G| \bmod p)^{-1} \sum_{\alpha \in G} \chi_{-x}(\alpha) \\ &= 0 \quad \text{for all} \quad x \neq 0_G \ . \end{aligned} \tag{46}$$

2. $\Rightarrow f(x) = (|G| \bmod p)^{-1} \sum_{\alpha \in G} \hat{f}(\alpha)\overline{\chi_{\alpha}(x)} = \hat{f}(0_G)(|G| \bmod p)^{-1}$, $\Leftarrow \hat{f}(\alpha) = \sum_{x \in G} f(x)\chi_{\alpha}(x) = \text{constant} \sum_{x \in G} \chi_{\alpha}(x) = 0$ for all $\neq 0_G$.

Lemma 3. Let $f: G \to GF(q)$. We define the autocorrelation function of f as

$$\begin{aligned} AC_f: \quad G &\to \mathrm{GF}(q) \\ \alpha &\mapsto \sum_{x \in G} d_{\alpha} f(x) \ . \end{aligned} \tag{47}$$

Then, for all $\alpha \in G$, $\widehat{AC_f}(\alpha) = \mathrm{norm}(\hat{f}(\alpha))$.
Proof. Let $\alpha \in G$.

$$\begin{aligned} \widehat{AC_f}(\alpha) &= \sum_{x \in G} AC_f(x)\chi_{\alpha}(x) \\ &= \sum_{x \in G} \sum_{y \in G} d_x f(y)\chi_{\alpha}(x) \\ &= \sum_{x \in G} \sum_{y \in G} f(xy)\overline{f(y)}\chi_{\alpha}(xy)\overline{\chi_{\alpha}(y)} \\ &= \hat{f}(\alpha)\overline{\hat{f}(\alpha)} \\ &= \mathrm{norm}(\hat{f}(\alpha)) \ . \end{aligned} \tag{48}$$

Theorem 3. *The function $f: G \to \mathcal{S}(\mathrm{GF}(q))$ is bent if, and only if, for all $\alpha \in G^*$, $\sum_{x \in G} d_{\alpha} f(x) = 0$.*
Proof. $\forall \alpha \in G^*,\ \sum_{x \in G} d_{\alpha} f(x) = 0$
$\Leftrightarrow \forall \alpha \in G^*,\ AC_f(\alpha) = 0$
$\Leftrightarrow \forall \alpha \in G,\ \widehat{AC_f}(\alpha) = AC_f(0_G)$
(according to lemma 2)
$\Leftrightarrow \forall \alpha \in G,\ \mathrm{norm}(\hat{f}(\alpha)) = \sum_{x \in G} f(x)\overline{f(x)}$
(according to lemma 3)
$\Leftrightarrow \forall \alpha \in G,\ \mathrm{norm}(\hat{f}(\alpha)) = \sum_{x \in G} \mathrm{norm}(f(x))$
$\Leftrightarrow \forall \alpha \in G,\ \mathrm{norm}(\hat{f}(\alpha)) = (|G| \bmod p)$
(because f is $\mathcal{S}(\mathrm{GF}(q))$-valued.)

7.2 Dual bent function

Again by analogy to the traditional notion (Carlet and Dubuc, 2001; Kumar, Scholtz, and Welch, 1985), it is also possible to define a dual bent function from a given bent function. Actually, as we see it below, $|G|$ must be a square in $\mathrm{GF}(p)$ to ensure the well-definition of a dual bent. So by using the *law of quadratic reciprocity*, we can add the following requirement (only needed for proposition 8).

Assumption 4. *If the prime number p is ≥ 3, then $|G|$ must also satisfy $|G|^{\frac{p-1}{2}} \equiv 1 (\bmod p)$. If the prime number $p = 2$, then there is no other assumptions on $|G|$ (than those already made).*

According to assumption 4, $|G|\text{mod}\, p$ is a square in $\text{GF}(p)$, then there is at least one $x \in \text{GF}(p)$ with $x^2 = |G|\text{mod}\, p$. If $p = 2$, then $x = 1$. If $p \geq 3$, then we choose for x the element $(|G|\text{mod}\, p)^{\frac{p+1}{4}}$. Indeed it is a square root of $|G|$mod p since

$$\begin{aligned}((|G|\text{mod p})^{\frac{p+1}{4}})^2 &= (|G|\text{mod p})^{\frac{p+1}{2}}\\ &= (|G|(\text{mod p}))(|G|(\text{mod p}))^{\frac{p-1}{2}}\\ &= |G|(\text{mod}\, p).\end{aligned}$$

In all cases we denote by $(|G|\text{mod}\, p)^{\frac{1}{2}}$ the chosen square root of $|G|\text{mod}\, p$. Since $|G|\text{mod}\, p \not\equiv 0$, then it is clear that this square root also is non-zero. Its inverse is denoted by $(|G|\text{mod}\, p)^{-\frac{1}{2}}$. Finally it is clear that $((|G|\text{mod}\, p)^{-\frac{1}{2}})^2 = (|G|\text{mod}\, p)^{-1}$.

Proposition 8. *Let $f\colon G \to \mathcal{S}(\text{GF}(q))$ be a bent function, then the following function $\tilde{f}$, called dual of f, is bent.*

$$\begin{array}{cccc}\tilde{f}\colon & G & \to & \mathcal{S}(\text{GF}(q))\\ & \alpha & \mapsto & (|G|\text{mod}\, p)^{-\frac{1}{2}}\hat{f}(\alpha)\ .\end{array} \tag{49}$$

Proof. Let us first check that $\tilde{\text{f}}$ is $\mathcal{S}$(GF(q))-valued. Let $\alpha \in \text{G}$. We have

$$\begin{aligned}\tilde{f}(\alpha)\overline{\tilde{f}(\alpha)} &= (|G|\text{mod}\, p)^{-\frac{1}{2}}\hat{f}(\alpha)\overline{(|G|\text{mod}\, p)^{-\frac{1}{2}}\hat{f}(\alpha)}\\ &= (|G|\text{mod}\, p)^{-1}\text{norm}(\hat{f}(\alpha))\\ &= 1\ \ (\text{since}\ \ f\ \ \text{is}\ \ \text{bent.})\end{aligned} \tag{50}$$

Let us check that the bentness property holds for $\tilde{f}$. Let $\alpha \in G$. We have $\hat{\tilde{f}}(\alpha) = (|G|\text{mod}\, p)^{-\frac{1}{2}}(|G|\text{mod}\, p)f(-\alpha)$ (by (30)). Then

$$\begin{aligned}\hat{\tilde{f}}(\alpha)\overline{\hat{\tilde{f}}(\alpha)} &= (|G|\text{mod}\, p)f(-\alpha)\overline{f(-\alpha)}\\ &= (|G|\text{mod}\, p)\text{norm}\big(f(-\alpha)\big)\\ &= (|G|\text{mod}\, p)\big(\text{since}\ \ f\ \ is\ \ \mathcal{S}\big(\text{GF}(q)\big) - \text{valued.}\big)\end{aligned} \tag{51}$$

7.3 Construction of bent functions

We present a simple version of the well-known Maiorana-McFarland construction ((Dillon, 1974), (McFarland, 1973)) for our bent functions.
Let $g\colon G \to \mathcal{S}(\text{GF}(q))$ be any function. Let f be the following function.

$$\begin{array}{cccc}f\colon & G^2 & \to & \mathcal{S}(\text{GF}(q))\\ & (x,y) & \mapsto & \chi_x(y)g(y)\ .\end{array} \tag{52}$$

Then f is bent. We observe that the fact that f is $\mathcal{S}(\text{GF}(q))$-valued is obvious by construction. So let us prove that f is indeed bent. We use the combinatorial characterization obtained in theorem 3. Let $\alpha, \beta, x, y \in G$.

$$\begin{aligned}
d_{(\alpha,\beta)}f(x,y) &= f(\alpha+x,\beta+y)\overline{f(x,y)} \\
&= \chi_{\alpha+x}(\beta+y)g(\beta+y)\overline{\chi_x(y)}\ \overline{g(y)} \\
&= \chi_\alpha(\beta+y)\chi_x(\beta+y)g(\beta+y)\overline{\chi_x(y)}\ \overline{g(y)} \\
&= \chi_\alpha(\beta)\chi_\alpha(y)\chi_x(\beta)\chi_x(y)g(\beta+y)\overline{\chi_x(y)}\ \overline{g(y)} \\
&= \chi_\alpha(\beta)\chi_\alpha(y)g(\beta+y)\overline{g(y)}\chi_x(\beta) \\
&= \chi_\alpha(\beta)\chi_\alpha(y)g(\beta+y)\overline{g(y)}\chi_\beta(x)
\end{aligned} \tag{53}$$

because $\chi_x(\beta) = \chi_\beta(x)$.
So for $(\alpha,\beta) \in (G^2)^* = G^2\backslash\{(0_G,0_G)\}$, we obtain

$$\begin{aligned}
\sum_{(x,y)\in G^2} d_{(\alpha,\beta)}f(x,y) &= \sum_{(x,y)\in G^2} \chi_\alpha(\beta)\chi_\alpha(y)g(\beta+y)\overline{g(y)}\chi_\beta(x) \\
&= \chi_\alpha(\beta)\sum_{y\in G}\chi_\alpha(y)g(\beta+y)\overline{g(y)}\sum_{x\in G}\chi_\beta(x)
\end{aligned} \tag{54}$$

The sum $\sum_{x\in G}\chi_\beta(x)$ is equal to 0 if $\beta \not\asymp 0_G$ and $|G|$mod p if $\beta = 0_G$ (according to lemma 1). Then the right member of the equality (54) is equal to 0 if $\beta \not\asymp 0_G$ and $(|G|\text{mod p})\chi_\alpha(\beta)\sum_{y\in G}\chi_\alpha(y)g(\beta+y)\overline{g(y)}$ if $\beta = 0_G$. So when $\beta \not\asymp 0_G$, $\sum_{(x,y)\in G^2} d_{(\alpha,\beta)}f(x,y) = 0$. Now let us assume that $\beta = 0_G$, then because $(\alpha,\beta) \in G^2\backslash\{(0_G,0_G)\}$, $\alpha \not\asymp 0_G$, we have

$$\begin{aligned}
\sum_{(x,y)\in G^2} d_{(\alpha,0_G)}f(x,y) &= (|G|\text{mod p})\chi_\alpha(0_G)\sum_{y\in G}\chi_\alpha(y)g(0_G+y)\overline{g(y)} \\
&= (|G|\text{mod p})\sum_{y\in G}\chi_\alpha(y) \\
&\quad \text{(because } g \text{ is } \mathcal{S}(\mathrm{GF}(q)) - \text{valued)} \\
&= 0 \quad \text{(because } \alpha = 0_G.\text{)}
\end{aligned} \tag{55}$$

So we have checked that for all $(\alpha,\beta) \in G^2\backslash\{(0_G,0_G)\}\sum_{(x,y)\in G^2} d_{(\alpha,\beta)}f(x,y) = 0$ and then according to theorem 3 this implies that f is bent.

8. CONCLUSION AND PERSPECTIVES

There is a close analogy between any quadratic extension of a finite field and the extension of the complex numbers over the field of real numbers. Indeed in both cases it is possible to define an Hermitian structure on the field extension based on a conjugation operation. As in the usual case this structure makes it possible to an introduce a notion of finite field valued characters of (some) finite Abelian groups. These characters form a basis (orthogonal in a certain sense) of the algebra of the group over the base finite field. With this characters in hand it is then possible to introduce a Fourier transform that shares the same properties as the usual one. The study of the Hermitian structure on a quadratic extension and of its consequences was the main objective of this contribution. Because the cryptographic notion of bent functions (particular highly non linear functions) is directly based on the Fourier transform it makes sense also to study this kind of functions in this new setting. This was the second objective of this contribution, achieved by providing two constructions of (finite field valued) bent functions. As an immediate perspective of our work is the analysis of the connections between the usual notion of bent functions and that introduced in the contribution. The relations between the two kinds of bent functions, if any, were outside the scope of this paper but should be the main goal of our future researches on this subject.

9. ACKNOWLEDGMENT

One the authors, Nadia El Mrabet, wishes to acknowledge support from Agence Nationale de la Recherche (France) under project ANR INS 2012 SYMPATIC.

REFERENCES

Ambrosimov, A. S. 1994. Properties of bent functions of q-valued logic over. *Discrete Mathematics and, 4*(4), 341-350.

Biham, E., and Shamir, A. 1991. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology, 4*(1), 3-72.

Blahut, R. E. 1983. *Theory and practice of error control codes.* Addison-Wesley.

Boneh, D., and Franklin, M. 2003. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing, 32*(3), 586-615.

Carlet, C. 2010. Boolean Functions for Cryptography and Error Correcting Codes. In Y. Crama, and P. L. Hammer (Eds.), *Boolean Models and Methods in Mathematics, Computer Science, and Engineering* (pp. 398-469). New York: Cambridge University Press.

Carlet, C., and Ding, C. 2004. Highly nonlinear mappings. *Journal of Complexity, 20*(2-3), 205-244.

Carlet, C., and Dubuc, S. 2001. On generalized bent and q-ary perfect nonlinear functions. In D. Jugnickel, and H. Niederreiter (Ed.), *Fifth International Conference on Finite Fields and Applications Fq5,* (pp. 81-94).

Dillon, J. F. 1974. *Elementary Hadamard difference sets (Ph.D Thesis).* University of Maryland.

Dobbertin, H., Leander, G., Canteaut, A., Carlet, C., Felke, P., and Gaborit, P. 2006. Construction of Bent Functions via Niho Power Functions. *Journal of Combinatorial Theory, Serie A, 113,* 779-798.

Hewitt, E., and Ross, K. A. 1994. *Abstract Harmonic Analysis* (2 ed., Vol. 1). Springer.

Kumar, P. V., Scholtz, R. A., and Welch, L. R. 1985. Generalized bent functions and their properties. *Journal of Combinatorial Theory A, 40,* 99-107.

Logachev, O. A., Salnikov, A. A., and Yashchenko, V. V. 1997. Bent functions on a finite Abelian group. *Discrete Math. Appl., 7*(6), 547-564.

Matsui, M. 1994. Linear cryptanalysis for DES cipher. In T. Hellesth (Ed.), *Advances in cryptology - Eurocrypt'93* (pp. 386-397). Springer.

McFarland, R. L. 1973. A family of difference sets in non-cyclic groups. *Journal of Combinatorial Theory, 15*, 1-10.

McLane, S. 1998. *Categories for the working mathematician* (2nd ed., Vol. 5). Springer.

Nyberg, K. 1990. Constructions of bent functions and difference sets. In I. Damgard (Ed.), *Advances in cryptology - Eurocrypt'90,* (pp. 151-160).

Poinsot, L. 2005. Multidimensional bent functions. *GESTS International Transactions on Computer Science and Engineering, 18*(1), 185-195.

Rothaus, O. S. 1976. On bent functions. *Journal of Combinatorial Theory A*, *20*, 300-365.

Survey and New Idea for Attribute-Based Identification Scheme Secure against Reset Attacks

[1]Ji-Jian Chin, [2]Hiroaki Anada, [3]Seiko Arita, [2,4]Kouichi Sakurai, [5]Swee-Huay Heng and [1]Raphael Phan

[1]Faculty of Engineering, Multimedia University
[2]Institute of Systems, Information Technologies and Nanotechnologies, Japan
[3]Institute of Information Security, Japan.
[4]Faculty of Information Science and Electrical Engineering, Kyushu University
[5]Faculty of Information Science and Technology, Multimedia University
Email: [1]jjchin@mmu.edu.my, [2]anada@isit.or.jp

ABSTRACT

Identification schemes are a common one-way authentication technique for a user to prove himself securely to a verifier. However, it is known that identification schemes based on the sigma-protocol are basically insecure against reset attacks. On the other-hand, attribute-based cryptography is a technique which allows for the secure implementation of access policies within a cryptosystem. In this paper, we report on the developments in the area of reset attacks for identification schemes as well as for attribute-based identification schemes. Then we put together a new idea to construct attribute-based identification schemes secure against reset attacks.

Keywords: reset-attacks, attribute-based, identification

1. INTRODUCTION

An identification scheme is a cryptographic primitive that allows one party, the prover, to prove himself convincingly to another party, the verifier, without revealing any knowledge about his private key. First proposed by Fiat and Shamir (1983), this primitive is usually used to facilitate access control to allow legitimate users to access resources upon being able to prove themselves securely to a verifying mechanism.

Identification schemes are generally categorized into two-move challenge-response and three-move sigma protocols. Two-move challenge-response protocols basically revolve around the capability of the prover to decrypt a challenge ciphertext or sign a verifiable message, given that he has a valid private key. However, in general, two-move protocols are more expensive operationally.

For three-move sigma protocols the prover and verifier engage in a three-step canonical interaction every time a prover wishes to prove itself. The prover begins by sending a commitment. The verifier follows by selecting a random challenge from a predefined challenge set. Then the prover provides a response using a combination of his private key, commitment as well as the challenge. The verifier will then decide to accept or reject a prover’s session based on the response.

Sigma-protocols have the following properties:

i) Completeness – provers with valid private keys should be given an “accept” except with negligible probability.

ii) Soundness – provers with invalid private keys should be given a reject decision except with negligible probability.

iii) Zero-knowledge – certain sigma protocols have a zero-knowledge property, where the verifier upon completing the interaction with the prover learns nothing about the user’s private key. This is proven by a simulator that is able to produce a valid interaction transcript with or without a prover’s participation. However, since it is hard to prove security against concurrent-active attacks for protocols with zero-knowledge properties, sometimes the requirement is relaxed to just satisfying a witness indistinguishability requirement (Fiege and Shamir, 1990), where a verifier cannot distinguish between the two witnesses used in the protocol.

1.1 Reset Attacks

While generally two-move challenge-response protocols are secure against reset attacks, unfortunately sigma protocols have an inherent weakness against reset attacks, where an adversary is allowed to reset the prover to where he first sent the commitment. Then due to the soundness property, with two different challenges, the adversary is able to extract a user's private key from the different responses and challenges but using the same commitment.

Reset attacks can be performed if an adversary has access to the verifying machine, for example a smart card reader that is able to tamper with the internal state of the smart card. Thus the adversary with access to this smart card reader will be able to extract an honest user's private key if the user interacts with it.

The reset attack was first addressed for identification schemes by Bellare *et al.* (2001). In their seminal paper, they tackled the problem of adversaries with the resetting capability and proposed several methods of overcoming this problem. We provide a more comprehensive review of these methods in a later section of this paper.

The power of reset attacks can be seen by the following scenarios given by Bellare *et al.* (2001), describing how a reset-attack can be mounted practically. Firstly, if an adversary captures a prover device such as a smart card, the adversary can disconnect and reinsert the battery to reset the card's secret internal state to its initial state. This can be done multiple times.

Secondly if an adversary is able to crash the prover device, such as by causing a stack/heap overflow, upon reinitializing the device will resume computation after the crash, forcing the device to reset itself.

Thus, reset-secure identification schemes are desirable due to the existence of these threats.

1.2 Identification Schemes without Certificates

In traditional public key cryptography, certificates are required to bind a user to his public key, which could otherwise be replaced by a malicious party. These certificates are issued by certificate authorities, and include a wide-array of information ranging from the public key to validity period. Any doubtful parties can verify that a user's public key actually belongs to a particular user by checking the Certificate Authority's digital signature on the certificate.

The certificate management issue occurs when the users of the cryptosystem grow large and a large overhead is required to issue, validate, manage and revoke these certificates. To circumvent this issue, Adi Shamir first proposed identity-based cryptography (Shamir, 1984), where users can implicitly certify themselves using a publicly known identity-string. Identity-based cryptography only kicked off in 2001 when Boneh and Franklin (2001) proposed the first identity-based encryption scheme. In 2004, the first identity-based identification schemes were proposed by Bellare *et al.* (2004) and Kurosawa and Heng (2004) independently.

Since then, many identity-based identification schemes have been proposed, but none of them are secure against reset attacks. The first identity-based identification scheme that is secure against reset attacks was first proposed by Thorncharoensri *et al.* (2009).

In addition to identity-based cryptography, other extensions for identification schemes that operate without the requirement of certificates have surfaced in the recent decade. Certificateless cryptography was proposed by Alriyami and Paterson (2001) to resolve the key escrow issue, where the central key generation center has access to every user's private key. In certificateless cryptography, the key generation center creates a partial private key, which the user combines with his component of the private key to create the full private key. Thus without the user's component the key generation center does not have complete access to the full private key. For the identification primitive, certificateless identification was first defined and proposed by Dehkordi and Alimoradi (2013) and Chin *et al.* (2013) independently. However, subsequently Chin *et al.* (2014) pointed out flaws in Dehkordi and Alimoradi (2013)'s design, therefore it is insecure against impersonation attacks.

Another new area of identification schemes without certificates is the attribute-based identification (ABID) scheme. Attribute-based identification was introduced by Anada *et al.* (2013). In an ABID scheme, each entity has credentials called attributes. An access policy is written as a boolean formula over these attributes. Thus, a verifier can identify that a prover possesses a certain set of attributes that satisfies the verifier's access policy. Hence, ABID schemes can be considered as an expansion of the usual ID schemes. In Anada *et al.* (2013)'s seminal paper, a two-move generic (and concrete) construction was presented. That is, by employing an attribute-based key encapsulation mechanism (Sahai and Waters 2005, Waters 2011), a challenge-and-response protocol was proposed. Their scheme was claimed to be secure against reset attacks, but only a brief sketch of security proof was provided. After their two-move construction, a three-move construction was presented by Anada *et al.* (2014a). This three-move construction was further extended to be a basic building block for attribute-based signature schemes using the Fiat-Shamir transform. (Anada *et al.* 2014b)

In contrast to the earlier construction by Anada *et al.* (2013), the three-move construction was based on the (traditional) sigma protocol (Cramer *et al.*, 2001). Enhancing the technique of OR-proof (Damgard, 2004), they succeeded to provide a three-move generic ABID scheme that can be concretely realized without pairings. Hence Anada *et al.* (2014a)'s three-move protocol can be said to be more efficient than the two-move protocol (Anada *et al.*, 2013). But their three-move protocol is not secure against reset attacks because its security is based on the Reset-Lemma (Bellare and Palacio, 2002). That is, under the condition that an adversary is allowed to reset an honest prover, the adversary can extract the prover's witness in polynomial-time.

1.3 Motivations and Contributions

Since its conception in 2004, identification schemes without certificates have received much attention, particularly attribute-based identification schemes. Secondly, the notion of reset attacks has not yet been examined in depth, particularly with regards for identification schemes without certificates.

In this paper, we introduce the reader to the security notions of reset-secure identification as well as attribute-based identification (ABID) schemes. After that, we provide the first generic construction to modify a three-move attribute-based identification scheme to be secure against reset attacks.

It is worthwhile to note that the security against reset attacks discussed in this paper is Concurrent-Reset-1 (CR1) security defined by Bellare *et al.* (2001). CR1 security is, even if we ignore concurrency, different from the security of resettable zero-knowledge and the security of resettable soundness (see Arita 2012 for definitions).

The rest of the paper is organized as follows: In Section 2 we begin by reviewing the definitions and security model of reset-secure identification schemes and ABID schemes. In Section 3, we introduce the first generic construction to modify three-move ABID schemes to be reset-secure. We conclude in Section 4 with some closing remarks.

2. PRELIMINARIES AND DEFINITIONS

In this section, we review the formal definitions and security notions for reset-secure identification schemes as well as ABID schemes.

2.1 Reset-Secure Identification Schemes

An identification scheme consists of three probabilistic polynomial-time algorithms: Keygen, Prover and Verifier.

Keygen takes in the security parameter 1^k and generates a public/private key pair for the user $\langle pk, sk\rangle$.

Prover takes in the private key sk while Verifier takes in the public key pk. Together they run the sigma protocol as such:

1) Prover sends the commitment CMT.
2) Verifier selects and sends a random challenge CHA from a set of predefined challenges.
3) Prover calculates his response RSP based on the challenge and returns it to Verifier. Verifier will then choose to accept/reject based on the response given.

An adversary towards an identification scheme is an impersonator. For normal identification schemes an impersonator can be a passive one, where he only eavesdrops on conversations, or an active one where he can play a cheating verifier to learn information by interacting with honest users before attempting impersonation.

For reset-secure identification schemes, an additional concurrent reset-attacker is defined. This attacker is more powerful than the conventional passive/active attacker and is able to run several instances of the prover interactions concurrently, interleaving executions and performing reset actions on the prover states. Bellare *et al.* first formalized these two types of concurrent reset attackers as CR1 and CR2 respectively.

For the CR1 attacker, the adversary may interact with the honest user's Prover algorithm as a verifier and in addition to identification queries, be able to perform a reset action for the Prover algorithm to any state. Later the adversary performs the impersonation attempt.

For the CR2 attacker, the adversary may do all the actions described for the CR1 attacker, but may attempt impersonation whenever it wishes to. Therefore, the CR1 attacker is a special case of CR2 attacker.

We describe the security for the reset-secure identification scheme using the following game played between a challenger C and an impersonator I.

Keygen: C takes in the security parameter 1^k, generates $\langle pk, sk\rangle$ and passes pk to I.

Phase 1: I is able to make the following queries:

i) Identification queries: I interacts as a cheating verifier with a prover simulated by C to learn information.
ii) Reset queries: I resets the prover simulated by C to any state that it wishes within the three-step sigma protocol.

Phase 2: I changes mode into a cheating prover trying to convince C. For CR2 impersonators, I can still continue to make any of the queries from Phase 1. I wins if it manages to convince C to accept its interaction with non-negligible probability.

We say an identification scheme is $(t, q_I, q_r, \varepsilon)$-secure under concurrent reset attacks if any reset concurrent impersonator I that runs in time t, $Pr[I \text{ can impersonate}] < \varepsilon$ where I can make at most q_I identification queries and q_r reset queries.

Bellare *et al.* (2001) also proposed four techniques in order to secure identification schemes that are constructed using the sigma protocol against reset attackers, which are naturally insecure against reset attacks. We briefly describe the four techniques here:

1) Stateless digital signatures: a prover can authenticate himself to a verifier by showing the capability of signing random documents the verifier chooses. Here the message becomes the challenge while the signature is used as the response. Statelessness is required so that the reset attacker cannot reset the state of the signer. However, this is generally a two-move protocol.
2) Encryption schemes: a prover can authenticate himself to a verifier by showing the capability to decrypt random ciphertexts the verifier chooses. Here the ciphertext becomes the challenge while the message becomes the response. However, reset-security requires that an encryption scheme secure against chosen-ciphertext attacks be used.
3) Trapdoor commitments: this technique uses a trapdoor commitment scheme to 'commit' a verifier's challenge. This commitment is used as the generator for the prover's salt using a pseudorandom function. One can therefore verify that upon revealing the verifier's challenge, the salt can be regenerated in order to create the proper response for the verifier. If the prover was reset, the regeneration of the salt would yield a different (and invalid) response.
4) Zero-knowledge proof of membership: a prover proves membership in a hard language rather than proving that it has a witness for the language. This is done by using a resettable zero-knowledge proof of language membership, as defined by Canetti *et al.* (2000).

In this work, we utilize the third technique as a generic way to construct reset-secure ABID schemes.

2.2 ABID Schemes

Let $U = \{1, \ldots, u\}$ be an attribute Universe. An access structure A, which means an access policy, is defined as a subset of $2^U \backslash \phi$. We only treat monotone access structures.

An ABID scheme consists of four PPT algorithms: Setup, KeyGen, Prover and Verifier.

Setup(1^k,U) → (PK, MSK). Setup takes as input the security parameter λ and the attribute universe U. It outputs a public key PK and a master secret key MSK.

KeyGen(PK, MSK, S) →SK_S . A key-generation algorithm KeyGen takes as input the public key PK, the master secret key MSK and an attribute set S. It outputs a secret key SK_S corresponding to S.

Prover(PK, SK_S) and Verifier(PK, A). Prover and Verifier are interactive algorithms. Prover takes as input the public key PK and the secret key K_S . Here the secret key SK_S is given to Prover by an authority that runs KeyGen(PK, MSK, S). Verifier takes as input the public key PK and an attribute set S. Prover is provided Verifier's access structure A by the first round. Prover and Verifier interact with each other for some rounds. Then, Verifier finally returns its decision bit b. When $b = 1$ it means that Verifier $accepts$ Prover in the sense Prover has a secret key SK_S such that S satisfies A. When $b = 0$ it means that Verifier $rejects$ Prover.

We require correctness of an ABID scheme that for any 1^k and U, and if $S \in A$, then the probability of Verifier outputting an $accept$ will always be true, namely

$$\Pr[(PK, MSK) \leftarrow Setup(1^k, U);$$
$$SK_S \leftarrow KeyGen(PK, MSK, S\,);$$
$$b \leftarrow \langle P(PK, SK_S\,), V(PK, A)\rangle : b = 1] = 1.$$

3. GENERIC CONSTRUCTION OF 3-MOVE RESET-SECURE ABID SCHEME

In this section, we present a new and generic idea for modifying three-move ABID schemes to be secure against reset attacks. We utilize Bellare *et al.* (2001)'s third paradigm, which is to use a trapdoor commitment scheme, and embed this scheme within the three-move ABID scheme. The resulting scheme consists of four-moves.

The construction of the scheme is described in Table 1.

<table>
<tr><td colspan="3"><b>Setup($\mathbf{1}^{k}, \boldsymbol{U}$)→ $(\boldsymbol{PK} \coloneqq (\boldsymbol{PK}_{\boldsymbol{ABID}}, \boldsymbol{PK}_{\boldsymbol{TDC}}), \boldsymbol{MSK})$:</b>

Setup takes in the security parameter 1^{k} and the space of the attribute universe U and outputs the public key and master secret key $\langle PK = (PK_{ABID}, PK_{TDC}), MSK\rangle$. However, the public key consists of two components, one for the ABID scheme PK_{ABID} and the other for the trapdoor commitment scheme PK_{TDC}.</td></tr>
<tr><td colspan="3"><b>KG($\boldsymbol{PK}_{\boldsymbol{ABID}}, \boldsymbol{MSK}, \boldsymbol{S}$)→ $\boldsymbol{SK}_{\boldsymbol{S}}$:</b>

Keygen KG takes in the public key for the ABID scheme, PK_{ABID}, the master secret key MSK and the set of attributes S and outputs the secret key SK_{S} corresponding to S.</td></tr>
<tr><td><b>Prover($\boldsymbol{PK}_{\boldsymbol{ABID}}, \boldsymbol{PK}_{\boldsymbol{TDC}}, \boldsymbol{SK}_{\boldsymbol{S}}$):</b>

$R_{ABID} \leftarrow PRF(R_{P}, TDCMT)$
$CMT \leftarrow ABID_{CMT}(SK_{S}, R_{ABID}, A)$

IF
$TDC_{VF}(PK_{TDC}, TDCMT, CHA_{V}||R_{C})$
$= accept$
THEN
$RSP \leftarrow$
$ABID_{RSP}(SK_{S}, CMT, CHA_{V}; R_{ABID})$
ELSE $RSP = \perp$</td><td>

$\xleftarrow{TDCMT,A}$
$\xrightarrow{CMT}$
$\xleftarrow{CHA_{V}||R_{C}}$

$\xrightarrow{RSP}$</td><td><b>Verifier($\boldsymbol{PK}, \boldsymbol{A}$)</b>
$CHA_{V} \leftarrow ABID_{CHA}(1^{k})$
$TDCMT \leftarrow TDC_{CMT}(PK_{TDC}, CHA_{V}; R_{C})$

$dec \leftarrow ABID_{VF}(PK_{ABID}, A, CMT, CHA_{V}RSP)$</td></tr>
<tr><td colspan="3"><u>Prover and Verifier engage in the identification protocol as follows:</u>
1) Upon receiving an initialization message from Prover, Verifier first generates a commitment $TDCMT$ for his random challenge CH_{V} using the trapdoor commitment scheme's commit algorithm TDC_{CMT} and sends it to Prover along with the access policy A.
2) Prover evaluates $TDCMT$ and his own internal coins R_{P} with a pseudorandom function PRF and generates the salt R_{ABID}. This salt is used to generate his commitment CMT and is sent to Verifier.
3) Verifier then sends his random challenge CH_{V} and random coins R_{C} to Prover.
4) The Prover uses the trapdoor commitment scheme's public key PK_{TDC}, the Verifier's trapdoor commitment $TDCMT$, as well as the newly received challenge CHA_{V} and random coins from the Verifier R_{C} to reveal the commitment for verification.
5) If verification of the commitment is an $accept$, Prover will then calculate the response RSP for the ABID scheme and send it to the Verifier. Otherwise it aborts.
6) Verifier then outputs the decision on whether to accept the Prover's response or not.</td></tr>
</table>

Table 1: Generic Construction of 3-move Reset-Secure ABID Scheme

Informally, the trapdoor commitment generated using a pre-determined challenge by the Verifier serves to fix the commitment value to be used by the Prover. Later on, when this pre-determined challenge is revealed as the challenge from the Verifier, the Prover then verifies that it was indeed the committed value by the Verifier before continuing with its response. If the Prover is reset to the commitment state, it cannot continue with a different challenge from the Verifier (which normally exposes the user secret key) due to the fact that the trapdoor commitment verification stage will fail.

The construction and provable security of a concrete scheme is currently a work-in-progress.

4. CONCLUSION

In this paper, we provided a review of the security notions of reset-secure identification as well as ABID schemes. We also provided a brief survey of all the work currently done in both reset-secure identification schemes as well as attribute-based identification schemes. Then, we gave a generic construction to modify three-move ABID schemes to be reset secure. Future work would include providing detailed proof of security as well as a concrete construction as a case study for the transformation work.

5. ACKNOWLEDGEMENTS

This research is partially supported by Grants-in-Aid for Scientific Research (Research Project Number:25540004) by the Japan Society for the Promotion of Science and The Fundamental Research Project Scheme (No.: FRGS/2/2013/ICT07/MMU/03/5) by the Ministry of Education of the Government of Malaysia.

REFERENCES

Fiat, A., and Shamir, A. 1986. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. *In proceedings of CRYPTO 1986*: 186-194.

Feige, U., and Shamir, A. 1990. Witness Indistinguishable and Witness Hiding Protocols. *In proceedings of STOC 1990*: 416-426.

Bellare, M., Fischlin, M., Goldwasser S., and Micali, S. 2001. Identification Protocols Secure against Reset Attacks. *In proceedings of EUROCRYPT 2001*: 495-511.

Shamir, A. 1984. Identity-Based Cryptosystems and Signature Schemes. *In proceedings of CRYPTO 1984*: 47-53.

Boneh, D. and Franklin, M.K. 2001. Identity-Based Encryption from the Weil Pairing. *In proceedings of CRYPTO 2001*: 213-229

Bellare, M., Namprempre, C. and Neven., G. 2004. Security Proofs for Identity-Based Identification and Signature Schemes. *In proceedings of EUROCRYPT 2004*: 268-286.

Kurosawa, K. and Heng, S.-H. 2004. From Digital Signature to ID-based Identification/Signature. *In proceedings of Public Key Cryptography 2004*: 248-261.

Thorncharoensri, P., Susilo, W., and Mu, Y. 2009. Identity-based identification scheme secure against concurrent-reset attacks without random oracles. In *Information Security Applications* (pp. 94-108). Springer Berlin Heidelberg.

Al-Riyami, S.S. and Paterson, K.G. 2003. Certificateless Public Key Cryptography. *In proceedings of ASIACRYPT 2003*: 452-473.

Dehkordi, M.H. and Alimoradi, R. 2013. Certificateless Identification Protocols from Super Singular Elliptic Curve. *Security and Communication Networks*, (7)6:979-986.

Chin, J.-J., Phan, R. C.-W., Behnia, R. and Heng, S.-H. 2013. An Efficient and Provably Secure Certificateless Identification Scheme. *In proceedings of SECRYPT 2013*: 371-378.

Chin, J.-J., Phan, R. C.-W., Behnia, R. and Heng, S.-H. 2014. Cryptanalysis of a Certificateless Identification Scheme. *Security and Communication Networks*, (7)4: Early View.

Anada, H., Arita, S., Handa, S., and Iwabuchi, Y. 2013. Attribute-based Identification: Definitions and Efficient Constructions. *In proceedings of ACISP 2013*: 168-186.

Sahai, A. and Waters, B. 2005. Fuzzy Identity-based Encryption. *In proceedings of EUROCRYPT 2005*: 457-473.

Waters, B. 2011. Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient and Provably Secure Realization. *In proceedings of PKC 2011*: 53-70.

Anada, H., Arita, S., and Sakurai K. 2014(a). Attribute-Based Identification Schemes of Proofs of Knowledge. In proceedings of SCIS2014, 3E3-3.

Anada, H., Arita, S., and Sakurai, K. 2014(b). Attribute-Based Signatures without Pairings via the Fiat-Shamir Paradigm. To appear in ASIAPKC2014.

Cramer, R., Damgard, I and Nielsen, J.B. 2001 Multiparty computation from threshold homomorphic encryption. *In the proceedings of EUROCRYPT 2001*: 280-300.

Damgard, I. and Nielsen., J.B. 2004. On Sigma-protocols. *Lecture notes on Cryptologic Protocol Theory*, 2010.

Bellare, M. and Palacio, A. 2002. GQ and Schnorr Identification Schemes: Proofs of Security against Impersonation under Active and Concurrent Attacks. *In the proceedings of CRYPTO 2002*: 162-177

Arita S. 2012. A Constant-Round Resettably-Sound Resettable Zero-Knowledge Argument in the BPK Model. *IEICE Transactions 2012*, (95-A)8: 1390-1401.

Canetti, R., Goldwasser, S., Goldriech, O. and Micali, S. 2000. Resettable zero-knowledge. *In the Proceedings of ACM 2000*.

An Identification Scheme Based On Bivariate Function Hard Problem

[1]Tea Boon-Chian and [2] Muhammad Rezal Kamel Ariffin
[1,2]Al-Kindi Cryptography Research Laboratory,
Institute for Mathematical Research,
Universiti Putra Malaysia (UPM), Selangor, Malaysia,
[2]Department of Mathematics, Faculty of Science,
Universiti Putra Malaysia (UPM), Selangor, Malaysia
Email: [1]teaboonchian@ymail.com, [2] rezal@putra.upm.edu.my

ABSTRACT

Recently, an identification scheme based on Diophantine Equation Hard Problem is proposed. The scheme was proven secure against impersonation under passive attack, assuming that solving Diophantine Equation Hard Problem is hard. However, further simulation showed that there exists non-negligible probability that the scheme can be easily impersonated without knowing the secret parameter. In this paper, another identification scheme based on Bivariate Function Hard Problem which is specific problem of Diophantine Equation Hard Problem (DEHP) is proposed. We prove by assuming that solving Bivariate Function Hard Problem is hard, our scheme is secure against the impersonation under passive, active and concurrent attacks. Also, our proposed is again more efficient than some selected schemes since the structure of our hard problem is similar to DEHP.

Keywords: Identification scheme, security analysis, Diophantine Equation Hard Problem, Bivariate Function Hard Problem.

1. INTRODUCTION

An identification scheme enables a prover to identify himself to the verifier without revealing important (private) information throughout the interaction. Typical identification scheme consists of 3-canonical moves that begin with the initiation of the commitment by the prover that binds the communication to the verifier; upon receiving the challenge output by the verifier, prover responses to the challenge and finally decision by the verifier whether accepts or rejects.

The goal of the adversary in an identification scheme is to impersonate such that it behaves as a cheating prover and succeeds in convincing the honest verifier to accept him. Three different attacks of passive, active and concurrent are commonly considered in discussing the impersonation attempts. For this reason, provable security becomes a critical criterion in designing a secure identification scheme.

2. RELATED WORK AND OUR CONTRIBUTION

Tea *et al.* proposed an identification scheme based on Diophantine Equation Hard Problem (DEHP) in 2013 (Tea *et al.*, 2013). The scheme was proven to be secure against impersonation under passive attack assuming that solving DEHP is hard. However, further analysis and simulation showed that the solution to the DEHP in the form of $U = \sum_{i=1}^{2} v_i x_i$ is unique with high probability, yet with carefully selects the interval of the solution as given in the proof of the paper, there is possibility that an impersonator can successfully impersonate the scheme without knowing the secret parameters, and resulting in acceptance by the verifier.

In this paper, we propose a newly design of identification scheme based on the Bivariate Function Hard Problem (BFHP), which is a specific form of Diophantine Equation Hard Problem (DEHP) proposed by Ariffin *et al.* in 2013 (Ariffin *et al.*, 2013). The structure of BFHP chosen is improvised such that solving the BFHP is infeasible and the secret prf-solution is unique. We show that our scheme is secure against impersonation under passive, active and concurrent attacks, assuming that solving BFHP is hard. Efficiency analysis is given to show that our new proposed scheme has similar efficiency as the previous scheme based on DEHP, which is still more efficient than selected schemes chosen.

The layout of the paper is as follows. In Section 3, we review the definition of BFHP. In Section 4 we propose the newly designed identification scheme based on BFHP, followed by the security analysis against the impersonation under passive, active and concurrent attacks in Section 5. In section 6 we provide efficiency analysis of our scheme. In Section 7, the conclusion about our identification scheme is made.

3. PRELIMINARIES

3.1 Bivariate Function Hard Problem (BFHP)

The following proposition gives a proper analytical description of the Bivariate Function Hard Problem (BFHP).

Definition 3.1 We define $\mathbb{Z}^+_{(2^{m-1},2^m-1)}$ as a set of positive integers in the interval $(2^{m-1}, 2^m - 1)$. In other words, if $x \in \mathbb{Z}^+_{(2^{m-1},2^m-1)}$, x is an *m*-bit positive integer.

Proposition 3.1 (Ariffin *et al.*, 2013)
Let $F(x_1, x_2, \dots, x_n)$ be a multiplicative one-way function that maps $F: \mathbb{Z}^n \to \mathbb{Z}^+_{(2^{m-1},2^m-1)}$. Let F_1 andF_2 be such function (either identical or non-identical) such that$A_1 = F(x_1, x_2, \dots, x_n), A_2 = F(y_1, y_2, \dots, y_n)$ and $\gcd(A_1, A_2) = 1$. Let $u, v \in \mathbb{Z}^+_{(2^{n-1},2^n-1)}$. Let (A_1, A_2) be public parameters and (u, v) be private parameters.
Let

$$G(u,v) = A_1 u + A_2 v \tag{1}$$

with the domain of the function G is $\mathbb{Z}^2_{(2^{n-1},2^n-1)}$ since the pair of positive integers $(u,v) \in \mathbb{Z}^2_{(2^{n-1},2^n-1)}$ and $\mathbb{Z}^+_{(2^{m+n-1},2^{m+n}-1)}$ is the codomain of G since $A_1 u + A_2 v \in \mathbb{Z}^+_{(2^{m+n-1},2^{m+n}-1)}$.

If at minimum $n - m - 1 = k$, where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, it is infeasible to determine (u, v) over $\mathbb{Z}$ from $G(u, v)$. Furthermore, $(u, v)$is unique for $G(u, v)$ with high probability.

Remark 3.1 We remark that the preferred pair (u, v) in $\mathbb{Z}$, is the *prf*-solution for (1). The preferred pair (u, v) is one of the possible solutions for (1) given by

$$u = u_0 + A_2 t \tag{2}$$

and

$$v = v_0 - A_1 t \tag{3}$$

for any $t \in \mathbb{Z}$.

Remark 3.2 Before we proceed with the proof, we remark here that the diophantine equation given by $G(u, v)$ is solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found. That is the BFHP is *prf*-solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found.

Proof. We begin by proving that (u, v) is unique for each $G(u, v)$ with high probability. Let $u_1 \neq u_2$ and $v_1 \neq v_2$ such that

$$A_1 u_1 + A_2 v_1 = A_1 u_2 + A_2 v_2 \tag{4}$$

We will then have

$$Y = v_2 - v_1 = \frac{A_1(u_1 - u_2)}{A_2}$$

Since $\gcd(A_1, A_2) = 1$ and $A_2 \approx 2^n$, then the probability that Y is an integer is 2^{-n}. Then the probability that $v_1 - v_2$ is an integer solution not equal to zero is 2^{-n}. Thus $v_1 = v_2$ with probability $1 - \frac{1}{2^n}$.

Next we proceed to prove that to *prf*-solved the Diophantine equation given by (1) is infeasible to be solved. The general solution for $G(u, v)$ is given by (2) and (3) for some integer t.
To find u within the stipulated interval $u \in (2^{n-1}, 2^n - 1)$ we have to find the integer t such that the inequality $2^{n-1} < u < 2^n - 1$ holds. This gives

$$\frac{2^{n-1} - u_0}{A_2} < t < \frac{2^n - 1 - u_0}{A_2}$$

Then, the difference between the upper and the lower bound is

$$\frac{2^{\,n} - 1 - 2^{n-1}}{A_2} = \frac{2^{n-1} - 1}{A_2} \approx \frac{2^{n-2}}{2^m} = 2^{n-m-2}$$

Since $n - m - 1 = k$ where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, we conclude that the difference is very large and finding the correct t is infeasible. This is also the same scenario for v.

Example 3.1 Let $A_1 = 191$ and $A_2 = 229$. Let $u = 41234$ and $v = 52167$. Then $G = 19821937$. Here we take $m = 16$ and $n = 8$. We now construct the parametric solution for this BFHP. The initial points are $u_0 = 118931622$ and $v_0 = -99109685$. The parametric general solution are:$u = u_0 + A_2 t$ and $v = v_0 - A_1 t$. There are approximately $286 \approx 2^9$ (i.e. $\frac{2^{16}}{229}$) values of t to try (i.e. difference between upper and lower bound), while at minimum the value is $t \approx 2^{16}$. In fact, the correct value is $t = 519172 \approx 2^{19}$.

Definition 3.2 We say that the BFHP is hard to be *prf*-solved if for all probabilistic polynomial time algorithm there exist a negligible function $\varepsilon(n)$ such that $Pr[BFHP_{solve} = 1] \leq \varepsilon(n)$.

3.2 The One-More Bivariate Function Hard Problem

Dealing with the impersonation under active and concurrent attackers by issuing valid transcript is no longer possible due to active interaction between both parties in the protocol. Hence, we introduce the One-More Bivariate Function Hard Problem (OMBFHP) which is similar as in GQ and Schnorr's schemes and it will be used in providing security analysis in the later section.

Definition 3.3 Suppose given a challenge oracle $(ChaO)$ that produces a random integer of G_t upon queried and a BFHP Oracle $(BFHPO)$ that output the *prf*-solution of (v_1, v_2, v_3, x) corresponding to the query G_t where $G_t = v_{t,1} + v_2$. Then the adversary is said to successfully impersonate if given t queries, he is able to solve all the t challenges with at most $t - 1$ queries to the BFHP Oracle.

3.3 Security Model of Identification Scheme

Security of an identification schemes relies on the probability of the impersonation by the adversary in which after certain interactions between the adversary and the honest verifier, the adversary succeeds in the impersonation attempt and convinces the verifier to accept with non-negligible probability.

Following denotes the three adversaries that commonly considered in the impersonation of an identification scheme:

1. **Passive Attacker**: The passive adversary eavesdrops on conversations between an honest prover and verifier to acquire information (usually conversation transcript).
2. **Active Attacker**: The active adversary interacts with honest prover sequentially as a cheating verifier to acquire information before attempting impersonation.
3. **Concurrent Attacker**: A special type of active adversary where it can interact with multiple provers at the same time to acquire information before attempting impersonation.

The impersonation attack between an impersonator and the challenger in an identification scheme is based on the two-phase game:

1. **Setup**. The challenger runs *Key Generation* upon taken in the security parameter. The resulting public system parameters are given to the impersonator while the master secret is kept to itself.
2. **Phase 1**. In this phase, the impersonator plays the role as a cheating verifier and can issue transcript queries to the challenger. The challenger responds by returning the commitment, challenge and corresponding response (valid transcript) to the impersonator. These queries are interleaved and asked adaptively.
3. **Phase 2**. The roles are exchanged where impersonator now behaves as a cheating prover and output a challenge which it wishes to impersonate and tries to convince the verifier to accept him. Impersonator is said to win the game if it successfully convinces the verifier in accepting it.

An identification scheme is said to be (t, q_t, ε)-secure under passive, active and concurrent attacks for any non-adaptive passive impersonator I who runs in time t, $Pr[I\ impersonates] < \varepsilon$, where I can makes at most q_t queries.

4. IDENTIFICATION SCHEME BASED ON BIVARIATE FUNCTION HARD PROBLEM (BFHP)

In this section, we propose the construction of identification scheme based on BFHP, assuming that solving BFHP is hard.

Key Generation:

1. Generates the secret parameters of $\{v_i\}_{i=1}^{3} \in \mathbb{Z}_{(2^{n-1}, 2^n - 1)}$
2. Computes the parameters of $G = v_1 + v_2, e_1 = v_3 - v_1, e_2 \equiv v_1^{-1} (mod\ e_1)$, and $x \equiv 1 - v_3^{-1}\ (mod\ G)$ with condition that $x \in \mathbb{Z}_{e_1}$. Otherwise repeat the process.
3. Publicize $\{G, e_1, e_2\}$ and keep $\{v_1, v_2, v_3, x\}$ secret.

Identification Protocol:

Prover, P		Verifier, V
$y \overset{R}{\leftarrow} \mathbb{Z}_{(2^{n-1}, 2^n - 1)}$ $Y \leftarrow y + v_2$	$\xrightarrow{Y}$	$c \overset{R}{\leftarrow} \{0,1\}$
	$\xleftarrow{c}$	
$z \leftarrow v_3 x^c - y$	$\xrightarrow{z}$	1. If $c = 0$ and $e_1 - z - Y \equiv 0\ (mod\ G)$, or 2. If $c = 1$ and $e_1 - z - Y \equiv 1\ (mod\ G)$, then accept else reject.

Completeness.
The following shows the completeness of the identification process:

$$\begin{aligned} e_1 - z - Y &= e_1 - (v_3x^c - y) - (y + v_2) \\ &= v_3 - v_1 - v_3x^c + y - y - v_2 \\ &= v_3 - v_3x^c - v_1 - v_2 \\ &= v_3(1 - x^c) - (v_1 + v_2) \\ &\equiv v_3(1 - x^c)\ (mod\ G). \blacksquare \end{aligned}$$

Remark 4.1
The scheme which relies on the BFHP as its underlying source of security can be understood as follows:

1. It is the main objective if the prover to prove that the variable *x* is known – without relaying the variable *x* to the verifier.
2. The public parameters published given by $\{G, e_1, e_2\}$ contain private parameters given by $\{v_1, v_2, v_3, x\}$.
3. In order for the scheme to remain secure it should not be feasible for an adversary to extract $\{v_1, v_2, v_3, x\}$ from $\{G, e_1, e_2\}$.
4. Observe the public parameters:

$$G \quad = v_1 + v_2 \tag{5}$$

$$e_1 \quad = v_3 - v_1 \tag{6}$$

$$e_2 \quad \equiv v_1^{-1}(mod\ e_1) \tag{7}$$

where (3) can be re-written as

$$e_2v_1 = 1 + e_1k \text{ for some } k \in \mathbb{Z}\ \&\ k{\sim}2^n \tag{8}$$

5. Each individual equations (5),(6) and (8) have the private parameters protected by the BFHP.
6. Furthermore, it is a system of three equations (5), (6) and (8) with 4 variables $\{v_1, v_2, v_3, k\}$.
7. If the 4 unknown variables $\{v_1, v_2, v_3, k\}$ are obtained from the public parameters (as in point 6), we say the scheme has been *prf*-solved. That is, the adversary can obtain $x \equiv 1 - v_3^{-1}\ (mod\ G)$.
8. We can understand point 7 more by the following explanation. The adversary is surely is able to successfully obtain $x \in \mathbb{Z}_{e_1}$(since this is the required condition upon x during key generation) by computing $x \equiv 1 - v_3^{-1}\ (mod\ G)$ by utilizing the preferred integer v_3.

Remark 4.2
We suppose that there exists another $v_3' \neq v_3$ such that $x = -(v_3')^{-1} + 1(mod\ G)$, which output the same *prf*-solution of x. Then it follows that we have

$$-v_3 + 1 \equiv -\left(v_3'\right)^{-1} + 1(\text{mod } G)$$

or

$$(v_3')^{-1} - v_3^{-1} \equiv 0(\text{mod } G)$$

It can be seen that $\frac{v_3' - v_3}{v_3'v_3} \equiv 0$ (mod G). Let $\Delta = \frac{1}{v_3'v_3}$ (mod G) in order for

$$\frac{(v_3' - v_3)\Delta}{G} \in \mathbb{Z}$$

we must have either $G|v_3' - v_3$ or $G|\Delta$. Since $v_3', v_3 \in \mathbb{Z}_G$ which results $G \nmid v_3' - v_3$ so it must be the case that $G|\Delta$. But $\Delta \in \mathbb{Z}_G$ too. Therefore $G \nmid \Delta$ too. This implies that $\frac{(v_3'-v_3)\Delta}{G} \notin \mathbb{Z}$. Hence, it proved that there does not exists $v_3' \neq v_3$ such that $x = -(v_3')^{-1} + 1(mod\ G)$, which output the same *prf*-solution of x.

5. SECURITY ANALYSIS OF IDENTIFICATION SCHEME BASED ON BFHP

5.1 Security Analysis against Impersonation under Passive Attack

Theorem 1. The identification scheme based on the BFHP is (t, q_t, ε)-secure against impersonation under passive attack assuming that solving the BFHP is (t', ε')-hard where:

$$\varepsilon \leq \sqrt{\varepsilon'} + \frac{1}{q}.$$

Proof. We prove by assuming that if there exists an impersonator who can (t, q_t, ε)-break the scheme, then there exists an efficient probabilistic polynomial time algorithm S that (t', q_t, ε')-solve the BFHP. S is then attempt to simulate a challenger for I.

In the beginning stage in Phase 1, S random chooses a set of public parameters $\{G, e_1, e_2\}$ and sends it to impersonator I. It should be reminded that S does not know all the secret parameters of $\{v_1, v_2, v_3, x\}$. Next, upon queried by I, S returns a valid transcript with z and $c \in \{0,1\}$ such that

$$\{R \equiv e_1 - z - v_3(1 - x^c)\ (mod\ G), c, z\}.$$

The transcript query above is continued until some time t where I is ready to challenge and impersonate.

After leaving Phase 1, I is assumed to behave as a cheating prover that tries to convince S. By resetting I to the commitment phase with two different challenges c_1 and c_2, responding to S with two responses z_1 and z_2 corresponds to c_1 and c_2 respectively enable S to obtain two valid transcripts of

$$\{R, c_1, z_1\} \text{ and } \{R, c_2, z_2\}.$$

S will next extract the secret of x by the following calculation

$$x \equiv (z_2 - z_1)e_2 + 1(mod\ e_1)$$

which has the probability more than $\left(\varepsilon - \frac{1}{q}\right)^2$ by Reset Lemma. Let $c_1 = 0$ and $c_2 = 1$. The extraction of the secrets above is known as soundness of the identification scheme in which its correctness can be proven as follows:

$$\begin{aligned} x &= (z_2 - z_1)e_2 + 1 \\ &= (v_3x^{c_2} - y - v_3x^{c_1} + y)e_2 + 1 \\ &= v_3(x^{c_2} - x^{c_1})e_2 + 1 \\ &= v_3e_2(x - 1) + 1 \\ &\equiv (x - 1) + 1\ (mod\ e_1) \\ &\equiv x\ (mod\ e_1). \end{aligned}$$

This completes the description of the simulation.

The analysis of the probability for the identification against impersonation under passive attack remains on simulator S winning the game and solves the BFHP. Let $\varepsilon = Adv_A^{imp-pa}(n)$ be the success probability of the impersonation under active and concurrent attacks and $\varepsilon' = Adv^{BFHP}(n)$ be the probability of the simulator S winning the game by solving the BFHP,

$$Pr[S\ solves\ BFHP] = Pr[S\ computes\ x]$$

$$\varepsilon' \geq \left(\varepsilon - \frac{1}{q}\right)^2$$

$$\varepsilon \leq \sqrt{\varepsilon'} + \frac{1}{q}$$

$$Adv_A^{imp-pa}(n) \leq \sqrt{Adv^{BFHP}(n)} + \frac{1}{q}. \blacksquare$$

5.2 Security Analysis against Impersonation under Active and Concurrent Attack

Theorem 2. The identification scheme based on the BFHP is (t, q_t, ε)-secure against impersonation under active and concurrent attacks assuming that solving the OMBFHP is (t'', ε'')-hard where:

$$\varepsilon \leq \sqrt{\varepsilon''} + \frac{1}{q}\ .$$

Proof. By assuming if there exists an Impersonator I who can (t, q_t, ε)-break the scheme, then there exists an efficient probabilistic polynomial time algorithm S that can $(t'', q_t, \varepsilon'')$-solve the OMBFHP with aid from I. S is then attempt to simulate a challenger for I.
In Phase 1, S access and query the Challenge Oracle $(ChaO)$ to obtain the first initial challenge set of $G_0 = v_{0,1} + v_2, e_1 = v_3 - v_2,$ and $e_2 \equiv v_{0,1}^{-1}(mod\ e_1)$. These public parameters $\{G_0, e_1, e_2\}$ are sent to I. S does not know the secret parameters of $\{v_{0,1}, v_2, v_3, x\}$.

In the identification query, I will first plays the role as cheating verifier to request S to prove itself:

Commitment: Upon queried t by I, S will query $ChaO$ for random challenge set of $G_t = v_{t,1} + v_2$ and sends it to I.

Challenge: I selects a random challenge $c_t \in \{0,1\}$ upon receiving the G_t and sends it to S.

Response: Upon accepting the challenge of c from I, S queries $BFHPO$ with $(e_1 - G_t - c_t)$ and then sends the result $z_t = v_3 x^{c_t} - v_{t,1}$ to I. Then, S increase t by 1.

The identification query above is carried on until some time t where I is ready to enter the challenge and impersonation phase.

Termination Phase 1 changes the role of I to be a cheating prover. By resetting I to the commitment phase in with two different challenges c_1 and c_2, S will able to obtain two valid transcripts of

$$\{R, c_1, z_1\} \text{ and } \{R, c_2, z_2\}$$

The extraction of the secret of x is then done as similarly as in Theorem 1, then followed by computing $v_{0,1}, v_2, v_3$ which then are used to solve all $v_{t,1}$.

$$v_3 = (1 - x)^{-1}\ (mod\ G_0)$$

$$v_{0,1} = v_3 - e_1$$
$$v_2 = G_0 - v_{0,1}$$
$$v_{t,1} = G_t - v_2$$

This completes the description of the simulation.

The analysis of the probability for the identification against impersonation under active and concurrent attacks accounts on S winning the game and solves OMBFHP with strictly less queries to $BFHPO$ than $ChaO$. Let $\varepsilon = Adv_A^{imp-aa/ca}(n)$ be the success probability of the impersonation under active and concurrent attacks and $\varepsilon'' = Adv^{OMBFHP}(n)$ be the probability of the simulator S winning the game by solving the OMBFHP,

$$Pr[S\ solves\ OMBFHP] = Pr[S\ computes\ x]$$
$$\varepsilon'' \geq \left(\varepsilon - \frac{1}{q}\right)^2$$
$$\varepsilon \leq \sqrt{\varepsilon''} + \frac{1}{q}$$

$$Adv_A^{imp-aa/ca}(n) \leq \sqrt{Adv^{OMBFHP}(n)} + \frac{1}{q}. \blacksquare$$

6. EFFICIENCY ANALYSIS

The efficiency in terms of running steps in each protocol of the proposed identification scheme based upon BFHP and some selected schemes based on different hard problems are given in the following Table 1 (Fiat and Shamir, 1986; Guillou and Quisquater, 1988; Schnorr, 1989):

	BFHP		
	Addition	Multiplication	Exponentiation
KeyGen	3	4	0
Prove	2	2	0
Verify	2	1	0

	Guillou-Quisquater (GQ)		
	Addition	Multiplication	Exponentiation
KeyGen	0	1	1
Prove	0	1	2
Verify	0	1	2

	Fiat-Shamir (FS)		
	Addition	Multiplication	Exponentiation
KeyGen	0	1	1
Prove	0	1	2
Verify	0	1	2

	Schnorr		
	Addition	Multiplication	Exponentiation
KeyGen	0	0	1
Prove	1	1	1
Verify	0	1	2

Table 1: Complexity comparison of identification schemes based on 4 different hard problem assumptions.

It can be seen that our identification scheme based on BFHP is more efficient than the other three selected schemes due to only simple addition and multiplication operations with no exponentiation and pairing operations.

7. CONCLUSION

We have proposed the new identification scheme based on new hard problem of Bivariate Function Hard Problem. The security analysis proved that our new proposed scheme is secure against impersonation under passive, active and concurrent attacks, assuming that solving BFHP is hard. Also, the efficiency analysis provided in section 5 shows that the running complexity of our new proposed scheme is smaller than three selected schemes based on different hard problems such as Fiat-Shamir, Guillou-Quisquater and Schnorr's. Hence, in choosing a desirable scheme, our scheme will be a more preferable choice.

8. ACKNOWLEDGEMENTS

The author would like to thank the Institute for Mathematical Research (INSPEM) and Universiti Putra Malaysia (UPM) for giving the opportunity to conduct this research.

REFERENCES

Ariffin, M.R.K., Asbullah, M.A., Abu, N.A. and Mahad, Z. 2013. *A New Efficient Asymmetric Cryptosystem Based on the Integer Factorization Problem of* $N = p^2q$. Malaysian Journal of Mathematical Sciences 7(S): 19-37.

Bellare, M. and Palacio, A. 2002. *GQ and Schnorr Identification Schemes: Proofs of Security against Impersonation under Active and Concurrent Attacks.* Advances in Cryptology, CRYPTOLOGY '02 2442, pp. 162–177.

Fiat, A. and Shamir, A. 1986. *How to Prove Yourself: Practical Solutions to Identification and Signature Problem*. Advances in Cryptology, CRYPTO '86 263, pp. 186–194.

Guillou, L. and Quisquater, J.J. 1988. *A "Paradoxical" Identity-Based Signature Scheme Resulting from Zero Knowledge*. Advances in Cryptology – CRTYPTO '88 403, pp. 216–231.

Schnorr, C.P. 1989. *Efficient Identification and Signature for Smart Card*. Advances in Cryptology, CRYPTO '89 435, pp. 239–252.

Tea, B.C., Ariffin, M.R.K. and Chin, J.J. 2013. *An Efficient Identification Scheme in Standard Model Based on the Diophantine Equation Hard Problem*. Malaysian Journal of Mathematical Sciences 7(S): 87-100.

Implementation of Identity-Based and Certificateless Identification on Android Platform

[1]Ji-Jian Chin, [2]Syh-Yuan Tan, [1]Yvonne Hwei-Syn Kam, and [2]Chee-Hoe Leong
[1]Faculty of Engineering, Multimedia University
[2]Faculty of Information Science and Technology, Multimedia University
Email: [1]{hskam,jjchin}@mmu.edu.my, [2]sytan@mmu.edu.my, eremitic.elvin@gmail.com

ABSTRACT

An identification scheme provides an access control mechanism where a prover authenticates himself to a verifier without providing the verifier with any information about his private key. Recently, pairing-based identification schemes have gained interest, particularly identification schemes without certificates. However there have been little results of implementation of these schemes on handheld mobile devices. In this paper, we provide implementation results for identification schemes without certificates that utilize pairings, on the Android platform.

Keywords: Implementation, simulator, identity-based, certificateless, identification

1. INTRODUCTION

First proposed by Fiat and Shamir (1986), an identification schemes is a cryptographic technique for a user to authenticate himself to a verifier securely. However, standard identification schemes, like other public key cryptographic primitives, require the use of a certificate to bind a public key to a user. This circumvents public key replacement attacks. However, as the number of users grows large for a cryptosystem, this becomes taxing for the Certificate Authority issuing and managing these certificates as the overhead for certificate management increases drastically.

Identity-based cryptography, first proposed by Shamir (1984) as an alternative to public key cryptography, resolves the certificate management problem by implicit certification of users through the user's identity-string. The identity-string acts a public identifier, binding a user to his user secret key. Identity-based identification (IBI) schemes, first proposed by Kurosawa and Heng (2004) and Bellare *et al.* (2004) independently, were first obtained through transformations from digital signatures and standard identification schemes. Kurosawa and Heng (2004) proposed a transformation that converts any digital signature secure against existential forgery under chosen message attacks to be a passive-secure IBI scheme. Bellare *et al.* (2004) on the other hand expanded standard identification schemes that have a trapdoor sampleable relation into identity-based schemes. Later on, Kurosawa and Heng (2005) proposed the first IBI scheme to be proven secure in the standard model while Chin *et al.* (2008) proposed the first IBI scheme provably secure in the standard model to be proven secure based on a hard-problem (non-transformation technique).

Identity-based cryptographic schemes still suffer from key escrow, where the key generation center has access to all users' secret keys. Certificateless cryptography, first introduced by Alriyami and Paterson, (2003), resolves the key escrow issue by allowing the user to create part of the private key to be combined with the key generation's part, thus generating a full private key that the key generation center does not have access to. The first certificateless identification (CLI) schemes were proposed by Dehkordi and Alimoradi (2013) and Chin *et al.* (2013) independently. However, Dehkordi and Alimoradi (2013) did not provide any proofs of security for their schemes, and later on flaws were discovered on their scheme design by Chin *et al.* (2014). Thus the only remaining CLI scheme that remains provable secure is Chin *et al.* (2013)'s scheme.

As developments advanced in the area of IBI and CLI schemes, Tan *et al.* (2009) pioneered the work for implementation using Java. The authors later expanded their work to encompass more schemes in subsequent work. (Tan *et al.*, 2010). However, since CLI schemes are newly introduced, no implementation results exist yet on the schemes.

Since the use of mobile devices is ever expanding in this day and age, it would be of interest to provide implementation results of secure identification schemes on these platforms, particularly for identification schemes without certificates since it saves the overhead of managing certificates. The Android platform is also the most popular platforms for mobile phones (Llamas *et al.* 2014). Therefore the motivation of this work is to provide implementation results for identification schemes without certificates on the Android platform as a prototype for mobile device access control in future deployment.

We build two simulators, one to implement and test an IBI scheme and another for a CLI scheme. We obtain the running time for the BLS-IBI scheme, first proposed by Kurosawa and Heng (2005) using their transformation technique. For the CLI result, we provide the running time for the BLS-CLI scheme, the first provable secure CLI scheme proposed by Chin *et al.* (2013). These two schemes were selected as the first prototype schemes because they are pairing-based and provable secure. The main attraction of pairing-based cryptography is that it offers the same security guarantees with much shorter system parameters. For example, a modulus size of 512 bits provides an equivalent 1024-bits of security on pairing-free systems such as RSA.

The rest of the paper is organized as follows: In Section 2, we review the definition and security notions for IBI and CLI schemes. In Section 3, we provide the scheme definition for BLS-IBI and BLS-CLI schemes. In Section 4 we provide our experiment results. We conclude with some remarks in Section 5.

2. PRELIMINARIES

In this section, we briefly review the formal definition and security model for IBI and CLI schemes.

2.1 Identity-Based Identification Schemes

An IBI scheme consists of four probabilistic polynomial-time (PPT) algorithms: SETUP, EXTRACT, PROVER and VERIFIER.

1) SETUP takes in the security parameters and generates the master public key mpk and master secret key msk.
2) EXTRACT takes in the master public key mpk, user identity-string ID and master secret key msk and outputs the user secret key usk.
3) PROVER and VERIFIER both take in the master public key mpk and user identity-string ID. Additionally, PROVER takes in the user secret key usk. They engage in the identification protocol as follows:
   i) PROVER creates and sends a commitment to VERIFIER.
   ii) VERIFIER selects a random challenge and sends it to PROVER.
   iii) PROVER calculates a response based on the challenge and commitment and sends it to VERIFIER. VERIFIER outputs “accept” or “reject” based on the response.

The adversary for IBI schemes is called an impersonator. There are two types of impersonators: the passive impersonator who eavesdrops on conversations between honest provers and verifiers, and the active impersonator who can interact with honest provers as a cheating verifier to learn information before the impersonation attempt. For active impersonators, if the impersonator can run several conversations simultaneously with several honest prover instances, then it is an active-concurrent attacker. Another notion which is stronger is the reset-attacker who can reset prover clones to whatever state within the identification protocol it desires, but this notion is seldom considered by most research.

The security of an IBI scheme can be described as the following game played between a challenger **C** and an impersonator **I**.

Setup: **C** generates the master public key and master secret key. It gives the master public key to **I** and keeps the master secret key to itself.

Phase 1: This is the learning phase. **I** can issue extract queries to obtain user secret keys from **C**. **I** can also issue identification queries in the form of transcript queries for passive impersonators and conversations as the cheating verifier interacting with **C** as the prover for active/concurrent impersonators.

Phase 2: This is the impersonation phase. **I** outputs a challenge identity on which it wishes impersonate. **I** wins if it manages to convince **C** to accept.

We say an IBI scheme is (t, q_I, ε)-secure under passive/active/concurrent attacks if for any impersonator I that runs in time t, $Pr[I \text{ can impersonate}] < \varepsilon$ where I can make at most q_I extract and identification queries.

2.2 Certificateless Identification Schemes

A CLI scheme consists of six PPT algorithms: KEYGEN, PARTIAL-PRIVATE-KEY-EXTRACT, SET-USER-KEY, SET-PRIVATE-KEY, PROVER and VERIFIER.

Similarly the adversary for CLI schemes is an impersonator, passive and active/concurrent. This is defined as a Type-1 adversary modeling a malicious user trying to exploit another user's private key. However, according to certificateless cryptography, we now have to define a new kind of adversary, the Type-2 adversary, modeling an adversarial server trying to exploit a user's private key.

Moreover, since adversaries in the certificateless setting have the capability to replace public keys for any user (except for the challenge identity for Type-2), we have to further divide the types of adversaries into:

a) Normal: cannot make identification queries after replacing the public key on a user instance.
b) Strong: can continue to make identification queries after replacing the public key on a user instance, provided the replaced public key's corresponding secret value is provided.
c) Super: can continue to make identification queries after replacing the public key on a user instance, and does not need to provide the corresponding secret value.

The security of a CLI scheme against a Type-1 impersonator can be described as the following game played between a challenger **C** and an impersonator $\mathbf{I_1}$**.**

KEYGEN: **C** runs KEYGEN and outputs the master public key and master secret key. It gives the master public key to I and keeps the master secret key to itself.

Phase 1: This is the learning phase. $\mathbf{I_1}$ can issue extract queries to obtain partial or full user private keys and user public keys from **C**. $\mathbf{I_1}$ can replace the public keys of users. $\mathbf{I_1}$ can also issue identification queries in the form of transcript queries for passive impersonators and conversations as the cheating verifier interacting with **C** as the prover for active/concurrent impersonators. The requirement of a corresponding public key for identification queries will correspond to the normal/strong/super Type-1 category $\mathbf{I_1}$ is in.

Phase 2: This is the impersonation phase. $\mathbf{I_1}$ outputs a challenge identity on which it wishes to impersonate. $\mathbf{I_1}$ wins if it manages to convince **C** to accept.

We say a CLI scheme is (t, q_I, ε)-secure under passive/active/concurrent attacks if for any normal/strong/super Type-1 impersonator $\mathbf{I_1}$ that runs in time t, $Pr[\mathbf{I_1} \text{ can impersonate}] < \varepsilon$ where $\mathbf{I_1}$ can make at most q_I extract (partial and full private keys) and identification queries.

The security of a CLI scheme against a Type-2 impersonator can be described as the following game played between a challenger **C** and an impersonator $\mathbf{I_2}$**.**

KEYGEN: **C** runs KEYGEN and outputs the master public key and master secret key and a challenge identity's secret value. It gives both the master public key and master secret key to I but keeps the challenge identity's secret value to itself.

Phase 1: This is the learning phase. $\mathbf{I_2}$ can issue extract queries to obtain full user private keys (partial private key extracts are not required since it has access to the master secret key) and user public keys from **C**. $\mathbf{I_2}$ can replace the public keys of users. $\mathbf{I_2}$ can also issue identification queries in the form of transcript queries for passive impersonators and conversations as the cheating verifier interacting with **C** as the prover for active/concurrent impersonators. The requirement of a corresponding public key for identification queries will correspond to the normal/strong/super Type-1 category $\mathbf{I_2}$ is in.

Phase 2: This is the impersonation phase. $\mathbf{I_2}$ outputs a challenge identity on which it wishes impersonate. $\mathbf{I_2}$ wins if it manages to convince **C** to accept.

We say a CLI scheme is (t, q_I, ε)-secure under passive/active/concurrent attacks if for any normal/strong/super Type-2 impersonator $\mathbf{I_2}$ that runs in time t, $Pr[\mathbf{I_2} \text{ can impersonate}] < \varepsilon$ where $\mathbf{I_2}$ can make at most q_I extract (partial and full private keys) and identification queries.

3. BLS-IBI and BLS-CLI SCHEMES

In this section we review the BLS-IBI scheme and the BLS-CLI scheme.

3.1 The BLS-IBI Scheme

The BLS-IBI scheme was first proposed by Kurosawa and Heng (2004) as a transformation from the Boneh-Lynn-Shacham (BLS) signature (Boneh *et al.*, 2001) to a passive-secure IBI scheme. Later on, Heng (2005) provided the proof of security against active and concurrent attacks.

We describe the scheme in detail in Table 1.

Setup($\mathbf{1^k}$)→ $\langle \boldsymbol{mpk}, \boldsymbol{msk} \rangle$ 1) On input 1^k, generate groups G, G_T of prime order q and choose a generator $g \in G$. 2) Choose random $s \overset{\$}{\leftarrow} Z_q$ and let $g_1 = g^s$. 3) Choose a hash function $H: \{0,1\}^* \to G$ 4) Select an efficient pairing $e: G \times G \to G_T$ 5) Publish the master public key $mpk = \langle G, q, e, g, g_1, H \rangle$. Master secret key $msk = \langle s \rangle$ is secret.		
Extract($\boldsymbol{mpk}, \boldsymbol{msk}, \boldsymbol{ID}$)→ $\langle \boldsymbol{usk} \rangle$ 1) On input (mpk, msk, ID), calculate $d = Q^s$ where $Q = H(ID)$ 2) Set the user secret key as $usk = \langle d \rangle$		
Prover($\boldsymbol{mpk}, \boldsymbol{usk}, \boldsymbol{ID}$)		**Verifier**($\boldsymbol{mpk}, \boldsymbol{ID}$)
1) Select $r \overset{\$}{\leftarrow} Z_q$ 2) Compute $U = Q^r$	$\xrightarrow{U}$	
4) $V = d^{r+c}$	$\xleftarrow{c}$	3) $c \overset{\$}{\leftarrow} Z_q$
	$\xrightarrow{V}$	5) Accept if and only if $e(g, V) = e(g_1, UQ^c)$

Table 1: The BLS-IBI scheme

The scheme is provable secure against impersonation under passive attacks if the Computational Diffie-Hellman assumption holds. The scheme is provable secure against impersonation under active and concurrent attacks if the One-More Computational Diffie-Hellman assumption holds.

3.2 The BLS-CLI Scheme

The BLS-CLI scheme was first proposed by Chin *et al.* (2013) as an extension of the BLS-IBI scheme to the certificateless setting. However, the extension was not trivial as it had to be secure against public key replacement attacks, a new kind of attack not applicable to IBI schemes. Security also had to be proven against both Type-1 and Type-2 impersonators.

We describe the scheme in detail in Table 2.

KGC:Keygen(1^k)→ $\langle mpk, msk\rangle$

1. On input 1^k, generate groups G, G_T of prime order q and choose a generator $g \in G$.
2. Choose random $s \overset{\$}{\leftarrow} Z_q$ and let $g_1 = g^s$.
3. Choose a hash function $H: \{0,1\}^* \to G$
4. Select an efficient pairing $e: G \times G \to G_T$
5. Publish the master public key $mpk = \langle G, q, e, g, g_1, H\rangle$. Master secret key $msk = \langle s\rangle$ is secret.

$User_{ID}$:Set-User-Key($1^k, mpk$)→ $\langle sv, upk\rangle$

1. Select $x_{ID} \overset{\$}{\leftarrow} Z_q$ and set the user secret value $sv = \langle x_{ID}\rangle$.
2. Set $upk = \langle X_{1,ID} = g^{x_{ID}}, X_{2,ID} = g_1^{x_{ID}}\rangle$

KGC:PPK-Extract(mpk, msk, ID, upk)→ $\langle ppk\rangle$

1. On input (mpk, msk, ID, upk), calculate $d = Q^s$, where $Q = H(ID, X_{1,ID}, X_{2,ID})$.
2. Set the user partial private key as $ppk = \langle d_{ID}\rangle$

$User_{ID}$:Set-Private-Key(mpk, ID, sv, upk, ppk)→ $\langle usk\rangle$

1. Check if $e(g, d_{ID}) = e(g_1, Q_{ID})$ to verify the partial private key is valid.
2. If valid, calculate $s_{ID} = (d_{ID}Q_{ID})^{x_{ID}}$
3. Set $usk = s_{ID}$

Prover(mpk, upk, ID, usk):		**Verifier(mpk, upk, ID)**
1. Select $r \overset{\$}{\leftarrow} Z_q$	$\overset{U}{\rightarrow}$	
2. Compute $U = Q^r$		
	$\overset{c}{\leftarrow}$	3. $c \overset{\$}{\leftarrow} Z_q$
4. $V = d^{r+c}$		
		5. Accept if and only if
	$\overset{V}{\rightarrow}$	i) $e(g, X_{2,ID}) = e(g_1, X_{1,ID})$
		ii) $e(g, V) = e(g_1, UQ^c)$

Table 2: The BLS-CLI Scheme

The BLS-CLI scheme is provable secure against Super-Type-1 and Super-Type-2 impersonation under passive attacks if the Computational Diffie-Hellman assumption holds, and is secure against Strong-Type-1 and Strong-Type-2 impersonation under active and concurrent attacks if the One-More Computational Diffie-Hellman assumption holds.

4. IMPLEMENTATION RESULTS ON ANDROID

We conduct the simulation using two Android platform mobile phones, the Samsung Galaxy S3 and the Samsung Galaxy Note. The Samsung Galaxy S3 is chosen as the platform to represent the middle to higher range processing capabilities while the Samsung Galaxy Note is chosen as the platform to represent the middle to lower range mobile devices.

For BLS-IBI, we run Extract for 10 times and the Identification Protocol for 100 times, taking the average times. Preprocessing is conducted prior to the pairing operation to speed up the pairing operation. The running time measurements are taken on 3 different settings corresponding to their equivalent level of security on RSA or discrete logarithms:

a) 1024-bit security -Order:160 bits, modulus:512 bits
b) 2048-bit security -Order:224 bits, modulus:1024 bits
c) 3072-bit security -Order:256 bits, modulus:1536 bits

To setup the system on the Samsung Galaxy S3, the average time taken is 4,408,264,461 nanoseconds for 512-bit security, 9,545,250,505 nanoseconds for 1024-bit security and 17,757,422,258 nanoseconds for 1536-bit security.

To setup the system on the Samsung Galaxy Note, the average time taken is 5,930,657,461 nanoseconds for 512-bit security, 25,874,033,970 nanoseconds for 1024-bit security and 18,768,758,093 nanoseconds for 1536-bit security.

The rest of the results of the simulation are displayed in **Table 3**.

Phone	Security Level	Extract Running Time (ns)	Identification Running Time (ns)	Preprocessing Time (ns)
S3	512-bit	882,318,613	1,289,219,686	426,680,084
	1024-bit	3,241,320,184	3,195,771,092	876,529,668
	1536-bit	7,971,923,920	6,242,970,037	1,402,820,125
Note	512-bit	982,694,517	1,377,724,928	557,291,959
	1024-bit	3,367,519,497	3,419,332,517	1,197,303,041
	1536-bit	7,662,051,399	6,402,078,598	1,459,599,500

Table 3: Running time for BLS-IBI Identification Protocol

For BLS-CLI, we run each algorithm in the key generation sequence for a user (Partial-Private-Key-Extract, Set-User-Key and Set-Private-Key) for 10 times each and the Identification Protocol for 100 times, taking the average times. We conduct the measurements on the 3 different settings similar to that of BLS-IBI.

To setup the system on the Samsung Galaxy S3, the average time taken is 2,667,524,250 nanoseconds for 512-bit security, 18,413,465,092 nanoseconds for 1024-bit security and 18,972,480,800 nanoseconds for 1536-bit security.

To setup the system on the Samsung Galaxy Note, the average time taken is 2,393,464,543 nanoseconds for 512-bit security, 14,161,100,673 nanoseconds for 1024-bit security and 17,360,391,841 nanoseconds for 1536-bit security.

The rest of the results of the simulation are displayed in Table 4.

Phone	Security Level	PPK-Extract Running Time (ns)	Set-User-Key Running Time (ns)	Set-Private-Key Running Time (ns)
S3	512-bit	811,130,000	441,430,360	748,263,739
	1024-bit	2,978,704,092	1,249,246,731	1,638,782,099
	1536-bit	7,249,742,014	2,317,421,145	3,020,873,281
Note	512-bit	795,332,038	464,622,295	797,018,030
	1024-bit	2,926,413,081	1,194,256,879	1,816,456,656
	1536-bit	7,221,552,677	2,415,738,546	3,788,430,922

Table 4: Running time for User Key Generation Algorithms for BLS-CLI

Phone	Security Level	Identification Running Time (ns)	Preprocessing Time (ns)
S3	512-bit	1,864,374,851	1,512,641,667
	1024-bit	4,124,783,833	1,993,273,458
	1536-bit	8,248,376,567	2,820,083,585
Note	512-bit	1,997,225,628	1,242,776,334
	1024-bit	4,621,573,606	2,168,200,835
	1536-bit	8,615,053,077	3,936,847,252

Table 5: Running time for BLS-CLI Identification Protocol

According to the results, the running times for the simulations do not differ much on both platforms. The identification protocol for BLS-IBI runs around 1 second at 512-bit level security, proving that pairing-based IBI schemes show certain potential for deployment in the future as an access control mechanism with acceptable authentication time.

The identification protocol for BLS-CLI runs slightly longer due to the necessity to validate the user’s public key in addition to doing the identification response verification check. However, running time is still only around 2 seconds.

5. CONCLUSION

In this paper, we presented the running times for a pairing-based IBI scheme and a pairing-based CLI scheme. Both schemes are provable secure in their respective security models.

At 512-bit security, both schemes perform with promising results. However, as one would expect, efficiency deteriorates with the increase of security. It is also interesting to note that the performance does not suffer even though a mobile device is inferior in computing power, as we can see that the performance between the Samsung Galaxy S3 and the Samsung Galaxy Note does not differ too much and in fact the Galaxy Note at times outperforms the Galaxy S3 depending on the random parameters for curves generated.

An immediate direction for future work for this research would be to improve the simulation environment to take into account processing cycle consumption, battery consumption as well as breaking of the algorithm to actual communication between two devices using NFC, WiFi or Bluetooth. This would better model the practical use of these IBI and CLI schemes, which to date have mainly remained theoretical in nature of research.

Secondly, pairing-free IBI schemes like the GQ and Schnorr IBI schemes should be simulated and compared to the BLS-IBI in terms of running time. Also it would be interesting to see implementation results of any other CLI schemes that have yet to be discovered.

Lastly, it would be interesting to take the development of IBI and CLI schemes to the IOS platform, the second most widely used operating system for mobile platforms (Llamas *et al.*, 2014). The performances of the schemes from both IOS and Android platforms can then be compared to see which is more efficient.

6. ACKNOWLEDGEMENTS

The authors are grateful to the Ministry of Education of the Government of Malaysia for partially funding this research under The Fundamental Research Project Scheme (No.: FRGS/2/2013/ICT07/MMU/03/5).

REFERENCES

Fiat, A. and Shamir, A. 1986. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. *In proceedings of CRYPTO 1986*: 186-194.

Shamir, A. 1984. Identity-Based Cryptosystems and Signature Schemes. *In proceedings of CRYPTO 1984*: 47-53.

Bellare, M., Namprempre, C. and Neven., G. 2004. Security Proofs for Identity-Based Identification and Signature Schemes. *In proceedings of EUROCRYPT 2004*: 268-286.

Kurosawa, K. and Heng, S.-H. 2004. From Digital Signature to ID-based Identification/Signature. *In proceedings of Public Key Cryptography 2004*: 248-261.

Al-Riyami, S.S. and Paterson, K.G. 2003. Certificateless Public Key Cryptography. *In proceedings of ASIACRYPT 2003*: 452-473.

Dehkordi, M.H. and Alimoradi, R. 2013. Certificateless Identification Protocols from Super Singular Elliptic Curve. *Security and Communication Networks*, (7)6:979-986.

Chin, J.-J., Phan, R. C.-W., Behnia, R. and Heng, S.-H. 2013. An Efficient and Provably Secure Certificateless Identification Scheme. *In proceedings of SECRYPT 2013*: 371-378.

Chin, J.-J., Phan, R. C.-W., Behnia, R. and Heng, S.-H. 2014. Cryptanalysis of a Certificateless Identification Scheme. *Security and Communication Networks*, (7)4: Early View.

Tan, S.-Y., Heng, S.-H., Goi, B.-M., Chin, J.-J. and Moon, S.-J. 2009. Java Implementation for Identity-Based Identification. *International Journal of Cryptology Research*, 1(1):21-32.

Tan, S.-Y., Heng, S.-H. and Goi, B.-M. 2010. Java Implementation for Pairing-Based Cryptosystems. *In Proceedings of ICCSA 2010*: 188-198.

Llamas, R., Reith, R. and Shirer, M. 2014. Android and iOS Continue to Dominate the Worldwide Smartphone Market with Android Shipments Just Shy of 800 Million in 2013. *Press release, 12 February 2014* *http://www.idc.com/getdoc.jsp?containerId=prUS24676414*.

Heng, S.-H. 2004. Design and Analysis of Some Cryptographic Primitives. *PhD Thesis*.

Boneh, D., Lynn, B. and Shacham H. 2001. Short Signatures from the Weil Pairing. *In proceedings of ASIACRYPT 2001*: 514-532.

Cryptanalysis of an ID-based Blind Signature Scheme

[1]Syh-Yuan Tan, [2]Wun-She Yap and [2]Bok-Min Goi
[1]Faculty of Information Science and Technology, Multimedia University
[2]Faculty of Engineering and Science, Universiti Tunku Abdul Rahman
Email: [1]sytan@mmu.edu.my, [2]{yapws,goibm}@utar.edu.my

ABSTRACT

In 2010, Rao *et al.* proposed an identity-based blind signature scheme based on bilinear pairings. The proposed scheme is claimed to have achieved blindness and also secure against unforgeability in the generic proofs. In this paper, we show that the security claim is invalid where the IBBS does not achieve blindness, though the signature is unforgeable. To be precise, the signer can link a blinded message to the corresponding signature signed for a user.

Keywords: Cryptanalysis, Identity-based cryptography, blind signature, linkability attack

1. INTRODUCTION

Public key cryptography (PKC) was introduced by Diffie and Hellman to overcome the key distribution problem of symmetric key cryptography. However, PKC suffers from the man-in-the-middle attack where an adversary C can replace the public key of a user A, to impersonate A in communicating with another user B. As B cannot verify the authenticity of A's public key, B will fall prey to C. To solve this problem, public key certification using digital signature scheme is needed where A and B obtain a signature generated by a trusted third party (TTP) on their public keys respectively.

However, when the number of user grows, TTP will face the certificate and public key management issues. In view of this, Shamir (1984) proposed the idea of identity-based cryptography (IBC) which achieves implicit certification through the unique private key generation. The user's identity which is verifiable publicly such as phone number, IC number, email, office room number and so on can be used as the public key. IBC was not realised until the work of Boneh and Franklin (2001) and many identity-based cryptographic primitives were formalised, including identity-based blind signature (IBBS) scheme.

In 2010, Rao *et al.* proposed an IBBS (Rao *et al.*, 2010) scheme and provided a brief security proof, indicated that their IBBS scheme fulfilled the blindness property and is unforgeable. In other words, the IBBS assures that the signer cannot identify which message does a signature belongs to while none can generate a valid signature on the unknown message except the signer himself. We disagree with the security claim by showing that the signer of Rao *et al.*'s IBBS scheme can link the message to its corresponding signature. We note that the unforgeability remains valid though, as the underlying Hess identity-based signature (IBS) scheme is proven secure.

The rest of the paper is organized as follows. In Section 2, we briefly describe the mathematical notations involved and review the definition of IBBS scheme. In Section 3, we present the cryptanalysis result by mounting a linkability attack on the IBBS scheme. We conclude the paper in Section 4.

2. PRELIMINARIES

In this section, we briefly review the definition for bilinear pairing as well as the scheme model and security notion of IBBS scheme.

2.1 Bilinear Pairing

A bilinear pairing is a pairing function e which pairs two elements $P, Q \in G_1$ to an element $Z \in G_2$ such that $e: P \times Q \rightarrow Z$. To be precise, the pairing function also fulfills the following properties:

1. Bilinearlity: $e(aP, Q) = e(P, aQ) = e(P, Q)^a = e(Q, P)^a$.

2. Non-degeneracy: $e(P,P) \neq 1 \neq e(Q,Q)$.
3. Easily computable.

2.2 Identity-Based Blind Signature Scheme

An IBBS scheme consists of four algorithms: Setup, Extract, Blind Issue and Verification.

Setup (1^k): Take as input a security parameter *k* and return the master public key *mpk* and a master secret key *msk*. *mpk* is published while *msk* is kept securely.

Extract (*mpk, msk, ID*): Take as inputs *mpk, msk* and a public identity *ID*. Return the user private key *upk*.

Issue (*mpk, upk, ID*): It is an interactive protocol between a signer and a user. The user is given (*m, mpk, ID*), where *m* is the message he wants to be signed on the identity *ID*. Except the blind message *V',* the signer is given *mpk, upk* and its own identity *ID.* The signer and the user run the blind signature issuing protocol. When they stop, the user outputs a blind signature σ on the identity *ID* and message *m*.

Verification (*mpk, ID, m, σ*): It is a deterministic algorithm that takes as input *mpk*, an identity *ID*, a message *m* and a blind signature *σ*. It outputs either accept or reject.

2.3 Security Requirements of IBBS Scheme

A secure IBBS scheme should achieve the property of blindness and unforgeability against adaptive chosen message attacks. We describe the security definition for the former property only (Heng *et al.*, 2007) since we are particularly dealing with this notion in this paper.

Definition 1. Blindness. Let A be the signer and A is involved in the following game with two honest users, namely U_0 and U_1.
1. (*ID,upk*) ← Extract(*mpk, msk, ID*).
2. (m_0,m_1)←A(*mpk, upk, ID*).
3. Select $i \in \{0,1\}$. Put m_i and m_{l-i} to the read-only input tape of U_0 and U_1
respectively.
4. A engages in the signature issuing protocol with U_0 and U_1 in an arbitrary
order.
5. If U_0 and U_1 output $\sigma(m_i)$ and $\sigma(m_{l-i})$ respectively using their private tapes, then return those outputs to A. Otherwise, return ⊥ to A.
6. A outputs a bit $i' \in \{0,1\}$.
We say that A wins the game if $i' = i$. An IBBS is blind if there is no PPT algorithm A that wins the game with probability at least $1/2 + 1/k^c$ for any constant $c > 0$. The probability is taken over the coin flips of Extract, U_0, U_1 and A.

3. CRYPTANALYSIS RESULT

In this section, we describe the Rao *et al.*'s IBBS scheme before mounting the linkability attack on it.

3.1 Rao et al.'sIBBS Scheme

Table 1 reviews Rao *et al.*'s IBBS scheme.

<table>
<tr><td colspan="3"><b>Setup(1^k)</b>
6) On input1^k, PKG generates groups G_1, G_2 of prime order q and randomly choose a generator $P \in G_1$. The PKG chooses $s \in Z_q^*$ as his master secret key msk and compute P_{pub} as sP. The published mpk is $\{G_1, G_2, e, P, P_{pub}, H_1, H_2\}$ where $H_1: G_1 \rightarrow \{0,1\}^*andH_2: \{0,1\}^* \times G_2 \rightarrow \{0,1\}^*$.</td></tr>
<tr><td colspan="3"><b>Extract(mpk, msk, ID)</b>
1. Given a singer's public identity $ID \in \{0,1\}^*$, compute the public key $Q_{ID} = H_1(ID)$ and the corresponding user private key $usk = d_{ID} = sQ_{ID}$.</td></tr>
<tr><td colspan="3"><b>Issue</b></td></tr>
<tr><td><b>Signer(mpk, upk)</b></td><td></td><td><b>User(m, mpk, ID)</b></td></tr>
<tr><td>1. <b>Initialisation</b>: Randomly select $k \in Z_q^*$ and compute$R = e(P,P)^k$. Sends R to user as commitment.</td><td>$\overset{R}{\rightarrow}$
$\overset{V}{\leftarrow}$</td><td>2. <b>Blinding</b>: Randomly select $a, b \in Z_q^*$ as blinding factors, compute $R' = e\left(bQ_{ID} + aP, P_{pub}\right) \cdot R, V = H_2(m, R') + b$ and sends $V$to signer.</td></tr>
<tr><td>3.<b>Signing</b>: Compute $S = Vd_{ID} + kP$ and sends S to user.</td><td>$\overset{S}{\rightarrow}$</td><td>4. <b>Unblinding</b>: Compute $S' = S + aP_{pub}$, $V' = V - b$ and outputs (m, σ) where $\sigma = (S', V')$ is the blind signature of message m.</td></tr>
<tr><td colspan="3"><b>Verification (mpk, ID, m, S', V')</b>
Accept the signature $\sigma = (S', V')$ if $V' = H_2\left(m, e(S', P)e\left(Q_{ID}, P_{pub}\right)^{-V'}\right)$, rejects otherwise.</td></tr>
</table>

Table 1: Rao *et al.*'s IBBS Scheme

3.2 Linkability Attack

We now show how to mount a linkability attack on Rao *et al.*'s IBBS scheme. Assume the signer is Bob with public identity *ID_{Bob}* and the user is Alice:

1. From Step 1, 2 and 3 in the **Issue** protocol, Bob has the knowledge of (R_0, V_0, S_0) on an unknown message m_i which belongs to Alice where $i \in \{1,0\}$.
2. Alice performs **Unblinding** and publishes $\sigma_0 = (S'_0, V'_0)$ as the signature for her message m_0 which is signed by Bob.
3. At this point, Bob will be exposed to the values of both message m_0 and the corresponding signature σ_0. He can now check whether m_0 is singed by him by performing the following steps:
   a. Extracting the blinding factor b_0:

$$\begin{aligned} V_0 - V'_0 &= V_0 - (V_0 - b_0) \\ &= H_2(m_0, R'_0) + b_0 - (H_2(m_0, R'_0) + b_0 - b_0) \\ &= b_0 \end{aligned}$$

   b. Extracting the value $a_0 P_{pub}$:

$$\begin{aligned} S'_0 - S_0 &= S_0 + aP_{pub} - S_0 \\ &= V_0 d_{ID_{Bob}} + k_0 P + a_0 P_{pub} - \left(V_0 d_{ID_{Bob}} + k_0 P\right) \\ &= a_0 P_{pub} \end{aligned}$$

c. Compute $R_0^* = e\big(b_0 d_{ID} + a_0 P_{pub}, P\big) \cdot R_0$.

d. Check if the condition $H_2(m_0, R_0^*) = V_0'$ holds? If yes, the message *m₀* is signed by Bob himself; else, it is signed by other. The correctness is as follows:

$$\begin{aligned} V_0' &= H_2(m_0, R_0') \\ &= H_2\big(m_0, e\big(b_0 Q_{ID_{Bob}} + a_0 P, P_{pub}\big) \cdot R_0\big) \\ &= H_2\big(m_0, e\big(b_0 Q_{ID_{Bob}} + a_0 P, sP\big) \cdot R_0\big) \\ &= H_2\big(m_0, e\big(s\big(b_0 Q_{ID_{Bob}} + a_0 P\big), P\big) \cdot R_0\big) \\ &= H_2\big(m_0, e\big(b_0 d_{ID_{Bob}} + a_0 P_{pub}, P\big) \cdot R_0\big) \\ &= H_2(m_0, R_0^*) \end{aligned}$$

4. For other message *m₁* which is not signed by Bob, the corresponding signature will be $\sigma_1 = (S'_1, V'_1)$ and the correctness will not hold. When Bob perform the same computations, he will:

   a. fail in extracting the blinding factor b_1:

$$\begin{aligned} V_0 - V'_1 &= V_0 - (V_1 - b_1) \\ &= H_2(m_0, R_0') + b_0 - (H_2(m_1, R_1') + b_1 - b_1 \\ &= b_X \neq b_1 \end{aligned}$$

   b. fail in extracting the value $a_1 P_{pub}$:

$$\begin{aligned} S_1' - S_0 &= S_1 + a_1 P_{pub} - S_0 \\ &= V_1 d_{ID_X} + k_1 P + a_1 P_{pub} - \big(V_0 d_{ID_{Bob}} + k_0 P\big) \\ &= a_X P_{pub} \neq a_1 P_{pub} \end{aligned}$$

   c. Fail in computing $R_1^* \neq R_X^* = e\big(b_X d_{ID_{Bob}} + a_X P_{pub}, P\big) \cdot R_0$.

   d. Obviously the condition $H_2(m_1, R_X^*) \neq V_1'$ does not hold at all now:

$$\begin{aligned} V_1' &= H_2(m_1, R_1') \\ &= H_2\big(m_1, e\big(b_1 Q_{ID_{Other}} + a_1 P, P_{pub}\big) \cdot R_1\big) \\ &= H_2\big(m_1, e\big(b_1 Q_{ID_{Other}} + a_1 P, sP\big) \cdot R_1\big) \\ &= H_2\big(m_1, e\big(s\big(b_1 Q_{ID_{Other}} + a_1 P\big), P\big) \cdot R_1\big) \\ &= H_2\big(m_1, e\big(b_1 d_{ID_{Other}} + a_1 P_{pub}, P\big) \cdot R_1\big) \\ &\neq H_2(m_1, R_X^*) \end{aligned}$$

Therefore, it is obvious that Rao *et al.*'s blindness is broken. The main reason behind the flaw is that Rao *et al.* did not hide well the blinding factor *b*. This factor is easily extractable as shown in the attack above.

There are some instances where linkability attacks were falsified (Heng *et al.*, 2007) but the same falsification does not apply here as the blinding factors are extractable and distinguishable from other blinding factors. Furthermore, Rao *et al.* misinterpreted the non-degeneracy property of bilinear pairing. They showed that $a, b \in Z_q^*$ existed uniquely such that there is always a pair of a, b which satisfies the blinding element R' such that:

$$R' = e(bQ_{ID} + aP, P_{pub}) \cdot R \leftrightarrow e\big(e\big(bQ_{ID} + aP, P_{pub}\big), P_{pub}\big)$$

However, bilinear pairing can only take in elements in G_1as input, not elements from G_2. Hence, the description on the relation between blinding factors a, b and the blinding element R' is flawed as well.

4. CONCLUSION

We mounted a linkability attack on an IBBS scheme and show that the IBBS scheme cannot provide blindness. The reason behind the flaw is that the blinding factors *a*, *b* are not well hidden within the values (S, S') and (V, V') respectively.

REFERENCES

Boneh, D., and Franklin, M. (2001, January). Identity-based encryption from the Weil pairing. In *Advances in Cryptology—CRYPTO 2001* (pp. 213-229). Springer Berlin Heidelberg.

Heng, S. H., Yap, W. S., and Khoo, K. (2007). Linkability of some blind signature schemes. In *Information Security Theory and Practices. Smart Cards, Mobile and Ubiquitous Computing Systems* (pp. 80-89). Springer Berlin Heidelberg.

Shamir, A. (1985, January). Identity-based cryptosystems and signature schemes. In *Advances in cryptology* (pp. 47-53). Springer Berlin Heidelberg.

Rao, B. U., Ajmath, K. A., Reddy, P. V., and Gowri, T. (2010). An ID-based Blind Signature Scheme from Bilinear Pairings. *International Journal of Computer Science and Security (IJCSS)*, *4*(1), 98.

Combined Encryption, Signature and Multisignature Schemes

[1]Moesfa Soeheila Mohamad and [2]Geong Sen Poh
[1]MIMOS Berhad, Technology Park Malaysia, Kuala Lumpur. [2]University Malaysia of Computer Science and Engineering,
Menara Z10, Jalan Alamanda 2, Putrajaya.
Email: [1]soeheila.mohamad@mimos.my, [2]poh@unimy.edu.my

ABSTRACT

This work presents a combined encryption, signature and multisignature scheme with joint security. The motivation lies in a scenario where a personnel in an organisation is required to sign documents individually, and at times involved in a group signing. Our combined scheme attending to the above concerns is based on the joint security notions proposed by Haber and Pinkas, and more recently by Paterson et al, and uses the same key pair for CS-Lite encryption scheme, Okamoto signature scheme and Ma-Weng-Li-Deng multisignature scheme.

Keywords: joint security, Cramer-Shoup encryption, Okamoto signature, multisignature

1. INTRODUCTION

In practical online applications, cryptographic schemes such as encryption and signature schemes are deployed in a security protocol to provide security services. Current recommended practise is to use distinct key pairs for different schemes because of the insecurity in the combination of textbook RSA encryption and signature schemes. In addition, Klíma-Rosa (2010) presented an attack on a single key-pair RSA OAEP and RSA signature schemes. On the other hand, Degabriele *et al.*(2012) proved that using the same key pair for encryption and signature schemes in EMV is secure. Nevertheless, security of combined scheme must be analysed for the usage of the same or related key pairs for more than one asymmetric scheme.

Consider the scenario whereby a user may have different roles; one in which the user is required to digitally sign a document individually, while another in which the user serves as a member of a group, and all members must collectively sign a document. In such organisation the system requires three schemes, namely encryption, signature and multisignature schemes. For such scenario, we propose the first combined encryption, signature and multisignature scheme to allow for a user to sign a document as an individual and as a team member using only a single key pair. The three schemes are combined to be more efficient in terms of key management (i.e. storage requirements, cost of key certification, time of verification and footprint of cryptographic code) as stated by Haber and Pinkas (2001) and Paterson et al(2011). Moreover, if the personnel's key pairs can be combined to form a group key, groups may be established conveniently whenever required.

Haber and Pinkas (2001) introduced the first security model for joint security. When two schemes S_1 and S_2 which have been proven secure against adversarial models A_1 and A_2 respectively are combined to use the same or related keys, their security analysis must consider the existence of the other scheme. This means the security analysis on S_1 as a component of the combined scheme must consider adversary A_1 with access to oracle of S_2. Similarly for scheme S_2. The combination is secure if for each component the adversary does not gain more advantage in attacking the scheme in the combination than to the scheme singularly.

Based on the joint security model, Coron *et al.*(2002) and Komano-Ohta (2003) proposed universal padding to break the duality of RSA decryption and signing algorithms. Komano-Ohta (2003) also specified the details of the games for encryption and signature scheme combinations.

Beyond breaking RSA duality, Haber and Pinkas (2001) combined DLP-based encryption scheme with public keys and private keys consisting of two elements, with a signature scheme having key elements of the same form. Using the encryption key pair as the combined scheme key pair, the first elements of public key and private key are used as the signature key pair. It was proven that Cramer-Shoup, Naor-Young and OAEP+ encryption schemes can be combined in the above way securely. Similar method is also secure for Poincheval-Stern ElGamal signature scheme to be combined with DLP-based encryption scheme. RSA-based signature schemes, namely Cramer-Shoup, Gennaro-Halevi-Rabin and PSS signature schemes can be combined with encryption schemes having relation with the modulus in the public key. The combination is defined similar to the DLP-based schemes.

Vasco et al (2008) combined DLP-based schemes with identical key generation algorithm. For the combination, the key generation algorithm is adopted without change and the generated key pair is used for both encryption and signature schemes. They combined Boneh-Franklin identity-based encryption (IBE) scheme with Hess ID-based signature scheme, and Pointcheval-Stern modified ElGamal encryption scheme with the signature scheme from applying Fujisaki-Okamoto conversion on ElGamal encryption scheme. Both combinations are proven secure in the random oracle model.

More recently, Paterson et al (2011) constructed combined schemes from one IBE scheme. The encryption component of the combined scheme is constructed using the CHK transform construction Canetti et. el. (2004) and a one-time signature scheme, and the signature component is constructed using Naor transform as used by Boneh-Franklin (2003). One IBE key pair is used for both. The scheme is also extended to include a signcryption scheme. The proposed combined schemes achieve security in the standard model.

2. CARTESIAN PRODUCT SCHEME

Following the notions in Paterson et al (2011), a straightforward combined encryption, signature and multisignature scheme can be obtained by concatenating the keys from the encryption, signature and multisignature schemes. Let **E**=(**E.Setup**, **E.KeyGen**, **E.Encrypt**, **E.Decrypt**) represents the encryption scheme, **S**=(**S.Setup**, **S.KeyGen**, **S.Sign**, **S.Verify**) represents the single-signer signature scheme and let **M** = (**M.Setup**, **M.KeyGen**, **M.Sign**, **M.Verify**) represents the multisignature scheme. A Cartesian product construction **CartCS** can be built as follows:

CartCS.Setup. Execute **E.Setup**, **S.Setup** and **M.Setup** to obtain the public parameters.
CartCS.KeyGen. Execute **E.KeyGen** to obtain (y_e, x_e). Execute **S.KeyGen** to obtain (y_s, x_s). Run **M.KeyGen** to obtain (y_m, x_m). So the public key is (y_e, y_s, y_m) and the private key is (x_e, x_s, x_m).
CartCS.Encrypt. Given a message m and public key y_e, output a ciphertext c=**E.Encrypt**(y_e,m).
CartCS.SingleSign. Given a message m and private key x_s, output a signature s=**S.Sign**(x_s,m).
CartCS.MultiSign. For a message m and private keys $\{x_{mi} \mid 1 \le i \le n\}$ for n the number of users, output a signature s=**M.Sign**($\{x_{mi} \mid 1 \le i \le n\}$,$m$).
CartCS.Decrypt. Given a ciphertext c and private key x_e, output decrypted message **E.Decrypt**(x_e,c).
CartCS.VerifySinglesig. Given a message m, a signature s and a public key y_s, output **S.Verify**(y_s,m,s).
CartCS.VerifyMultisig. Given a message m, a signature s and a set of public keys L, output **M.Verify**(L,m,s).

For this construction, security is achieved if the key pairs are independent of each other. This construction can be used as a benchmark to at least measure the key size for potentially more efficient combined scheme.

3. THE COMBINED SCHEME

Our combined scheme is constructed based on the light version of Cramer-Shoup encryption scheme (CS-Lite) defined by Bellare and Pallacio (2004), signature scheme by Okamoto (1993) and multisignature scheme by Ma, Weng, Li and Deng (MWLD) (2010). These schemes are selected because they have the same key elements. The Okamoto signature and MWLD multisignature shcemes uses the same keys, while the CS-Lite encryption scheme uses the same key as the signature schemes, with one additional element in the private and public key. Hence, to combine these schemes we define the signature keys as part of the encryption key in the same way as Haber and Pinkas (2001).

Setup(1^λ)
1. Generate two primes p and q of length λ such that $q|p-1$.
2. Choose a subgroup $\mathbb{G}$ of $\mathbb{Z}_p^*$ of order q and two generators $g, h \in \mathbb{G}$.
3. Choose two hash functions $H_0: \mathbb{G}^* \times \{0,1\}^* \to \mathbb{Z}_q^*$ and $H_1: \mathbb{Z}_q^* \times \mathbb{G} \to \mathbb{Z}_q^*$.

KeyGen(p,q,g,h)
1. Choose randomly $x_{i1}, x_{i2}, x_{i3} \in \mathbb{Z}_q^*$.
2. Compute $y_{i1} = g^{x_{i1}}h^{x_{i2}} \bmod p$ and $y_{i2} = g^{x_{i3}} \bmod p$.
3. Output to user P_i, private key (x_{i1}, x_{i2}, x_{i3}) and public key (y_{i1}, y_{i2}).

Encrypt ((y_{i1}, y_{i2}),m)
1. Generate a random number $r \in \mathbb{Z}_q^*$.
2. Compute $c_1 = g^r \bmod p$, $c_2 = h^r \bmod p$ and $c_3 = y_{i1}^r \bmod p$.
3. Compute $c_4 = my_{i2}^r \bmod p$.
4. Output ciphertext (c_1,c_2,c_3,c_4).

SingleSign((x_{i1}, x_{i2}, x_{i3}),m)
1. Generate two random numbers $k_1, k_2 \in \mathbb{Z}_q^*$.
2. Compute $r = g^{k_1}h^{k_2} \bmod p$.
3. Compute $c=H_0(r\|y_{i1}\|m)$.
4. Compute $s_1 = k_1 - cx_{i1} \bmod q$ and $s_2 = k_2 - cx_{i2}$.
5. Output signature (r,s_1,s_2).

MultiSign Given a message m, each user P_i prepares the signature share which is finally sent to an assigned group member P_1, to be combined.
1. Generate two random numbers $k_{i1}, k_{i2} \in \mathbb{Z}_q^*$.
2. Compute $R_i = g^{k_{i1}}h^{k_{i2}} \bmod p$.
3. Broadcast (y_{i1},R_i) to all group members.
When user P_i receives (y_{j1},R_j) from j=1,2, . . . , n,
1. Set $L=\{y_{1,1}, y_{2,1}, \ldots, y_{n1}\}$.
2. Compute $R = \prod_{j=1}^{n} R_j \bmod p$.
3. Compute $c=H_0(R\|L\|m)$ and $v_i=H_1(c\|y_i)$.
4. Compute $s_{i1} = k_{i1} - x_{i1}v_i$ and $s_{i2} = k_{i2} - x_{i2}v_i$.
5. Send (s_{i1},s_{i2}) to P_1.
When P_1 receives (s_{j1},s_{j2}) from j=1, 2, . . . , n,
1. Compute $v_j'=H_1(c\|y_{j1})$ for j=1, 2, . . . , n.
2. Compute $r_j' = g^{s_{j1}}h^{s_{j2}}y_{j1}^{v_j'} \bmod p$ for j=1, 2, . . . , n.
3. Compare r_j' to R_j . If there is j with $r_j' \neq R_j$, output ⊥.
4. Compute $s_1 = \sum_{j=1}^{n} s_{j1} \bmod q$ and $s_2 = \sum_{j=1}^{n} s_{j2} \bmod q$.
5. Output signature (c,s_1,s_2).

Decrypt ((x_{i1}, x_{i2}, x_{i3}), (c_1,c_2,c_3,c_4))
1. Compute $t = c_1^{x_{i1}}c_2^{x_{i2}} \bmod p$.
2. If $t \neq c_3$ output ⊥.
3. Compute $m = c_4c_1^{-x_{i3}} \bmod p$.
4. Output m.

VerifySinglesig (m,(c,s_1,s_2),(y_{i1},y_{i2}))
1. Compute $r' = g^{s_1}h^{s_2}y_{i1}^{c} \bmod p$.
2. Compute c'=H$_0$(r'‖y_{i1}‖m).
3. If c' = c, signature is valid. Otherwise signature is not valid.

VerifyMultisig(m,(c,s_1,s_2),L)
1. Compute v_j'=H$_1$(c‖y_{j1}) for j = 1, 2 . . . , n.
2. Compute $R' = g^{s_1}h^{s_2}\prod_{j=1}^{n} y_{j1}^{v_j'} \bmod p$.

3. Compute c'=H$_0$(R‖$'L$‖m).
4. If c'=c, the signature is valid; otherwise it is not valid.

4. SECURITY

We adopt the joint security model defined by Paterson et al(2011). The security goals to be achieved are Indistinguishability (IND) under Chosen Ciphertext Attack (CCA) for the encryption component and, Existential UnForgeability (EUF) under Chosen Message Attack (CMA) for the single-signer signature and the multisignature components.

EUF-CMA of signature component in the presence of decryption, signing and multisignature oracles. We argue that if there is an adversary A which can break EUF-CMA of the signature component in the combination, then there exists an adversary which can find second preimage of hash function H$_0$.

Let A be an adversary against the EUF-CMA of the signature component in the combined scheme. If the adversary A uses only the signing oracle during the query phase and produce a valid forgery, then it implies Okamoto signature scheme does not achieve EUF-CMA. Thus, A must also use the decryption and multisignature oracles to produce the forgery. We show that an adversary B of the hash function may be constructed to find a second preimage.

The challenger gives B the challenge output of H$_1$, H. Firstly, B generates the system parameters primes p and q, and generators g,h∈$\mathbb{Z}_q$ and generates a key pair, private key (x_1^*, x_2^*, x_3^*) and $(y_1^*, y_2^*) = (g^{x_1^*}h^{x_2^*} \bmod p, g^{x_3^*} \bmod p)$. The adversary A is given (y_1^*, y_2^*). The adversary B also acts as the decryption, signing and multisignature share oracles by replying all queries from adversary A. The adversary B is able to reply correctly because B knows the private key of the target user. Hash function H$_1$ is modelled as a random oracle.

When A makes signing queries m to the signature oracle, B uses the generated private key (x_1^*, x_2^*, x_3^*) and H$_0$ to produce signature (r,s_1,s_2) valid under(y_1^*, y_2^*). When A submits the first query (m_0,L_0) to the multisignature share oracle, B generates $R^* = g_{k1}h_{k2}$ where k_1,k_2 are randomly chosen from $\mathbb{Z}_q$. After receiving all other randomness contributions from A, B computes R, and then compute c$_0$=H$_0$(R‖L_0‖m_0). Next B sets H$_1$(c_0‖ y_1^*)=H, and compute the multisignatureshare (s_1^*, s_2^*)=(k_1−Hx_1^*,k_2−Hx_2^*). For all other queries to the multisignature oracles, B set the values for H$_1$ randomly from $\mathbb{Z}_q$\{H}.

Finally, A produces a valid forgery, (m’,(c’,s_1’,s_2’)). If c’=H, s_1’=s_1^* and s$_2$’=s_1^*, B outputs R*‖ y_1^* ‖m’. Otherwise B outputs ⊥. The adversary B succeed everytime A submits such forgery because validity of the signature means c’=H is equal to c′=H$_0$(r‖ y_1^* ‖m’) where

$$r' = g^{s_1^*}h^{s_2^*}y_1^{*H} = g^{k_1-Hx_1^*}h^{k_2-Hx_2^*}y_1^{*H} = g^{k_1}h^{k_2}(g^{x_1^*}h^{x_2^*})^{-H}y_1^{*H} = R^*.$$

EUF-CMA of multisignature component with n-1 traitors in the presence of encryption and signature oracles. Suppose A is an adversary against EUF-CMA of multisignature component in the presence of the signing and mulstisignature share oracles. Here we define an algorithm *B* which can find second preimage of H_1 given such *A*. Let the challenge hash value given to *B* be *H* and model H_0 as a random oracle.

The adversary *B* generates and gives *A* the system parameters, namely primes p and q, and generators g,h∈$\mathbb{Z}_q$ and, private key (x_1^*, x_2^*, x_3^*) and public key $(y_1^*, y_2^*) = (g^{x_1^*}h^{x_2^*} \bmod p, g^{x_3^*} \bmod p)$. It is assumed that *A* has control over all signers except the honest signer with public key (y_1^*, y_2^*), which means private keys for other signers are known to *A*. The adversary *B* also acts as the decryption, signing and multisignature share oracles by replying all queries from adversary *A*. Since *B* knows the private key of the target user, all replies are exactly the same as replies from component oracles. Hash function H_1 is modelled as a random oracle.

During the query phase, *A* makes queries to decryption oracle, signing oracle and the multisignature share oracle. If the adversary *A* queries only the multisignature oracle to forge a multisignature, it implies that the MWLD multisignature scheme does not achieve EUF-CMA. So, *A* must also query the signing oracle.

When *A* queries *m* to multisignature share oracle, *B* generates the randomness contribution R_q and the multisignature share (s_1^*, s_2^*), by querying H_0 and computing H_1. For these queries, outputs of new queries to H_0 are set randomly from $\mathbb{Z}_q \backslash \{H\}$.

When *A* queries m_j to signing oracle, *B* can produce the signature using the private keys and querying H_0. For the first query m_0 to the signing oracle, *B* sets $H_0(r^* \| \ y_1^* \ \| m_0) = H$, where $r^* = g^{k_1}h^{k_2}$ and k_1,k_2 are chosen randomly from $\mathbb{Z}_q$. Then $(s_1^*, s_2^*) = (k_1 - Hx_1^*, k_2 - Hx_2^*)$ is returned to *A*. For all subsequent queries, outputs of new queries to H_0 are set randomly from $\mathbb{Z}_q \backslash \{H\}$.

Finally, when A produce a valid forgery $(m', (c', s_1', s_2'), L')$ with y∗∈ L', $s_1' = s_1^*$ and $s_2' = s_2^*$, B checks whether $H_1(c' \| \ y_1^*)$ equals *H*. If so, *B* outputs $c' \| \ y_1^*$ as preimage of *H* under H_1. Otherwise *B* outputs ⊥.

Since the condition of producing a forgery is that *m*' has never been queried to any of the two signing oracles, the input $c' \| \ y_1^*$ has not been queried to H_1, because $c' = H_0(r \| L \| m)$ will only be queried when *m*' is queried to the multisignature oracle. Also, the preimage is correct because, the multisignature validity implies *r** is calculated correctly,

$$r^* = g^{s_1^*}h^{s_2^*}{y_1^*}^{v^*} = g^{k_1 - Hx_1^*}h^{k_2 - Hx_2^*}{y_1^*}^{v^*} = g^{k_1}h^{k_2}(g^{x_1^*}h^{x_2^*})^{-H}{y_1^*}^{v^*} = r^*{y_1^*}^{-H}{y_1^*}^{v^*}$$

which implies $v^* = H_1(c \| \ y_1^*)$ equals *H*.

IND-CCA of encryption component in the presence of signature and multisignature oracles. Suppose there is an adversary *A* which can break IND-CCA of the encryption component of the combination. We show that there exists adversary *B* which can break the IND-CCA of the CS-Lite.

The challenger setup the system parameters primes *p* and *q*, and generators g,h∈$\mathbb{Z}_q$ and gives the adversary *B* the public key y^* of a target user. To invoke *A*, *B* prepares another set of parameters, the public key (y_1^*, y_2^*) for A by setting $y_1^* = y^*$ and computing $y_2^* = g^{x_3'}$ for some random $x_3' \in \mathbb{Z}_q^*$. The hash functions H_0 and H_1 are modelled by random oracles.

When *A* makes a decryption query on (c_1, c_2, c_3, c_4), *B* computes $m = c_4 c_1^{-x_3'} \bmod p$. Then compute $c_1' = \left({}^{c_4}\!/_{m}\right)^{-x_3'}$ and compares *c*' to *c*. If they are equal, *B* replies *m*, otherwise replies ⊥. This decryption simulator gives the same answer as the actual decryption oracle because the check$\left({c_4}/{m}\right)^{-x_3'} = g^r = c_1$ is always correct.

When A makes a signing query on m, B chooses c,s_1,s_2 randomly and compute $r = g^{s_1}h^{s_2}y_1^{*c}$. Then B sets H$_0$(r‖ y_1^* ‖m)=c. Finally, the signature (c,s_1,s_2) is output to A. This signature is valid for the public key by the design of value r.

For a multisignature share queries, adversary A submits a message m, a list of public keys L and input r_1,r_2,… $r_{n\text{-}1}$ from n-1 users controlled by A. For such query, B creates the share by randomly selecting v,s_1,s_2 and computing $r^* = g^{s_1}h^{s_2}y_1^{*v}$. Then at first stage, B broadcast r*. In the second stage when B receives r_1,r_2,… $r_{n\text{-}1}$ from A, B computes $R = r^*\prod_{j=1}^{n-1} r_j \bmod p$ and query the random oracle to get c=H$_0$(R‖L‖m). Then B sets H$_1$(c‖ y_1^*) to v. The multisignature share output by B is valid by the choice of r*.

Since B can provide correct replies to all decryption, signing and multisignature share queries from A, A will be successful in breaking IND-CCA of the encryption component in the combination. The adversary B uses output of the game from A as his reply to the IND challenge. The adversary B is successful whenever A is successful. Hence, the encryption component in the combination achieves IND-CCA at the same level as the CS-Lite encryption scheme.

5. DISCUSSION

Decryption oracle. In the security arguments for EUF-CMA of the signature and multisignature components of the combination, no decryption queries were made. This is because the security relies on the strength of the double hash techniques while the encryption component does not use any of the hash functions.

Key Length. In this work the key of the combined scheme is shorter than the Cartesian combination, while the ciphertext and signature lengths remain. The Cartesian product would have seven elements of $\mathbb{Z}_q$ as private key ((x_{e1}, x_{e2}, x_{e3}),(x_{s1},x_{s2}),(x_{m1},x_{m2})) and four elements of $\mathbb{Z}_p^*$ as public key (($y_{e1} = g^{x_{e1}}h^{x_{e2}} \bmod p, y_{e2} = g^{x_{e3}} \bmod p), y_{s1} = g^{x_{s1}}h^{x_{s2}} \bmod p, y_{m1} = g^{x_{m1}}h^{x_{m2}} \bmod p$). Our proposed scheme halved the key length by using the same key pair for all component schemes, which consists of three $\mathbb{Z}_p$ elements as the private key, ((x_1, x_2, x_3)) and two element of $\mathbb{Z}_p$ as public key, ($y_1 = g^{x_1}h^{x_2} \bmod p, y_2 = g^{x_3} \bmod p$). The ciphertext length remains at four elements of $\mathbb{Z}_p$ and signature length for both schemes remains at three elements of $\mathbb{Z}_p$.

6. CONCLUSION

We introduced the combination of encryption, signature and multisignature scheme to meet the requirement of enterprise systems where a personnel needs to produce individual signatures and contribute towards a committee signature, besides receiving encrypted documents. We proposed the combination of CS-Lite encryption, the Okamoto signature scheme and MWLD multisignature scheme for such purpose. The resulting key length is half of the Cartesian product construction. We presented the security arguments in the random oracle model.

REFERENCES

Bellare, M. and Palacio, A. 2004. Towards plaintext-aware public-key encryption without random oracles. *Advances in Cryptology – ASIACRYPT 2004*. LNCS **3329**: 42-68.

Coron. J-S., Joyce, M., Naccache, D., Paillier, P., Rosa, T. 2002. Universal padding schemes fro RSA. *Advances in Cryptology – Crypto 2002*. LNCS **2442**: 226-241.

Degabriele, J. P., Lehmann, A., Paterson, K. G., Smart, N. P. and Strefler, M. 2012. On the joint security of encryption and signature in EMV. *Topics in Cryptology – CT-RSA 2012*. LNCS **7178**: 116-135.

Haber, S. and Pinkas, B. 2001. Securely combining public key cryptosystems. *ACM Conference on Computer and Communications Security – CCS 2001*. 215-224.

Klíma, V. and Rosa, T. 2002. Further results and considerations on side channel attacks on RSA. *4th International Workshop on Cryptographic Hardware and Embedded Systems – CHES 2002*. LNCS **2523**: 244-259.

Komano, Y. and Ohta, K. 2003. Efficient universal padding techniques for multiplicative trapdoor one-way permutation. *Advances in Cryptology – Crypto 2003*. LNCS **2729**: 366-382.

Ma, C., Wend, J., Li, Y., Deng, R. 2010. Efficient discrete-log based multi-signature scheme in the plain public key model. *Design, Codes and Cryptography*. **54**: 121-133.

Okamoto, T. 1993. Provably secure and practical identification schemes and corresponding signature schemes. *Advances in Cryptology – Crypto '02*. LNCS **740**: 31-53.

Paterson K. G., Schuldt, J. C. N., Stam, M. and Thompson, S. 2011. On the joint security of encryption and signature, revisited. *Advances in Cryptology –ASIACRYPT 2011*. LNCS **7073**: 161-178.

Vasco, M. I. G., Hess, F. and Steinwart, R. 2008. Combined (identity-based) public key schemes. *Cryptology ePrint Archive*. 2008/466.

A New Arbitrated Signing Scheme Based on BFHP

[1]Amir Hamzah Abd Ghafar and [2]Muhammad Rezal Kamel Ariffin
[1,2]Al Kindi Laboratory, Institute for Mathematical Research, Universiti Putra Malaysia,
43400 Serdang ,
[2]Department of Mathematics, Universiti Putra Malaysia, 43400 Serdang
Email: [1]amirghafar87@gmail.com, [2]rezal@upm.edu.my

ABSTRACT

Digital signing is commonly used in any electronic authentication. It preserves data integrity while maintaining non-repudiation value for signer and verifier involved. As digital world strives towards paperless operation, a derivative of digital signing known as arbitrated digital signing fills the need to use a trusted third party (TTP) to monitor the signing and verification process. In this paper, a new arbitrated digital signing scheme based on Bivariate Function Hard Problem (BFHP) is discussed.

Keywords: Bivariate function hard problem, digital signing scheme.

1. INTRODUCTION

Recent computational technologies can handle millions of electronic communications done every day. But for large organizations and corporations which normally handles same amount of communications per second, it can be a tremendous task even when using more advanced technologies. Communications which either happens internally or externally must be authenticated to make sure its digital integrity. Hence, efficient authentication process must be designed.

Authentication involves two parties which is the sender and recipient of the message. As in the real world, sender needs to sign the message to indicate the he is the message's original sender while recipient needs to verify that the signing is true and indeed is the sender's signature. In the digital world involving electronic communications, this is known as the digital signing process.

The notion of digital signing was conjectured by Whittfield (Diffie and Hellman, 1976) in their renowned paper introducing public key cryptography. By using similar concept of public key cryptography, sender signs the message using his or her own private key while recipient will verify by using sender's public key. Several digital signing schemes have been proposed and implemented. Among the extensively used schemes are the RSA (Rivest, Shamir, and Adleman, 1979) and El-Gammal (Gammal, 1985) digital signing schemes. The latter has become a precursor of Digital Signature Standard (DSS) which is a formal standard endorsed by National Institute of Standards and Technology (NIST) (National Institute of Standards and Technology, 2013).

Both RSA and El-Gammal schemes require a form of modular exponentiation calculation$g^a \pmod N$ for a $\in \mathbb{Z}_N$. The complexity of this calculation if using classical multiplication is $O(n^3)$ if a has same size with N (Galbraith, 2012). Though this complexity is sufficient to be operated by machines, but for millions of rapid communications, it can be clogged. This paper intends to propose a method to reduce the complexity to $O(n^2)$ by using only multiplication and addition operation.

The structure of this paper is as follows. We will introduce a concept of hard mathematical problem called bivariate function hard problem in the next section. Then, we will give an insight definition of arbitrated digital signing scheme before we propose and discuss our own arbitrated digital signing scheme. Then we will end this paper with some future works need to be done and conclude it.

2. PRELIMINARIES

2.1 Bivariate Function Hard Problem (BFHP)

The following proposition gives a proper analytical description of the Bivariate Function Hard Problem (BFHP).

Definition 2.1 We define $\mathbb{Z}^{+}_{(2^{m-1},2^{m}-1)}$ as a set of positive integers in the interval $(2^{m-1}, 2^{m}-1)$. In other words, if $x \in \mathbb{Z}^{+}_{(2^{m-1},2^{m}-1)}$, x is a m-bit positive integer.

Proposition 2.1 (Ariffin *et al.*, 2013)

Let $F(x_1, x_2, \dots, x_n)$ be a multiplicative one-way function that maps $F: \mathbb{Z}^{n} \to \mathbb{Z}^{+}_{(2^{m-1},2^{m}-1)}$. Let F_1 andF_2 be such function (either identical or non-identical) such that$A_1 = F(x_1, x_2, \dots, x_n), A_2 = F(y_1, y_2, \dots, y_n)$ and $\gcd(A_1, A_2) = 1$. Let $u, v \in \mathbb{Z}^{+}_{(2^{n-1},2^{n}-1)}$. Let (A_1, A_2) be public parameters and (u, v) be private parameters.

Let

$$G(u,v) = A_1u + A_2v \tag{1}$$

with the domain of the function G is $\mathbb{Z}^{2}_{(2^{n-1},2^{n}-1)}$ since the pair of positive integers $(u, v) \in \mathbb{Z}^{2}_{(2^{n-1},2^{n}-1)}$ and $\mathbb{Z}^{+}_{(2^{m+n-1},2^{m+n}-1)}$ is the codomain of G since $A_1u + A_2v \in \mathbb{Z}^{+}_{(2^{m+n-1},2^{m+n}-1)}$.

If at minimum $n - m - 1 = k$, where 2^{k} is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, it is infeasible to determine (u, v) over $\mathbb{Z}$ from $G(u, v)$. Furthermore, $(u, v)$is unique for $G(u, v)$ with high probability.

Remark 2.1 We remark that the preferred pair (u, v) in $\mathbb{Z}$, is the *prf*-solution for (1). The preferred pair (u, v) is one of the possible solutions for (1) given by

$$u = u_0 + A_2t \tag{2}$$

and

$$v = v_0 - A_1t \tag{3}$$

for any $t \in \mathbb{Z}$.

Remark 2.2 Before we proceed with the proof, we remark here that the diophantine equation given by $G(u, v)$ is solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found. That is the BFHP is *prf*-solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found.

Proof. We begin by proving that (u, v) is unique for each $G(u, v)$ with high probability. Let $u_1 \neq u_2$ and $v_1 \neq v_2$ such that

$$A_1u_1 + A_2v_1 = A_1u_2 + A_2v_2 \tag{4}$$

We will then have

$$Y = v_2 - v_1 = \frac{A_1(u_1 - u_2)}{A_2}$$

Since $\gcd(A_1, A_2) = 1$ and $A_2 \approx 2^{n}$, then the probability that Y is an integer is 2^{-n}. Then the probability that $v_1 - v_2$ is an integer solution not equal to zero is 2^{-n}. Thus $v_1 = v_2$ with probability $1 - \frac{1}{2^{n}}$.

Next we proceed to prove that to *prf*-solved the Diophantine equation given by (1) is infeasible to be solved. The general solution for $G(u, v)$ is given by (2) and (3) for some integer t.

To find u within the stipulated interval $u \in (2^{n-1}, 2^n - 1)$ we have to find the integer t such that the inequality $2^{n-1} < u < 2^n - 1$ holds. This gives

$$\frac{2^{n-1} - u_0}{A_2} < t < \frac{2^n - 1 - u_0}{A_2}$$

Then, the difference between the upper and the lower bound is

$$\frac{2^{\,n} - 1 - 2^{n-1}}{A_2} = \frac{2^{n-1} - 1}{A_2} \approx \frac{2^{n-2}}{2^m} = 2^{n-m-2}$$

Since $n - m - 1 = k$ where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, we conclude that the difference is very large and finding the correct t is infeasible. This is also the same scenario for v.■

Example 2.1 Let $A_1 = 191$ and $A_2 = 229$. Let $u = 41234$ and $v = 52167$. Then $G = 19821937$. Here we take $m = 16$ and $n = 8$. We now construct the parametric solution for this BFHP. The initial points are $u_0 = 118931622$ and $v_0 = -99109685$. The parametric general solution are: $u = u_0 + A_2 t$ and $v = v_0 - A_1 t$. There are approximately $286 \approx 2^9$ (i.e. $\frac{2^{16}}{229}$) values of t to try (i.e. difference between upper and lower bound), while at minimum the value is $t \approx 2^{16}$. In fact, the correct value is $t = 519172 \approx 2^{19}$.

3. ARBITRATED DIGITAL SIGNING

A digital signature scheme which require unconditionally trusted third party (TTP) to become a part of entity who aid the signing and verification process is called arbitrated digital signing scheme (Menezes, Oorschot, and Vanstone, 1997). The TTP may act in roles of an authority body, internal section of a bank or a commercial-based third party. The deal is both Alice and Bob must not have any doubt of information that being sent by TTP.

In practical, arbitrated digital signing needs a secure symmetric key encryption such as AES to initiate the communication between TTP, Alice and Bob. This causes a drawback because TTP needs an additional public key communication with entities involved to distribute their secret keys. Our scheme wants to tackle this problem.

We show an arbitrated digital signing scheme with symmetric key encryption in Algorithm 1.

Algorithm 1: Textbook Symmetric Arbitrated Signing Scheme

1. Key Generation
    a. Alice and Bob generate their own secret key, k_A and k_B respectively.
    b. Both k_A and k_B are sent to TTP secretly and authentic means to be used as their symmetric key shared with TTP.
2. Signature Generation
    a. Alice calculates message digest of the message, $H = h(m)$.
    b. Alice encrypts H with a symmetric encryption scheme, E using k_A to produce $u = E_{k_A}(H)$.
    c. Alice sends u with her identification string I_A to TTP.
    d. TTP decrypts $E_{k_A}^{-1}(u)$ to get H.
    e. TTP calculates $s = E_{k_T}(H \parallel I_A)$ and sends s to Alice.

```
        f. Alice's signature is $s$.
    3. Verification
        a. Bob calculates $v = E_{k_B}(s)$.
        b. Bob sends $v$ with his identification string $I_B$ to TTP.
        c. TTP decrypts $E_{k_B}^{-1}(v)$ and get $s$.
        d. TTP decrypts $E_{k_T}^{-1}(s)$ and get $H \parallel I_A$.
        e. TTP encrypts $w = E_{k_B}(H \parallel I_A)$ and sends $w$ to Bob.
        f. Bob decrypts $E_{k_B}^{-1}(w)$ to get $H \parallel I_A$.
        g. Bob calculates $H' = h(m)$ from $m$.
        h. Bob accepts Alice's signature if and only if $H' = H$.
```

The symmetric-key algorithm makes the scheme to be fast. However additional exchanges information between the entities and TTP may cause further risk of being intercepted by the attacker. Hence, we propose a scheme with less additional communication together with an asymmetric encryption scheme which is much faster speed than other commercial public key cryptosystem.

4. NEW ARBITRATED DIGITAL SIGNING SCHEMES BASED ON BFHP

Our new arbitrated digital signing schemes use BFHP as its underlying hard mathematical problem. As mentioned in previous sections, the sizes of both the public and private parameters are very crucial to ensure the security of the schemes safe.

We refer to two real-world scenarios as basis for our schemes. Both scenarios emphasize on a need for arbitrated characteristics and make our schemes to become scenario-based schemes.

4.1 First Scenario

An operation center of a bank needs details from its clients to complete a financial transaction. However, the center itself is restricted to only trust details that have been verified by another unit or branch from the same or different bank that has a direct contact with clients. This is a real-world scenario occurs in Real-Time Gross Settlement Systems (Bank for International Settlements, 1997) that being used by banks around the world. We propose the scheme in Algorithm 2 that can handle the communications endured in this scenario.

We need to state that the client is in the role of Alice while the trusted unit or branch that have direct contact with clients acts as TTP and the operation center plays Bob's role. We also state here that 2^r is exponentially large.

Algorithm 2: Arbitrated Signing Scheme I

```
1. Key Generation
    (a) TTP generates two distinct $r$ – bit primes, $p_A$ and $p_B$.
    (b) TTP generates two random numbers, $d_A$ and $d_T$ in the size of $r$ – bit.
    (c) TTP computes $e_A \equiv d_A^{-1} (\bmod p_A)$ and $e_T \equiv d_T^{-1} (\bmod p_B)$.
    (d) Using an asymmetric scheme, TTP encrypts $(p_A, d_A)$ with Alice's public key and
        $(p_B, e_T)$ with Bob's public key and sends the ciphers to their respective owners.
2. Signature Generation
    (a) Alice calculates message digest of the message, $v = h(m)$ in the size of $2r$ – bit.
```

(b) Alice chooses random secret, u_A in the size of $2r$ – bit.
(c) Alice computes her signature, $S_A = u_A p_A + v d_A$
(d) Alice sends S_A to TTP publicly (without any encryption means).

3. Verification (TTP)
   (a) TTP check whether $S_A = \frac{v}{e_A}$. If yes, reject (i.e It means sender does not sign anything). Else, calculates $W_A \equiv S_A e_A (\mathrm{mod}\, p_A)$
   (b) TTP accepts Alice's signature if $W_A \equiv v (\mathrm{mod}\, p_A)$.
   (c) TTP chooses random secret key u_T in the size of $2r$ – bit.
   (d) TTP computes $S_T = u_T p_B + v d_T$
   (e) TTP sends S_T to Bob publicly.
4. Verification (Bob)
   (a) Bob check whether $S_T = \frac{v}{e_T}$. If yes, reject (i.e It means sender does not sign anything). Else, calculates $W_T \equiv S_T e_T (\mathrm{mod}\, p_B)$.
   (b) Bob accepts Alice's signature if $W_T \equiv v (\mathrm{mod}\, p_B)$

Proposition 4.1 (Completeness) If Alice, Bob and the TTP are honest parties in the proposed arbitrated digital signing scheme in Algorithm 2, Bob will accept Alice's signature.

Proof. If Alice is an honest party, TTP can verify the message using e_A. That is

$$S_A e_A \equiv u_A p_A e_A + v d_A e_A \equiv v \ (\mathrm{mod}\, p_A).$$

Consequently, if TTP is honest, it will sign the verified message from Alice using its private parameter, d_T and Bob can verify the message from Alice is indeed has been verified by TTP before by verifying

$$S_T e_T \equiv u_T p_B e_T + v d_T e_T \equiv v \ (\mathrm{mod}\, p_B).$$

■

4.2 Security Analysis

Remark 4.1. From $S_A = u_A p_A + v d_A$ we can rewrite it as $S_A = X + vY$ where $X = u_A p_A$ and $Y = d_A$ are unknown parameters. Let

$$X = X_0 + vt \tag{5}$$

$$Y = X_0 - t \tag{6}$$

be the parametric solution set for S_A. From (5), the interval range for variable t is approximately

$$\frac{2^{3r}}{2^{2r}} \approx 2^r \tag{7}$$

and from (6), it is approximately

$$2^r. \tag{8}$$

Thus, as discussed in section 2, S_A is protected by BFHP.

Proposition 4.2 Given S_A and v, an attacker cannot deduce most significant bits (MSB) of d_A.

Proof. If $\lambda = \frac{u_A p_A}{v}$, we have

$$\frac{S_A}{v} = \lambda + d_A$$

where both $[\lambda]$ and d_A are approximately of the same length. That is $\left\lfloor \frac{S_A}{v} \right\rfloor \neq \lambda$ and $\left\lfloor \frac{S_A}{v} \right\rfloor \neq d_A$. Thus, no information of MSB for d_A will be leaked.

Proposition 4.3 (Forgery Attack) If the attacker is not able to *prf*-solve the BFHP upon S_A and S_T, the proposed arbitrated digital signing scheme in Algorithm 2 can withstand forgery attack.

Proof. S_A is secured by BFHP. That is, the secret key, p_A and ephemeral parameter key, d_A are protected by BFHP. If BFHP can be *prf*-solved, both p_A and d_A can be found. But, based on Proposition 2.1, it is infeasible to find (p_A, d_A). Hence, the proposed scheme can withstand the forgery attack.

The same proof is applied on S_T.

■

Proposition 4.4 (Key only attack) Given e_A, an attacker cannot find the values of secret parameters, (p_A, d_A) if 2^r is exponentially large.

Proof. We can see that $e_A \equiv d_A^{-1} \pmod{p_A}$. Linearly, it can be written as

$$e_A d_A = 1 + k_1 p_A$$

for $k_1 \in \mathbb{Z}$. It is trivial to see that attacker will not know value of e_A if he does not have any knowledge of d_A and M_A. However, an attacker can forge a signing by calculating

$$e_A d'_A = 1 + k_2 p'_A$$

for random $d'_A \neq d_A, p'_A \neq p_A$ and $k_2 \in \mathbb{Z}$. This will not be the case because during verification, TTP will check if $S'_A = u'_A p'_A + v d'_A$ has been forged by calculating $v (\mathrm{mod}\ p_A)$. It is easy to see that in order for $v (\mathrm{mod}\ p'_A) \neq v (\mathrm{mod}\ p_A)$, the probability is $\frac{1}{2^{4r}}$.

The same proof applies on TTP’s signature to Bob.

■

Proposition 4.5 (Known-message attack) Given (S_A, e_A) an attacker that can recompute $v = h(m)$ cannot find the values of secret parameters, (u_A, p_A, d_A).

Proof.

i. The equation $S_A = u_A p_A + v d_A$ consists of 3 variables and protected by BFHP. Thus, (u_A, p_A, d_A) cannot be extracted.

ii. From $e_A \equiv d_A^{-1} \pmod{p_A}$, we have $e_A d_A = 1 + k_1 p_A$. This is 1 equiation with 3 variables and protected by BFHP. Thus, (d_A, p_A) cannot be extracted. As a consequence, variable $\mathrm{u_A}$ from $\mathrm{S_A}$ cannot be extracted. ■

Remark 4.1 Proposition 4.5 ensures that no information about (p_A, u_A, k_1) can be obtained. This means the scheme also can withstand chosen-message attack and adaptive chosen-message attack.

Remark 4.2 Proposition 4.5 also shows that every time S_A and $v = h(m)$ changes, the hard problem of BFHP embedded in the scheme still holds. This increases the efficiency of the scheme in rapid communications because Alice does not have to change her secret keys every time she signs a different message to TTP.

4.3 Performance Analysis

To provide the complexity of the scheme, we will use the major operation in our scheme.

Proposition 4.6 For $p_A, p_b \sim 2^r$ bit in size, overall complexity of proposed arbitrated digital signing scheme is $O(r^2)$.

Proof. Both signatures only use multiplication and addition. Hence at most the complexity is $O_1(2r^2)$. The verification involves modular multiplication which at most also produces $O_2(r^2)$.

So, the overall complexity is $O_1(2r^2) + O_2(r^2) = O(3r^2) = O(r^2)$.

■

Now, we propose the second scheme that is based on scenario below.

4.4 Second Scenario

Two entities want to accept an agreement that they have discussed together. However, due to several issues, both do not trust each other. They only trust another third party. In other words, any communication comes from one of the entity will not be trusted by the other. Hence, to indicate both entities have accepted the agreement, they need to sign it and send to the trusted third party (TTP). TTP then will verify both agreements if only if the signed document of the agreement have the same digital fingerprint.

Our next scheme in Algorithm 3 will fit into the environment. The two entities involve in the agreement will be take Alice and Bob's roles and the third party retain TTP's role.

Algorithm 3: AA_β Arbitrated Signing Scheme II

```
1. Key Generation
    (a) TTP generates two distinct $r$ – bit primes, $p_A$ and $p_B$.
    (e) Using $AA_\beta$ encryption scheme, TTP encrypts $p_A$ with Alice's public key and $p_B$ with Bob's public key and sends the ciphers to their respective owners.
2. Signature Generation (Alice)
    (a) Alice calculates message digest of the message, $v = h(m)$ with size of $2r$ – bit.
    (b) Alice chooses random secret keys, $d_A$ in the size of $r$– bit and $U_A$ in the size of $2r$ – bit.
    (c) Alice computes her public parameter $e_A \equiv d_A^{-1} (\text{mod}\, p_A)$.
    (d) Alice computes her signature $S_A = u_A p_A + v d_A$.
    (e) Alice sends $S_A$ and $e_A$ to TTP publicly (without any encryption means).
3. Signature Generation (Bob)
    (a) Bob calculates message digest of the message, $v = h(m)$.
    (f) Bob chooses random secret keys, $d_B$ in size of $r$– bit and $U_B$ in the size of $2r$ – bit.
    (b) Bob computes $e_B \equiv d_B^{-1} (\text{mod}\, p_B)$
    (c) Bob computes his signature $S_B = u_B p_B + v d_B$.
```

(d) Bob sends S_B and e_B to TTP publicly.

4. Verification
   (a) TTP calculates $v = h(m)$.
   (b) TTP calculates $W_A \equiv S_A e_A (\text{mod } p_A)$
   (c) TTP accepts Alice's signature if $W_A \equiv v (\text{mod } p_A)$.
   (d) TTP calculates $W_B \equiv S_B e_B (\text{mod } p_B)$
   (e) TTP accepts Bob's signature if $W_B \equiv v (\text{mod } p_B)$.

Algorithm 3 has the same security and performance features as scheme in Algorithm 2. The only difference is instead of TTP produce a second signature in Algorithm 2, this second scheme requires Bob to sign the document to be verified by TTP. Other than the flow of signature between entities involved, both our proposed scheme has the same structures.

4.5 Comparative Analysis

We show Table 1 to compare our proposed arbitrated digital signing schemes with textbook arbitrated digital signing scheme.

	Textbook Symmetric Arbitrated Digital Signing Scheme (Menezes, Oorschot, and Vanstone, 1997)	New Arbitrated Digital Signing Scheme I	New Arbitrated Digital Signing Scheme II
Major operation	XOR	Multiplication and addition	Multiplication and addition
Number of Signing	4 (stated as encryption)	2	2
Number of Verification	4 (stated as decryption)	2	2
Number of Transmissions	6	4	4

Table 1: Comparative Analysis between Textbook Symmetric Arbitrated Digital Signing Scheme and Proposed Arbitrated Digital Signing Scheme

Although the textbook symmetric arbitrated has a faster XOR operation as major operation, but our proposed arbitrated digital signing schemes have less number in terms of signings, verifications and transmissions have to be done by the entities involved.

5. CONCLUSION

A digital signing scheme must be flexible according to its applications and necessity in real world scenario. We have established two digital signing schemes which involve participation of trusted third party in its signing and verification process. The schemes are scenario-based and do not operate like conventional digital signing schemes. The schemes also use BFHP as its security backbone and we have provided several possible attacks that can be launched onto the schemes. Up to this point, the schemes are still computationally secure and its performance is $O(r^2)$. The schemes also have advantages compared to textbook arbitrated digital signing scheme in terms of number of signing, verification and data transmission between parties involved.

REFERENCES

Ariffin, M. R., Asbullah, M. A., Abu, N. A., and Mahad, Z. 2013. A New Efficient Asymmetric Cryptosystem Based on the Integer Factorization Problem of $\mathrm{N} = \mathrm{p}^2\mathrm{q}$. *Malaysian Journal of Mathematical Sciences*: 19-37.

Bank for International Settlements. 1997. *Report of Real-Time Gross Settlement Systems.* Retrieved from Bank for International Settlements website: http://www.bis.org/publ/cpss22.pdf

Diffie, W., and Hellman, M. 1976. New directions in cryptography. *IEEE Transactions on Information Theory*: 644-654.

Galbraith, S. D. 2012. *Mathematics of Public Key Cryptography.* New York: Cambridge University Press.

Gammal, T. E. 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. *Proceedings of CRYPTO 84 on Advances in cryptology*: 10-18.

Goldwasser, S., Micali, S., and Rivest, R. L. 1988. A Digital Signature Scheme Secure against Adaptive Chosen-Message Attacks. *SIAM Journal on Computing*: 281-308.

Menezes, A. J., Oorschot, P. C., and Vanstone, S. A. 1997. *Handbook of Applied Cryptography.* Florida: CRC Press LLC.

National Institute of Standards and Technology. 2013. Retrieved from NIST: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

Rivest, R. L., Shamir, A., and Adleman, L. 1979. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*: 120-126.

Comparative Analysis of Three Asymmetric Encryption Schemes Based Upon the Intractability of Square Roots Modulo $N = p^2q$

[1]Muhammad Asyraf Asbullah and [2]Muhammad Rezal Kamel Ariffin
[1,2]Al-Kindi Cryptography Research Labarotary,
Institute for Mathematical Research,
[2]Department of Mathematics, Faculty of Sciences,
Universiti Putra Malaysia
Email: [1]ma_asyraf@upm.edu.my, [2]rezal@upm.edu.my

ABSTRACT

In this paper, we conduct a comparative study for three encryption schemes based upon the difficulties to compute square roots modulo $N = p^2q$, namely HIME(R), Rabin-Takagi and AA_β public key cryptosystem. The running time estimation for each scheme is presented using the single-precision multiplication measurement. We then evaluate the memory cost for system parameters and accumulators during the encryption and decryption process. We observe that there is a trade-off between speed and memory consumption as our result shows AA_β encryption is slower than the other two schemes, but slightly faster when decryption. Due to its large size of plaintext, AA_β consume a greater amount of memory during encryption while use less memory usage for decryption relatively to HIME(R) and Rabin-Takagi.

Keywords: asymmetric encryption, running time, single-precision multiplication

1. INTRODUCTION

A year after the invention of the famous RSA cryptosystem, another cryptosystem namely the Rabin encryption scheme, was proposed. The design is based on the intractability to solve the square root modulo problem of a composite integer. In fact, it is the first asymmetric cryptosystem that can be proven equivalent to problem of factoring $N = pq$. Interestingly, the use of the public exponent 2 for Rabin encryption gives a computational advantage over the RSA (Rabin, 1979). However, the decryption of the Rabin scheme produces four distinct outcomes; consequently additional information is required to find unambiguously the correct plaintext (Elia *et al.*, 2011).

In spite of the situation of 4-to-1 mapping of Rabin's decryption, several attempts were made to solve this problem adequately. For example such as adding some redundancies in the plaintext (Menezes *et al.* (1997)) or incorporating the Jacobi and the Legendre symbol (Kurosawa *et al.*, 2001). Boneh (2001) suggest an elegant strategy by integrating a special kind of padding to the Rabin scheme, namely the Simplified-OAEP (i.e. in short SAEP). SAEP is actually a refinement for Optimal Asymmetric Encryption Padding (OAEP) that was earlier proposed by Bellare and Rogaway (1994) for RSA cryptosystem.

The public key cryptosystem is somehow relatively slow compared to its symmetric counterpart; thus it is not suited for encrypting large bulk of data. Consequently, in modern cryptography principal, the role of the public key encryption scheme is not to encrypt the plaintext directly, instead encrypting the corresponding secret key for symmetric cryptosystem. Nevertheless, some cryptographic protocols such as Secure Electronic Transaction (SET) do not only encrypt the secret symmetric key but also attaches other information together. For example, user's identification information or users account authentication (Hitachi Ltd, 2002). Henceforth, it is significant to design a public key encryption scheme that manifests such purpose. As a result, the Hitachi Ltd took this as a motivation to construct a new encryption scheme known as HIME(R). As claimed in the self evaluation report the plaintext space of HIME(R) scheme is large enough to contain the secret encryption key with the attached information. Note that the HIME(R) is actually a variant of the Rabin encryption scheme using the modulus of type $N = p^2q$.

Alternatively, Takagi (1998) also made a contribution regarding to utilization of moduli $N = p^2q$, prior to the introduction of HIME(R). Although in his paper, Takagi stressed out that such moduli to be used for RSA cryptosystem, but it is possible to apply his method for Rabin scheme also (from now on we will refer it as Rabin-Takagi). Basically, HIME(R) and Rabin-Takagi are quite similar as both perform the Rabin function for encryption, and the Rabin primitive as parts of the decryption. On the contrary, both schemes used very different approach as to solve the square roots modulo $N = p^2q$. In this work, we take in the Rabin-Takagi for comparative purposes.

Recently a new Rabin primitive based encryption scheme known as AA_β was founded which its security also utilized the modulus of type $N = p^2q$. This cryptosystem acquired the quality to secure large data sets. In addition, the decryption method for AA_β is able to produce a unique solution without engaging with any padding or redundancies, while still occupying the Rabin primitive (Ariffin *et al.*, 2013). Observe that the three asymmetric schemes mentioned above have similarities such as; the public key of modulus of type $N = p^2q$, the decryption method involves the Rabin primitive and the Chinese Remainder Theorem (CRT) and despite using different method of the decryption process but the aim is still in common, that is to find all the square roots modulo N.

The rest of the paper is structured as follows. Section 2 provides brief preliminaries on the Rabin primitive, CRT and Garner's algorithm. Section 2 also introduces definitions to the single-precision multiplications measurement, the system parameters and the accumulators. A brief description about the HIME(R), the Rabin-Takagi and the AA_β cryptosystem are presented in Section 3. The running time and the memory cost for each scheme are evaluated in this section. In Section 4, we further our discussion with comparative study of all schemes using several criterions which will be mentioned later. Finally, conclusion is made in Section 5.

2. PRELIMINARIES

In this section a brief overview of the Rabin primitive and the CRT are provided. Single-precision measurement is placed in the following sub sections. Finally we define the system parameters and accumulators.

2.1 Rabin Primitive

In this paper, the Rabin primitive is referring to the operation to find all the square roots modulo a composite integer, coupled with the recombination process of the simultaneous congruence using the Chinese Remainder Theorem. The best example is the decryption part on the Rabin cryptosystem itself.

2.2 Chinese Remainder Theorem and Garner's Algorithm

The CRT algorithm is a well-studied algorithm and can be found in many literatures, for instance Menezes *et al.* (1997) and Galbraith (2012). However, there exists a faster and more efficient method called the Garner's algorithm to solve the CRT. Furthermore, Garner's algorithm performs better when operating with large integers. One may refer to Vuillaume (2003) for the detail analysis on the computational advantages of Garner's algorithm.

2.3 Single-precision Multiplication

The running time for an algorithm is possibly measured in numerous ways such as the number of steps, the machine instructions or the clock cycles (Menezes *et al.*, 1997). Running time is important since it can show the performance for an algorithm, and could be used as a method of comparison between algorithms.

We choose the 'single-precision multiplication' measurement in order to determine the running time, as was used by CRYPTREC (2002) to conduct their analysis on several algorithms. For the record, CRYTPREC is a cryptographic evaluation community establish by the Japanese Government to analyze and suggest cryptographic techniques for government and industrial use.

Definition A single-precision multiplication (*spm*) is referred as the multiplication of the two base b digits. In cryptography, $b = 2$ is favored which is binary representation of any digits. We suppose the running time for an addition and a subtraction is operations that can perform very quickly compared to a multiplication or a division. Furthermore, division is the most complex and expensive amongst basic arithmetic operations. We will use the following measurement for our estimation.

- (*Addition/subtraction*). Suppose we add or subtract for two α-bit integers then we have α *spm*.
- (*Multiplication*). Suppose we multiply α-bit and β-bit integers, thus this operation require $(\alpha + 1)(\beta + 1)$ *spm*. Hence, for multiplication of two α-bit integers we have $\alpha^2 + 2\alpha + 1$ *spm*.
- (*Division*). Suppose we divide α-bit by β-bit integers, thus this operation requires $(\alpha - \beta)(\beta + 3)$ *spm*.
- (*Modular multiplication*). Suppose a, b and c are integers of α-bit integers. Let $ab \ (mod\ c)$ is a modular multiplication, then this operation requires $2\alpha^2 + 5\alpha + 1$ *spm*.
- (*Modular reduction*). The direct approach to carry out a modular reduction is to obtain the remainder after the division of α-bit by β-bit integers; therefore it is similar to the division running time.
- (*Modular Inversion*). A modular inversion is as 30 times as fast as a modular multiplication, as suggested by Okeya and Sakurai (2001).
- (*Modular Exponentiation*). Suppose a, b and c are integers of α-bit integers. Let a^b (mod c) is modular exponentiation where exponent b are the ε-bit and c are integers of α-bit integers. Then this operation requires $(3\varepsilon - 2)\ (\alpha^2 + \alpha)$ *spm*.

2.4 System parameter and accumulator

Definition A system parameter is a constant or a variable; normally was pre-computed and possibly was fixed prior to any computational activity. The system parameter will be stored permanently in memory space. For instance, the secret keys embedded in the hardware (i.e. hardwired with IC).

Definition Accumulator is a constant or a variable value of any current computational activity, and its resultant value is temporarily being stored in the memory. Once all of the required computations are finished, then the parameters will be deleted. In other word, the memory space for accumulator are dynamically being stored and deleted.

3. SCHEMES SPECIFICATION AND ANALYSIS

In the following, we will describe the key generation, encryption and decryption procedure for HIME(R), Rabin-Takagi and AA_β cryptosystem, respectively. In this work, the specification on the hash function or any padding mechanism for each schemes are omitted but detailed descriptions can be found in their respective original document. For each scheme, we estimate the running time during encryption and decryption. In addition, we also estimate the cost of memory consumption for the system parameters and the accumulators.

1.1 HIME(R) Cryptosystem (Hitachi Ltd, 2002)

HIME(R) Key Generation

```
INPUT: The size $k$ of the prime numbers.
OUTPUT: A public key tuple $(k, N)$ and a private key tuple $(p,\ q,\ \alpha, \beta, j)$.
1. Generate random and distinct $k$-bit primes $p,\ q$ such that $p,\ q \equiv 3\ (mod\ 4)$ where $2^k < p,\ q < 2^{k+1}$.
2. Compute $N =\ p^2q$ and $j = p^{-1}$ (mod $q$)
3. Set $\alpha = \frac{p+1}{4}$ and $\beta = \frac{q+1}{4}$.
4. Return the public key tuple $(k, N)$ and a private key tuple $(p,\ q,\ \alpha, \beta, j)$.
```

HIME(R) Encryption

```
INPUT: The plaintext $m, N$
OUTPUT: A ciphertext $C$.
1. Choose integer $m \in \mathbb{Z}_N$.
2. Compute $C =\ m^2$ mod $N$.
3. Return the ciphertext $C$.
```

3.1.1 Running Time Estimation for HIME(R)-encryption

First of all, a plaintext m of a positive integer less than N is chosen. Note that the maximal size of the plaintext m is $3k$-bits. In step 2 we compute a multiplication of two $3k$-bit integers and a modular reduction of $9k$-bit and $3k$-bit, thus the cost are $9k^2 + 6k + 1$ *spm* and $6k^2 + 18k$ *spm*, respectively. Hence, the overall running time for HIME(R)-encryption is $15k^2 + 24k + 1$ *spm.*

3.1.2 Memory Cost for HIME(R)-encryption

Register names	Bits	Number of registers
m, N	$3k$	2
C	$3k$	1
	Subtotal	$6k + 3k = 9k$

Table 1. System parameters memory for HIME(R)-encryption

HIME(R) Decryption

```
INPUT: A ciphertext $C, p, q, pq, \alpha, \beta, j$
OUTPUT: The plaintext $m$
1. Compute $C_p \leftarrow C(\text{mod } p)$
2. Compute $C_q \leftarrow C(\text{mod } q)$
3. Compute $a_1 \leftarrow C_p^{\alpha}$ (mod $p$).If $a_1^2(\text{mod } p) = C_p$, then compute $a_2 = p - a_1$, else reject.
4. Compute $b_1 \leftarrow C_q^{\beta}$ (mod $q$). If $b_1^2(\text{mod } p) = C_p$, then compute $b_2 = p - b_1$, else reject.
5. Compute $h_1 \leftarrow (b_1 - a_1)j$ (mod $q$)
6. Compute $h_2 \leftarrow (b_1 - a_2)j$ (mod $q$)
7. Compute $h_3 \leftarrow (b_2 - a_1)j$ (mod $q$)
8. Compute $h_4 \leftarrow (b_2 - a_2)j$ (mod $q$)
9. Compute $H_1 \leftarrow a_1 + h_1p$
10. Compute $H_2 \leftarrow a_2 + h_2p$
11. Compute $H_3 \leftarrow a_1 + h_3p$
12. Compute $H_4 \leftarrow a_2 + h_4p$
```

```
13. For $i$ from 1 to 4 compute $X_i \leftarrow \frac{[C-H_i^2 \pmod N)]}{pq}$
14. For $i$ from 1 to 2 compute $y_i \leftarrow (2a_i)^{-1} \pmod p$
15. Compute $Y_1 \leftarrow X_1 y_1 \pmod p$
16. Compute $Y_2 \leftarrow X_2 y_2 \pmod p$
17. Compute $Y_3 \leftarrow X_3 y_1 \pmod p$
18. Compute $Y_4 \leftarrow X_4 y_2 \pmod p$
19. For $i$ from 1 to 4 compute $m_i \leftarrow H_i + Y_i pq$
20. Find the proper $m$ from $m_i$ for $i = 1,2,3,4$
21. Return $(m)$
```

3.1.3 Running Time Estimation for HIME(R)-decryption

Let the private key be the tuple $(p, q, pq, \alpha, \beta, j)$ with the size of (p, q, α, β, j) are k-bit and pq is $2k$-bits. Consider the HIME(R)-decryption algorithm, hence we show the details of its running time as follows.

In step 1 and step 2, we perform for each step a modular reduction operation of $C \pmod p$ and $C \pmod q$ of $3k$-bit by k-bit. Hence the running time for both is $4k^2 + 12k$ *spm*. In step 3, we compute an exponentiation o f k-bit exponent with k-bit modulus. This step cost $3k^3 + k^2 - 2k$ *spm*. We then further with a multiplication of two k-bit integers, a modular reduction of $2k$-bit and k-bit integer and an addition of k-bit which is cost $2k^2 + 6k + 1$ *spm*. Thus the running time of whole step 3 computation cost $3k^3 + 3k^2 + 4k + 1$ *spm*. Step 4 also does the same operation as step 3. Thus the running time also exhibit in similar manner.

From step 5 – 8, each step conduct a subtraction of k-bit and a modular multiplication of two k-bit integers. To each one, $2k^2 + 6k + 1$ *spm* is needed and as a result, the overall running time for step 7 – 10 is $8k^2 + 24k + 4$ *spm*. An addition of $2k$-bit and a multiplication of two k-bit integers are computed for the next steps (i.e. step 9 – 12). Each step needs $k^2 + 4k + 1$ *spm* and as a result, the whole running time for every step cost $4k^2 + 16k + 4$ *spm*.

Step 13 will carry on four times of calculations for each calculation consist a multiplication of two $2k$-bit integers, a subtraction $4k$-bit integer, a modular reduction of $4k$-bit by $3k$-bit integer and a division of $3k$-bit and $2k$-bit integer. For each calculation, a multiplication of $2k$-bit and $2k$-bit will cost $4k^2 + 4k + 1$ *spm*, the cost of subtraction needs $4k$ *spm*. Then proceed with a modular reduction which requires $3k^2 + 3k$ *spm*. Lastly we perform a division operation of $3k$-bit by $2k$-bit, thus it need$2k^2 + 3k$ *spm*. Since this step is repeated four times, hence the running time for this step is $36k^2 + 56k + 4$ *spm*.

A modular inversion requires 30 times as fast as the running time of modular multiplication of two k-bit, which actually cost $30(2k^2 + 5k + 1) = 60k^2 + 150k + 30$ *spm*. In step 14, we compute two modular inversions; hence the running time for these steps is $120k^2 + 300k + 60$ *spm*. A modular multiplication of two k-bit integers is performed for the next steps (i.e. step 15 – 18). Each step needs $2k^2 + 5k + 1$ *spm* thus the running time for all four steps is needs totally $8k^2 + 20k + 4$ *spm*.

An addition of $3k$-bit and a multiplication of of k-bit and $2k$-bit integers are executed in the step 19. Each step needs $2k^2 + 6k + 1$ *spm* and since this step is repeated four times, as a result the total computation is $8k^2 + 24k + 4$ *spm*.

To conclude, the decryption algorithm for HIME(R) requires $6k^3 + 194k^2 + 460k + 82$ *spm*.

3.1.4 Memory Cost for HIME(R)-decryption

Register names	Bits	Number of registers
C	$3k$	1
p, q, α, β, j	k	5
pq	$2k$	1
	Subtotal	$3k + 5k + 2k = 10k$

Table 2. System parameters memory for HIME(R)-decryption

Register names	Bits	Number of registers
C_p, C_q	k	2
a_1, a_2, b_1, b_2	k	4
h_1, h_2, h_3, h_4	k	4
H_1, H_2, H_3, H_4	$2k$	4
X_1, X_2, X_3, X_4	k	4
y_1, y_2	k	2
Y_1, Y_2, Y_3, Y_4	k	4
m_1, m_2, m_3, m_4	$3k$	4
	Subtotal	$40k$

Table 3. Accumulators' memory for HIME(R)-decryption

3.2 Rabin-Takagi Cryptosystem (Takagi, 1998)

Rabin-Takagi Key Generation

INPUT: The size k of the prime numbers.

OUTPUT: A public key tuple (k, N) and a private key tuple $(p,\ q,\ l, p^2)$.

1. Generate random and distinct k-bit primes $p,\ q$ such that $p,\ q \equiv 3 \pmod 4$ where $2^k < p,\ q < 2^{k+1}$.
2. Compute $N = p^2q$ and $l = (p^2)^{-1} \pmod q$
3. Return the public key tuple (k, N) and a private key tuple $(p,\ q,\ l, p^2)$.

Rabin-Takagi Encryption

INPUT: The plaintext m, N.

OUTPUT: A ciphertext C.

1. Choose integer $m \in \mathbb{Z}_N$.
2. Compute $C = m^2 \bmod N$.
3. Return the ciphertext C.

3.2.1 Running Time Estimation for Rabin-Takagi encryption

Observe that the encryption process of Rabin-Takagi is similar to encryption of HIME(R) cryptosystem. Hence, the overall running time for Rabin-Takagi-encryption is $15k^2 + 24k + 1$ *spm.*

3.2.2 Memory Cost for Rabin-Takagi encryption

Register names	Bits	Number of registers
m, N	$3k$	2
C	$3k$	1
	Subtotal	$6k + 3k = 9k$

Table 4. System parameters memory for Rabin-Takagi encryption

Rabin-Takagi Decryption

```
INPUT: A ciphertext $C, p, q, l, p^2$.
OUTPUT: The plaintext $m$
1. Compute $M_p \leftarrow C^{\frac{p+1}{4}}$ (mod $p$)
2. Compute $M_q \leftarrow C^{\frac{q+1}{4}}$(mod $q$)
3. Compute $F_1 \leftarrow M_p^2$ (mod $p^2$)
4. Compute $F_2 \leftarrow (p - M_p)^2$ (mod $p^2$)
5. For $i$ from 1 to 2 compute $E_i \leftarrow C - F_i$ (mod $p^2$)
6. For $i$ from 1 to 2 compute $B_i \leftarrow \frac{E_i}{p}$
7. For $i$ from 1 to 2 compute $k_i \leftarrow (2F_i)^{-1}$ (mod $p$)
8. Compute $K_1 \leftarrow M_p B_1 k_1$ (mod $p$)
9. Compute $K_2 \leftarrow (p - M_p) B_2 k_2$ (mod $p$)
10. For $i$ from 1 to 2 compute $A_i \leftarrow M_p + K_i p$
11. Compute $V_1 \leftarrow (M_q - A_1) l$ (mod $q$)
12. Compute $V_2 \leftarrow (M_q - A_2) l$ (mod $q$)
13. Compute $V_3 \leftarrow (q - M_q - A_1) l$ (mod $q$)
14. Compute $V_4 \leftarrow (q - M_q - A_2) l$ (mod $q$)
15. Compute $m_1 \leftarrow A_1 + V_1 p^2$
16. Compute $m_2 \leftarrow A_2 + V_2 p^2$
17. Compute $m_3 \leftarrow A_1 + V_3 p^2$
18. Compute $m_4 \leftarrow A_2 + V_4 p^2$
19. Find the proper $m$ from $m_i$ for $i = 1,2,3,4$
20. Return $(m)$
```

3.2.3 Running Time Estimation for Rabin-Takagi decryption

Let the private key be the tuple (p, q, p^2, l) with the size of (p, q, l) are k-bit and p^2 is $2k$-bits. Consider the Rabin-Takagi-decryption algorithm, hence we show the details of its running time as follows.

In step 1, we perform an exponentiation of k-bit exponent with k-bit modulus. This step cost $3k^3 + k^2 - 2k$ *spm*. This process is repeated for step 2, thus just now it double the cost to $6k^3 + 2k^2 - 4k$ *spm*. A multiplication of two k-bit integers and a modular reduction of $2k$-bit and $2k$-bit integer are executed during step 3. Thus the running time of step 3 is $k^2 + 3k + 4$ *spm*. Step 4 does the same operation as step 3 with an extra operation (i.e. subtraction of k-bit) and its running time is $k^2 + 4k + 4$.

From step 5, we conduct two calculations of a subtraction of $3k$-bit and a modular reduction of $3k$-bit by $2k$-bit integer. Each calculation cost $2k^2 + 3k$ *spm*. Two division of $2k$-bit by k-bit integer is performed in step 6, thus we need $2k^2 + 6k$ *spm*. A modular inversion requires 30 times as fast as the running time of modular multiplication of two k-bit. However, in step 7, we compute two modular inversions of two $2k$-bit; hence the running time for these steps is $120k^2 + 360k + 240$ *spm*.

A multiplication of two k-bit integers, a multiplication of $2k$-bit and k-bit integer and a modular reduction of $3k$-bit by k-bit integer are executed during step 8. Thus the running time of this step is $5k^2 + 11k + 2$ *spm*. Step 9 does the same operation as step 8 with an extra operation (i.e. subtraction of k-bit), therefore its running time is $5k^2 + 12k + 2$. An addition of $2k$-bit and a multiplication of two k-

bit integers are computed, twice for the next steps (i.e. step 10). Hence, the whole running time for step 10 cost $2k^2 + 8k + 2$ *spm*.

Step 11 – 14 will carry on four times of calculations for each step consist a subtraction of k-bit integer and a modular multiplication of two $2k$-bit integers. For each step, the cost of subtraction needs k *spm* and the modular multiplication will cost $2k^2 + 5k + 1$ *spm*. Since this step is repeated four times, hence the running time for this step is $8k^2 + 24k + 4$ *spm*. Continuing on Step 15 – 18, for each step consist an addition of $3k$-bit integer which cost $3k$ *spm* and a multiplication of $2k$-bit and k-bit integer that will cost $2k^2 + 3k + 1$ *spm*. Since this step is repeated four times, hence the running time for this step is $8k^2 + 24k + 4$ *spm*.

To conclude, the decryption algorithm for Rabin-Takagi requires $6k^3 + 158k^2 + 460k + 262$ *spm*.

3.2.4 Memory Cost for Rabin-Takagi decryption

Register names	Bits	Number of registers
C	$3k$	1
p^2	$2k$	1
p, q, l	k	3
	Subtotal	$3k + 2k + 3k = 8k$

Table 5. System parameters memory for Rabin-Takagi decryption

Register names	Bits	Number of registers
M_p, M_q	k	2
F_1, F_2, E_1, E_4	$2k$	4
A_1, A_2, B_1, B_2	k	4
k_1, k_2, K_1, K_2	k	4
v_1, v_2, v_3, v_4	k	4
m_1, m_2, m_3, m_4	$3k$	4
	Subtotal	$34k$

Table 6. Accumulators' memory for Rabin-Takagi decryption

3.3 AA_β Cryptosystem (Ariffin *et al*.., 2013)

***AA_β* Key Generation**

INPUT: The size k of the prime numbers.

OUTPUT: A public key tuple (A_1, A_2) and a private key tuple $(p,\ q,\ d)$.

```
1. Generate random and distinct $k$-bit strong primes $p,\ q$ such that $p,\ q \equiv 3 \pmod 4$ where $2^k < p,\ q < 2^{k+1}$.
2. Choose random $d$ such that $\gcd(d, pq) = 1$.
3. Compute integer $e_0$ such that $e_0 d \equiv 1 (\mathrm{mod}\ pq)$. Add multiples of $pq$ for some integer $i$ until $2^{3k+4} < e = e_0 + ipq < 2^{3k+6}$ (if necessary).
4. Compute $j \leftarrow p^{-1} \pmod q$
5. Set $A_1 = p^2q$ and set $A_2 = e$.
6. Return the public key tuple $(A_1, A_2)$ and a private key tuple $(p, q, pq, d, j)$.
```

AA_β Encryption

```
INPUT: The plaintext tuple $(u, v, A_1, A_2)$.
OUTPUT: A ciphertext $C$.
  1. Choose integer $u \in (2^{4k}, 2^{4k+1})$.
  2. Choose integer $v \in (2^{2k-2}, 2^{2k-1})$.
  3. Compute $C = uA_1 + v^2 A_2$.
  4. Return the ciphertext $C$.
```

3.3.1 Running Time Estimation for AA_β-encryption

In the step 1 and step 2, we choose two integers; u of the size $4k$-bit and v with the size $2k$-bit. In step 3 we compute a multiplication of two $2k$-bit and two multiplication of $4k$-bit and $3k$-bit, thus the cost are $4k^2 + 4k + 1$ *spm* and $24k^2 + 14k + 2$ *spm*, respectively. Finally, we add two $7k$-bit integers that cost $7k$ *spm*. Hence, the overall running time for AA_β-encryption is $28k^2 + 18k + 3$ *spm*.

3.3.2 Memory Cost for AA_β-encryption

Register names	Bits	Number of registers
u	$4k$	1
v	$2k$	1
C	$7k$	1
A_1, A_2	$3k$	2
	Subtotal	$19k$

Table 7. System parameters memory for AA_β-encryption

AA_β Decryption (Asbullah and Ariffin, 2014)

```
INPUT: A ciphertext $C$ and private key tuple $(p, q, pq, j, d)$
OUTPUT: The plaintext tuple $(u, v)$
  1. Compute $W \leftarrow Cd (\text{mod } pq)$
  2. Compute $x_p \leftarrow W^{\frac{p+1}{4}} (\text{mod } p)$
  3. Compute $x_q \leftarrow W^{\frac{q+1}{4}} (\text{mod } q)$
  4. Compute $h_1 \leftarrow (x_q - x_p)j \ (\text{mod } q)$
  5. Compute $h_2 \leftarrow (-x_q - x_p)j \ (\text{mod } q)$
  6. Compute $v_1 \leftarrow x_p + h_1 p$
  7. Compute $v_2 \leftarrow x_p + h_2 p$
  8. Compute $v_3 \leftarrow pq - v_2$
  9. Compute $v_4 \leftarrow pq - v_1$
  10. For $i = 1,2,3,4$ compute $u_i \leftarrow \frac{C - {v_i}^2 e}{A_1}$ if $v_i < \frac{pq}{2}$, else reject
  11. Sort the pair $(u_i, v_i)$ for integer $u_i$
  12. Return $(u, v)$
```

We remark that the decryption algorithm for the AA_β scheme used here is taken from the work of Asbullah and Ariffin (2014), since it is more efficient than its earlier version of Ariffin *et al.*, 2013.

3.3.3 Running Time Estimation for AA_β-decryption

Let the private key be the tuple (p, q, pq, j, d) with the size of (p, q, j, d) are k-bit and pq is $2k$-bits. Consider the AA_β-decryption algorithm, hence we show the details of its running time as follows.

In step 1, we perform a multiplication of $7k$-bit and $2k$-bit then continue with a modular reduction on the resultant integer by pq which is $2k$-bit. Hence the running time for this operation is $14k^2 + 9k + 1$ *spm* and $14k^2 + 21k$ *spm*, respectively. Overall estimation for step 1 is $28k^2 + 30k + 1$ *spm*. In step 2, we compute an exponentiation of k-bit exponent with k-bit modulus. This step cost $3k^3 + k^2 - 2k$ *spm*. Step 3 also execute the same operation, thus for both step, the running time cost is $6k^3 + 2k^2 - 4k$ *spm*. A subtraction of k-bit and a modular multiplication of two k-bit integers are executed during step 4 and step 5. Each step needs $2k^2 + 6k + 1$ *spm* and as a result, the overall computation for step 4 and step 5 is $4k^2 + 12k + 2$ *spm*.

An addition of $2k$-bit and a multiplication of two k-bit integers is computed for the next steps (i.e. step 6 – 7). Each step needs $k^2 + 4k + 1$ *spm* and as a result, the total computation for both steps is $2k^2 + 8k + 2$ *spm*. Then we carry out two subtractions in step 8 – 9, which cost $4k$ *spm*.

Amongst the four integers v_1, v_2, v_3 and v_4, only two values are less than $\frac{pq}{2}$, thus there exist at least two integers of v_i that can be discarded during decryption, where $i = 1,2,3,4$. Since we already discarded two integers v_i greater than $\frac{pq}{2}$, thus we may assume that for the maximum size of v_1 is $2k$-bit. Thus, step 10 will carry on two calculations, and each calculation consist a subtraction $7k$-bit integer, a multiplication of two $2k$-bit integers, a multiplication of $4k$-bit and $3k$-bit integer and a division of $7k$-bit and $3k$-bit integer. For each calculation, the cost of subtraction needs $7k$ *spm*; a multiplication of $2k$-bit and $2k$-bit will cost $4k^2 + 4k + 1$ *spm*. Then proceed with a multiplication of $4k$-bit and $3k$-bit integer that cost $12k^2 + 7k + 1$ *spm*. Finally, we continue with a division of $7k$-bit by $3k$-bit, thus we need $12k^2 + 12k$ *spm*. Since this procedure repeated two times, hence the running time for step 10 is $56k^2 + 60k + 2$ *spm*.

To conclude, step 1 – 10 of AA_β-decryption algorithm requires $6k^3 + 92k^2 + 110k + 7$ *spm*.

3.3.4 Memory Cost for AA_β-decryption

Register names	Bits	Number of registers
p, q, j	k	3
pq, d	$2k$	2
A_1, A_2	$3k$	2
C	$7k$	1
	Subtotal	$20k$

Table 8. System parameters memory for AA_β-decryption

Register names	Bits	Number of registers
x_p, x_q, h_1, h_2	k	4
W, v_1, v_2, v_3, v_4	$2k$	5
u_i	$4k$	1
	Subtotal	$18k$

Table 9. Accumulators' memory for AA_β-decryption

4. COMPARATIVE STUDY

All of the three cryptosystems share some common features such as the modulus of type $N = p^2q$ and the security that based on the hardness of finding the square roots modulo N. The encryption for both HIME(R) and Rabin-Takagi uses the Rabin function, while for AA_β-encryption, it only involves the normal addition and multiplications without modular reduction. All mentioned schemes in this paper perform the Rabin primitive (i.e. to find all the square roots modulo N) as parts of their decryption process, however the fundamental difference lies on the mathematical method.

This section gives the comparison between the three schemes that was mentioned earlier. We use several criteria to provide a detail analysis between all of the three schemes. The comparison criterias are described in the following tables.

	Public parameters	Memory (bits)	Private parameters	Memory (bits)
HIME(R)	$N = p^2q$	$3k$	$p, q, pq, \alpha, \beta, j$	$7k$
Rabin-Takagi	$N = p^2q$	$3k$	p, q, p^2, l	$5k$
AA_β	$A_1 = p^2q, A_2 = e$	$6k$	p, q, pq, d, j	$7k$

Table 10: Public and private parameters with its memory consumption.

	Ratio of plaintext and ciphertext size (bits)
HIME(R)	$3k$:$3k$
Rabin-Takagi	$3k$:$3k$
AA_β	$6k$:$7k$

Table 11: Ratio of plaintext and ciphertext

Table 10 displays the memory consumption for the public and the private parameters. Notice that all schemes require more memory to store the private parameters against its public parameters. The AA_β public keys need twice as much as the amount of memory of the other two schemes. The ratio between the length of plaintext and ciphertext are the same for HIME(R) and Rabin-Takagi as depicted in Table 11, whilst it is slightly different for AA_β.

	Running time (*spm*)	For $k = 512$	For $k = 1024$
HIME(R)	$15k^2 + 24k + 1$	3944449	15753217
Rabin-Takagi	$15k^2 + 24k + 1$	3944449	15753217
AA_β	$28k^2 + 18k + 3$	7349251	29378563

Table 12: The running time estimation of encryption stage

Table 12 presents the running time for each scheme in term of single-precision multiplication. We also examine the concrete running time using k equal to 512 and 1024 bits. Obviously, we see that the encryption running time outline by Rabin-Takagi is similar to HIME(R)-encryption, whereas it is almost doubled the cost for AA_β.

	Running time (spm)	$k = 512$	$k = 1024$
HIME(R)	$6k^3 + 196k^2 + 454k + 82$	856919122	6648436818
Rabin-Takagi	$6k^3 + 158k^2 + 460k + 262$	846960902	6608597254
AA_β	$6k^3 + 92k^2 + 110k + 7$	829479943	6539032583

Table 13: The running time estimation of decryption stage

Table 13 shows that the amount of the single precision multiplications taken to complete the decryption process. The result shows that the decryption speed in decreasing order as AA_β is faster followed with Rabin-Takagi and then HIME(R). This is quite a surprising result since the ciphertext of AA_β is carried double the size as both HIME(R) and Rabin-Takagi, yet the decryption is still comparably fast enough. Interestingly, this result suggests that the AA_β can send twice the size of the plaintext as large as HIME(R) or Rabin-Takagi; nevertheless its performance is equivalence to the computational cost of HIME(R) while decrypting. Instead, if we use HIME(R) then we need to encrypt and decrypt two times for the same amount of the plaintext sents by AA_β. The same argument is applied to Rabin-Takagi. This could be a huge saving in the matter of computation or even as far as storage is concerned.

	System parameters	Accumulators	Total memory
HIME(R)	m, N, C	None	$9k$
Rabin-Takagi	m, N, C	None	$9k$
AA_β	u, v, A_1, A_2, C	None	$19k$

Table 14: The memory consumption during encryption stage

Table 14 summarizes the memory usage during the encryption stage. The systems parameters of HIME(R) and Rabin-Takagi are similar, reported to use the same amount of the memory. However, the total memory occupied by AA_β during its encryption process is double as much as occupied by HIME(R). This event is logical since the objective of AA_β is to transmit large data sets. Furthermore, its encryption algorithm does not perform any modular reduction; therefore the size of its corresponding ciphertext should be large, as shown in the previous section. Note that there is no memory consumption for accumulators during encryption process.

	System parameters	Accumulators	Total memory
HIME(R)	$10k$	$40k$	$50k$
Rabin-Takagi	$8k$	$34k$	$42k$
AA_β	$20k$	$18k$	$38k$

Table 15: The memory consumption during decryption stage

Table 15 reports the cost of memory during the decryption process. In comparison to HIME(R) and Rabin-Takagi, indeed AA_β occupies more memory for the system parameters, due to the large size of the public keys and its ciphertext. In spite of that, it turns out that AA_β uses the least memory for the accumulators during decryption computational procedure. Importantly, the total memory consumption demonstrated by AA_β suggests that this cryptosystem is not only faster during decryption but also uses less memory compared to HIME(R) and Rabin-Takagi.

5. CONCLUSION

Comparative results indicate that HIME(R) and Rabin-Takagi encrypt twice as fast as AA_β. The result also show that AA_β could be used to transmit large data but comes with reasonable trade off; AA_β-encryption is relatively slow and needs additional memory space due the large size of its plaintext. In this paper, if we consider selecting a scheme with faster encryption, then Rabin-Takagi is the best choice. On the contrary, the result suggests that it is rather economical to choose the AA_β whenever we want to transmit a large bulk of data but relatively faster for decryption process. Moreover, AA_β-decryption uses less memory for systems parameters and accumulators as compared to HIME(R) and Rabin-Takagi.

REFERENCES

Ariffin, M.R.K., Asbullah, M.A., Abu, N.A. and Mahad, Z. 2013. A New Efficient Asymmetric Cryptosystem Based on the Integer Factorization Problem of $N = p^2q$. *Malaysian Journal of Mathematical Sciences*, **7**(S): 19 – 37.

Asbullah, M.A. and Ariffin, M.R.K. 2014. Fast Decryption Method for a Rabin Primitive-based Cryptosystem. *International Journal of Advancements in Computing Technology*, **6**(1): 56 – 67.

Bellare, M. and Rogaway, P. 1995. Optimal asymmetric encryption – How to encrypt with RSA. *Advances in Cryptology – Eurocrypt '94, LNCS 950, Springer-Verlag*: 92 – 111.

Boneh, D. 2001. Simplified OAEP for the RSA and Rabin functions. *Advances in Cryptology – Crypto2001, LNCS 2139, Springer-Verlag*: 275 – 291.

CRYPTREC, 2002. Evaluation Report of HIME(R). Accessed 19th February 2014. Sourced from http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/1019_hime.pdf.

Elia, M., Piva, M. and Schipani, D. 2011. *The Rabin Cryptosystem revisited*. Accessed 19th February 2014. Sourced from *www.arxiv.org*.

Galbraith, S.D. 2012. *Mathematics of Public Key Cryptography*. Cambridge: Cambridge University Press.

Hitachi Ltd. 2002. *Self evaluation report: HIME(R) Cryptosystem*. Accessed 19th February 2014. Sourced from http://www.hitachi.com/rd/yrl/crypto/hime/HIME_R_eval_0310.pdf

Hitachi Ltd. 2002. *Specification of HIME(R) Cryptosystem*. Accessed 19th February 2014. Sourced from http://www.hitachi.com/rd/yrl/crypto/hime/HIME_R_specE.pdf

Kurosawa, K., Ogata, W., Matsuo, T. and Makishima, S. 2001. IND-CCA Public Key Schemes Equivalent to Factoring $n = pq$. *Public Key Cryptography 2001*: 36-47.

Menezes, A.J., van Oorschot, P.C. and Vanstone, S.A. 1997. *Handbook of Applied Cryptography*. CRC Press.

Okeya, K. and Sakurai, K. 2001. Efficient Elliptic Curve Cryptosystems from a Scalar Multiplication Algorithm with Recovery of the y-Coordinate on a Montgomery Form Elliptic Curve. *In Proceeding of the CHES2001, LNCS 2162, Springer-Verlag*: 126 – 141.

Rabin, M.O. 1979. Digitalized Signatures and Public-Key Functions As Intractable As Factorizatio. *Technical Report MIT/LCS/TR-212*, MIT Laboratory for Computer Science.

Takagi, T. 1998. Fast RSA-type Cryptosystem Modulo p^kq. *Advances in Cryptology – Crypto'98, LNCS 1462, Springer-Verlag*: 318 – 326.

Vuillaume, C. 2003. Efficiency Comparison of Several RSA Variants Master Thesis, Fachbereich Informatik der TU-Darmstadt.

Rabin-RZ: A New Efficient Method to Overcome Rabin Cryptosystem Decryption Failure Problem

[1]Z. Mahad and [2]M.R.K. Ariffin
[1,2]Laboratory of Cryptography, Analysis and Structure, Institute for Mathematical Research, Universiti Putra Malaysia
[2]Department of Mathematic, Faculty of Science, Universiti Putra Malaysia
Email: [1]zaharimahad@upm.edu.my, [2]rezal@upm.edu.my

ABSTRACT

We propose a new efficient method to overcome the 4 to 1 decryption failure for the Rabin cryptosystem by reducing the phase space of plaintext from $M \in \mathbb{Z}_{pq}$to $M \in (2^{2n-2}, 2^{2n-1}) \subset \mathbb{Z}_{pq}$, where pq is a product of 2 strong primes and $pq \in (2^{2n}, 2^{2n+2})$. Instead of utilizing the pubic modulus $N = pq$, we use $N = p^2q$. Upon decrypting by using the private modulus $d = pq$ via the Chinese Remainder Theorem, we prove that there exist only one plaintext from the 4 roots obtained that will reside within the interval $(2^{2n}, 2^{2n+2})$. As a result, the decryption failure is overcome and this technique also enhances the decryption process for the Rabin cryptosystem. Furthermore, we make analytical comparison with other methods designed in previous literature to overcome the Rabin cryptosystem problem.

Keywords: Rabin Cryptosystem, Rabin-Williams Cryptosystem, Integer Factorization Problem, Square Root Modulo

1. INTRODUCTION

The Rabin cryptosystem which was introduced in 1979 was designed with its cryptographic primitive being the integer factorization problem (IFP) of 2 strong primes. In comparison with the RSA cryptosystem, Rabin's cryptosystem utilizes the public exponent value $e = 2$. Hence, it is implied by the designers to be an optimal/efficient implementation of the RSA cryptosystem (Rabin, 1979). Furthermore, the Rabin cryptosystem has an established security proof that is complete. It can be observed via the following well established propositions.

Proposition 1. *Let $M \in \mathbb{Z}_{pq}$ and $C \equiv M^2 \ (mod\ pq)$. If the modulus $N = pq$ can be factored, then the square root of C can be obtained and if the square root of C can be obtained then the modulus $N = pq$ can be factored.*

It is well established in the literature that when decrypting the Rabin ciphertext, the receiver of the ciphertext comes upon 4 possible plaintexts. This scenario is due to the process of solving the square root problem by the Chinese Remainder Theorem (CRT). Since 1979, much effort has been put into research to find an efficient method to solve the 4 to 1 decryption failure. It is easy to see that if a polynomial time solution is found, then the Rabin cryptosystem has potential to be utilized due to its short public exponent.

In 2013, Ariffin *et al.* designed an asymmetric scheme based on the IFP together with the square root scenario and is analytically proven to have 1 to 1 decryption (Ariffin *et al.*, 2013). Based on work by Ariffin *et al.*, we managed to redesign the Rabin cryptosystem to be more efficient to use. Instead of focusing on the IFP for $N = pq$, we utilized the IFP within $N = p^2q$ as was established earlier within the Okamoto-Uchiyama's scheme in 1998 (Okamoto and Uchiyama, 1998) and Schmidt-Samoa' system in 2006 (Schmidt-Samoa, 2006).

Our paper is structured as following manner. In Section 2, we reproduce the Rabin cryptosystem and discuss in brief the 4 to 1 decryption scenario. We also describe 3 existing strategies to overcome the 4 to 1 decryption failure. In Section 3, we produce our strategy such that Rabin's cryptosystem has an analytically proven unique decryption. In Section 4, a comparison between our new strategies against all existing 3 strategies is presented. We conclude in Section 5.

2. RABIN CRYPTOSYSTEM

The following is an overview of Rabin's cryptosystem (Rabin, 1979).

Key Generation

INPUT: The size n-bit of the prime numbers.
OUTPUT: A public key $N = pq$.

1. Generate two random and distinct n-bit strong primes p and q satisfying
$$\begin{cases} p \equiv 3 \pmod 4, 2^n < p < 2^{n+1} \\ q \equiv 3 \pmod 4, 2^n < q < 2^{n+1} \end{cases}$$
2. Set $N = pq$.

Encryption

INPUT: The public key $N = pq$ and the plaintext M.
OUTPUT: The ciphertext C.

1. Plaintext is an integer $M \in \mathbb{Z}_{pq}$.
2. Compute $C \equiv M^2 \pmod N$.

Decryption

INPUT: The private key pair (p, q) and the ciphertext C.
OUTPUT: The plaintext M.

1. Solve square root of C via CRT utilizing the private key pair (p, q).
2. Return 4 possible plaintexts M_1, M_2, M_3 and M_4.

Remark 1. The Rabin cryptosystem is known to have decryption failure due to its 4 to 1 mapping. The following is a list of strategies to overcome this feature of the Rabin cryptosystem.

1. **Redundancy in the message** (Menezes *et al.*, 1996). This scheme has a probability decryption failure of approximately $\frac{1}{2^{k-1}}$ where k is the least significant binary string of the message.
2. **Extra bits** (Kurosawa *et al.*, 2001). One will send 2 extra bits of information to specify the square root. The encryption process requires the computation of the Jacobi symbol. This results in a computational overhead which is much more than just computing a single square modulo N.
3. **Williams's technique** (Williams, 1980). The encryption process requires the encrypter to compute a Jacobi symbol. Hence, losing the performance advantage of Rabin over RSA (as in point no. 2).

3. RABIN-RZ: THE MODIFIED RABIN CRYPTOSYSTEM

We begin by describing our modified version of Rabin's cryptosystem, Rabin-RZ.

Key Generation

INPUT: The size n-bit of the prime numbers.
OUTPUT: A public key $N = p^2q$ and private key $d = pq$.

1. Generate two random and distinct n-bit strong primes p and q satisfying

$$\begin{cases} p \equiv 3 \ (\text{mod } 4), 2^n < p < 2^{n+1} \\ q \equiv 3 \ (\text{mod } 4), 2^n < q < 2^{n+1} \end{cases}$$

2. Set $= p^2q$.
3. Set $d = pq$.

Encryption

INPUT: The public key $N = p^2q$ and the plaintext M.
OUTPUT: The ciphertext C.

1. Plaintext is an integer $M \in (2^{2n-2}, 2^{2n-1}) \subset \mathbb{Z}_{pq}$.
2. Compute $C \equiv M^2$ (mod N).

Decryption

INPUT: The private key tuple (d, p, q) and the ciphertext C.
OUTPUT: The plaintext M.

1. Compute $V \equiv C$ (mod d).
2. Solve square root of V via CRT utilizing the private key pair (p, q).
3. Return 4 possible plaintexts M_1, M_2, M_3 and M_4.
4. For $i = 1$ to 4 compute $W_i = \frac{C - {M_i}^2}{N}$.
5. Return the plaintext M_i which produces $W_i \in \mathbb{Z}$.

We now provide the proof of correctness. We begin with the following lemma.

Lemma 1. *Let $N = p^2q$ and $d = pq$. Choose $x \in \mathbb{Z}_d$. If $y \equiv x^2$ $(mod\ N)$ and $V \equiv y$ $(mod\ d)$, then $V \equiv x^2$ $(mod\ d)$.*

Proof. We have

$$y = x^2 + Nk_1 \text{ where } k_1 \in \mathbb{Z} \tag{1}$$

and

$$v = y + dk_2 \text{ where } k_2 \in \mathbb{Z} \tag{2}$$

From (1) and (2) we have

$$v = x^2 + Nk_1 + dk_2$$

Finally,

$$v \equiv x^2 \ (\text{mod } d) \ \blacksquare$$

Proposition 2. *Let C be an integer representing a ciphertext encrypted by the Rabin-RZ scheme. Then, $C \equiv M^2$ (mod N) has a unique solution for M.*

Proof. We begin with the proof of correctness of the decryption procedure. Since $M \in \mathbb{Z}_d$, by solving $V \equiv C$ (mod d) using the CRT we will obtain all 4 roots of V. Also by Lemma 1, indeed $V \equiv M^2$ (mod d). Furthermore, since $M \in \mathbb{Z}_d$ and $d < N$, certainly one of the roots is a solution for $C \equiv M^2$ (mod N).

We now proceed to prove uniqueness. We re-write the congruence relation as the equation $C \equiv M^2 \pmod{N}$ as $C = M^2 - Nt$ with $t \in \mathbb{Z}$. Suppose there are two solutions M_1 and M_2 of the equation $C = M^2 - Nt$ with $t \in \mathbb{Z}, M_1 \neq M_2$ and for $i = 1, 2, M_i < 2^{2n-1}$. Then, ${M_1}^2 - Nt_1 = {M_2}^2 - Nt_2$. Using $N = p^2q$, this leads to

$$ {M_1}^2 - {M_2}^2 = (t_1 - t_2)N. $$

Case 1

$(t_1 - t_2)|({M_1}^2 - {M_2}^2)$. The probability that $(t_1 - t_2)|({M_1}^2 - {M_2}^2)$ and not equal to zero is 2^{-n}. Conversely, the probability that $(t_1 - t_2)|({M_1}^2 - {M_2}^2)$ and equal to zero is $1 - \frac{1}{2^n}$. Thus, ${M_1}^2 = {M_2}^2$ is with probability $1 - \frac{1}{2^n}$ and since $M \in (2^{2n-2}, 2^{2n-1})$, then $M_1 = M_2$. Hence, the equation $C = M^2 - Nt$ has only one solution.

Case 2

$N|(M_1 + M_2)(M_1 - M_2)$. The conditions that should be satisfied is either one of the following

$$ \begin{cases} pq|(M_1 \pm M_2) \\ p|(M_1 \mp M_2) \end{cases} \text{or} \quad \begin{cases} p^2|(M_1 \pm M_2) \\ q|(M_1 \mp M_2) \end{cases} $$

Observe that $pq, p^2 > 2^{2n}$ while $|M_1 \pm M_2| < 2 \cdot 2^{2n-1} = 2^{2n}$. This implies that either condition is not possible. ■

Example

The scenario is A (Along) will send his public key to B (Busu) and Busu will encrypt to Along. Along will choose the primes $p = 100669$, $q = 69859$ and compute $N = 707968400363899$ and $d = 7032635671$. Let says Busu want to sends a message $M = 1439948310$ to Along. Busu will compute

$$ 519659206359828 \equiv 1439948310^2 \pmod{707968400363899} $$

and sends to Along. To decrypt, Along computes

$$ 3691358296 \equiv 519659206359828 \pmod{7032635671} $$

Then, Along uses the CRT and his private keys to compute the four square roots of 3691358296 modulo d, which are

1. $M_1 = 3890433108,$
2. $M_2 = 1439948310,$
3. $M_3 = 5592687361,$
4. $M_4 = 3142202563.$

Then, to determine the correct message Along computes for $i = 1$ to 4:

$$ W_i = \frac{C - {M_i}^2}{N} $$

In this example, only M_2 produces $W \in \mathbb{Z}$.

4. COMPARATIVE ANALYSIS BETWEEN RABIN CRYPTOSYSTEM AND ITS IMPROVEMENTS

In this section, we provide comparison via the complexity order of each Rabin improvement with the fundamental Rabin cryptosystem as was disclosed in 1979. We also provide the advantage and disadvantage of each Rabin improvement.

Algorithm	**Encryption Speed**	**Decryption Speed**
New Rabin-RZ	$O(4n^2 + 3n)$	$O(2n^3 + 12n^2 + 4n)$
Menezes *et al.*	$O(8n^2 + 3n)$	$O(2n^3 + 16n^2)$
Kurosawa *et al.*	$O(5n^2 + n)$	$O(2n^3 + 4n^2 + 2n)$
Williams	$O(5n^2 + 5n)$	$O(n^3 + n^2 + 9n)$
Rabin (1979)	$O(5n^2 + 2n)$	$O(2n^3 + 12n^2 + 4n)$

Table 1: Complexity time between improvements of Rabin cryptosystems

It is obvious that from Table 1, those improvements of Rabin (1979) – that is with no decryption failure is in the following list in descending effectiveness.

1. New Rabin-RZ
2. Kurosawa
3. Williams

Observe we did not include the method by Menezes in the list because of a possible decryption failure. In Table 2, we also provide comparison advantage and disadvantage between improvements of Rabin cryptosystem.

	Menezes Technique	**Kurosawa Technique**	**Williams Technique**	**Rabin-RZ Technique**
Advantage	Overcome Rabin decryption failure with probability $\frac{1}{2^{l-1}}$ where l is number of bits use as redundancy message.	Decryption never fails.	Decryption never fails. Decryption speed is faster than other methods.	Decryption never fails. No extra computation needed during decryption. Domain for plaintext restricted. Instead for any $M \in \mathbb{Z}_{pq}$, we restrict to the interval $2^{2n-2} \leq M \leq 2^{2n-1}$. Encryption speed is faster than other methods. Note: Even though the

				message domain is restricted to the above-mentioned interval, the interval still contains exponentially many message candidates.
Disadvantage	Probability decryption failure of approximately $\frac{1}{2^{l-1}}$.	Slow in term of performance because of the encryption process requires the computational of the Jacobi symbol, this results in a computational overhead, which is much more than just computing a single square modulo N.	Slow in term of performance because of the encryption process requires the computational of the Jacobi symbol, this results in a computational overhead, which is much more than just computing a single square modulo N.	Domain for plaintext is restricted to the interval $2^{2n-2} \leq M \leq 2^{2n-1}$.

Table 2: Comparison advantage and disadvantage between enhancement methods of Rabin cryptosystem

5. CONCLUSION

Through the presentation of this work, we have provided an efficient mechanism to utilize the IFP couple with the square root problem which initially had difficulties to be executed under the circumstances of a 4 to 1 decryption scenario like Rabin cryptosystem.

Extending the results through complexity order analysis, it could be seen that with an encryption and decryption speed of $O(n^2)$ for encryption and $O(n^3)$ for decryption, the Rabin-RZ is able to provide an ideal platform for application that rely on fast encryption and decryption masses. In concluding, we have overcome decryption failure of the Rabin cryptosystem in the most effective manner as opposed to existing methods.

REFERENCES

Rabin, M. O. 1979. Digitalized Signatures and Public-Key Functions as Intractable as Factorization. *Tech. Report MIT/LCS/TR-212, MIT Laboratory for Computer Science*.

Menezes, R. L., van Oorschot, P. C. and Vanstone, S. A. 1996. *Handbook of Applied Cryptography*. CRC Press.

Kurosawa, K., Ogata, W., Matsuo, T. and Makishima, S. 2001. IND-CCA Public Key Schemes Equivalent to Factoring $N = pq$. *Public Key Cryptography 2001*: 36-47.

Williams, H. C. 1980. A Modification of the RSA Public Key Encryption Procedure. *IEEE Trans. Inf. Theory*. 26(6): 726-729.

Ariffin, M. R. K., Asbullah, M. A., Abu, N. A. and Mahad, Z. 2013. A New Efficient Asymmetric Cryptosystem Based on the Integer Factorization Problem of $N = p^2q$. *Malaysian Journal Mathematical Sciences.* 7(S): 19-37.

Okamoto, T. and Uchiyama, S. 1998. A New Public-Key Cryptosystem as Secure as Factoring. *EUROCRYPT-98, Lecture Notes in Computer Science*. 1430: 308-318.

Schmidt-Samoa, K. 2006. A New Rabin-type Trapdoor Permutation Equivalent to factoring. *Electronic Notes in Theoretical Computer Science (ENTCS).* 157(3): 79-94.

Fast Computation in wNAF Expansion Method for Integer Sub-Decomposition Elliptic Scalar Multiplication

[1]Ruma Kareem K. Ajeena and [2]Hailiza Kamarulhaili

[1,2] School of Mathematical Sciences, Universiti Sains Malaysia, 11800 Minden, Penang, Malaysia

Email: [1]ruma.usm@gmail.com, [2]Hailiza Kamarulhaili

ABSTRACT

In this work, we developed an approach that proposed to compute a scalar multiplication on classes of elliptic curve over prime field that have efficiently-computable endomorphisms. Proposed approach, namely, integer sub-decomposition (ISD), based on the GLV method of Gallant, Lambert and Vanstone that was initially proposed in the year 2001, uses speed parallel computation of endomorphisms ψ_i for $i = 1,2$ to compute the multiple kP of a point P of order n lying on an elliptic curve. The decomposition of a scalar k according to GLV method produces two integers k_1 and k_2 lie inside the range $\pm\sqrt{n}$ on the interval $[1, n\text{-}1]$. In more generic this decomposition gives also integers k_1 and k_2 lie outside the range $\pm\sqrt{n}$ on the same interval $[1, n\text{-}1]$. With the last output of decomposition, the GLV idea cannot work. On ISD approach, the solving of this problem, complement the work of GLV method and increasing the percentage of successful computation kP can be done. In this paper, the main idea is to present the parallel computation of ISD elliptic scalar multiplication which is defined by

$$kP= k_{11}P+ k_{12}\psi_1(P)+ k_{21}P+ k_{22}\psi_2(P), \text{ with } |k_{11}|, |k_{12}|, k_{21}|, |k_{22}| < \pm\sqrt{n}$$

through two models that uses the interleaving methods based on parallel computation of $w_{\text{j-NAF}}$ expansions for j=1,2,3,4. The parallel processing on two proposed interleaving methods gives more speeding of computation in comparison with the computations that carry out individually.

Keywords: Elliptic curve cryptography, Scalar multiplication, Parallel interleaving method, Parallel wNAF expansions, Efficiently-computable endomorphisms, Integer sub-decomposition.

1. INTRODUCTION

The application of elliptic curve in cryptography was proposed by Victor Miller (Miller, 1986) and Neil Koblitz (Koblitz, 1987) in the year 1985, and it has attracted increasing attention recently because of their shorter key length requirement when compared to other public-key cryptosystems like RSA. Elliptic curve cryptography (*ECC*) is derived from hardness of the discrete logarithm problem over the additive group of points on an elliptic curve over finite fields. Among the benefits of *ECC* are shorter key length, higher speed and lower power consumption. These advantages are useful for some devices like mobile and wireless which, typically, have limited computational resources and bandwidth.

A pair of keys, which we called them, public and private keys, are used by public-key cryptosystems to carry out cryptographic processes such encryption/decryption of data and signing/ verification of digital signatures. Concerning ECC, private keys are considered as scalar values which should be kept in secret, and public keys are considered as points on *E* which are made public. By giving a secret scalar *k* and points P and kP on *E*, where kP is a multiple of the point P, we can define the elliptic curve discrete logarithm problem as the problem of determining k, with P and kP known Longa and Miri (2011).

Scalar multiplication, in general, represented by kP is considered as the central time-consuming operation in *ECC*. In order to compute this operation, it is necessary to perform iterative addition (*ECADD*) and doubling (*ECDBL*) of points, which we referred to as *ECC* point operations, and their efficient performance is essential to speed up the computation of scalar multiplication Hao *et al.* (2008).

Elliptic curves have a well-known facts and distinct theoretical aspects for the algebraic structures and also the endomorphism applications which can be applied to improve performance fast in elliptic curve scalar multiplication. The extension idea of using Frobenius endomorphism $\psi \in End(E)$ on elliptic curves of arbitrary characteristic $p \geq 3$ splits a large computation into a sequence of cheaper ones so that the overall computational cost is lowered Hankerson *et al.* (2004). Such a technique, which contrary to previous ones, also applied to curves defined over large prime fields, was used, recently, by Gallant, Lambert and Vanstone (Gallant *et al.*, 2001). Their method uses an efficiently computable endomorphism $\psi \in End(E)$ to rewrite kP as

$$kP = k_1 P + k_2 \psi(P), \text{ with } \max\{|k_1|, |k_2|\} = O\left(\sqrt{n}\right).$$

In Gallant, Lambert and Vanstone (GLV) method, the value k is decomposed into the values k_1 and k_2 with the condition that both values are bounded by $\pm\sqrt{n}.$ There are some failing points in GLV method, among them, the main weakness point is, it does not determine the case when the values of k_1 and k_2 are not within the range $\pm\sqrt{n}.$ So, the GLV method will not work with this case. As result, we have proposed new method is called Integer Sub-Decomposition (ISD) Ajeena and Kamarulhaili (2013); Ajeena and Kamarulhaili (2014 a,b) to increase the percentage of a successful computation of $kP.$ The basic idea of ISD method is the sub-decomposition of the values k_1 and k_2 into the values k_{11}, k_{12}, k_{21} and $k_{22}.$ The sub-decomposition from

$$k = k_1 + k_2 \lambda (\bmod n) \tag{1}$$

is elucidated as the following:

$$k_1 = k_{11} + k_{12}\lambda_1 \ (\text{mod } n) \text{ and } k_2 = k_{21} + k_{22}\lambda_2 \ (\text{mod } n). \tag{2}$$

The meaningful role of the method lies in the definition of the group homomorphism (the ISD reduction map)

$$\begin{gathered} T : Z \times Z \to Z/n \\ (i, j) \to i + \lambda_m j \ (\bmod n), \ m = 1, 2. \end{gathered} \tag{3}$$

In particular, we compute the sub-decomposition as follows:

$$\begin{aligned} kP &= k_{11}P + k_{12}\lambda_1 P + k_{21}P + k_{22}\lambda_2 P \\ &= k_{11}P + k_{12}(\lambda_1 P) + k_{21}P + k_{22}(\lambda_2 P) \\ &= k_{11}P + k_{12}\psi_1(P) + k_{21}P + k_{22}\psi_2(P). \end{aligned} \tag{4}$$

The computation of *kP* that defined in equation (4) can be carried out through two proposed methods which uses the computation of the interleaving based on wNAF expansions in parallel. Computing interleavings on the first proposed method is performed in two parallel lines (threads) . The sum of two outputs which form as two elliptic curve points gives the final result of the ISD elliptic scalar multiplication *kP*. Whereas, the computation with the second proposed method is done in one parallel line (thread) to find the final result of *kP* directly.

2. MATHEMATICAL FOUNDATIONS

2.1 Elliptic Curve over prime field

Definition 1. Hankerson *et al.* (2004); Washington (2008) An elliptic curve *E* over a field *K* is defined by an equation

$$E: y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6 \tag{5}$$

where a_1, a_2, a_3, a_4, $a_6 \in K$ and $D_E \neq 0$, where D_E is the discriminant of E.

Definition 2. Hankerson *et al.* (2004) A Weierstrass equation defined over K in equation (5) can be simplified considerably by applying admissible changes of variables. If $Char(K) \neq 2$ or 3, then the admissible change of variables is

$$(x, y) \to \left(\frac{x - 3a_1^2 - 12a_2}{36}, \frac{y - 3a_1x}{216} - \frac{a_1^3 + 4a_1a_2 - 12a_3}{24} \right),$$

transforms E to the curve

$$E' : y^2=x^3+ax+b, \tag{6}$$

where $a, b \in K$. The discriminant of this curve is $D_E = -16(4a^3+27b^2)$. If the elliptic curve E' defined over prime field F_p, then equation (6) is expressed as:

$$E' : y^2=x^3+ax+b \pmod p, \tag{7}$$

where a, $b \in F_p$. The curve E' is said to be non-singular if it has no double zeroes, which means the discriminant $D_E = -16(4a^3+27b^2) \neq 0 \pmod p$.

Definition 3. Hankerson *et al.* (2004); Washington (2008) Let an elliptic curve be defined as E : $y^2=x^3+ax+b \pmod p$ over the finite field with $Char(K) \neq 2,3$. Then, the following arithmetic properties of E should be considered:

1. Identity. $P+\infty=\infty+ P = P$ for all $P \in E(K)$.
2. Negatives**.** If $P = (x, y) \in E(K)$, then $(x, y) + (x,-y)=\infty$. The point $(x,-y)$ is denoted by $-P$ and is called the negative of P, note that $-P$ is indeed a point in $E(K)$. Also, $-\infty=\infty$.
3. Point addition. Let $P = (x_1,y_1) \in E(K)$ and $Q = (x_2,y_2) \in E(K)$, where $P \neq \pm Q$. Then $P + Q = (x_3,y_3)$, where

3.1. If $x_1 \neq x_2$, then

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1} \right)^2 - x_1 - x_2$$

and

$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1} \right)(x_1 - x_3) - y_1.$$

3.2. If $x_1 = x_2$ but $y_1 \neq y_2$, then $P + Q = \infty$.

4. Point doubling. Let $P = (x_1,y_1) \in E(K)$, where

4.1. If $P = Q$ and $y_1 \neq 0$. Then $2P = (x_3, y_3)$, where

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1} \right)^2 - 2x_1$$

and

$$y_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)(x_1 - x_3) - y_1.$$

4.2. If $P = Q$ and $y_1 = 0$, then $P + Q = \infty$.

Definition 4. Gallant *et al.* (2001); Hankerson *et al.* (2004) Assume that *E* is an elliptic curve defined over the finite field F_p. The point at infinity is denoted by O_E. The set of F_p - rational points on *E* forms the group $E(F_p)$. A rational map ψ: $E \rightarrow E$ satisfies $\psi(O_E)= O_E$, which is called an endomorphism of *E*. The endomorphism ψ is defined over F_q, where $q=p^n$ if the rational map is defined over F_q. Thus, for any n ≥ 1, ψ is a group homomorphism of $E(F_p)$ and $E(F_q)$.

Definition 5. Hankerson *et al.* (2004) The endomorphism of elliptic curve *E* defined over F_q is the *m*-multiplication map [*m*]: $E \rightarrow E$ defined by

$$p \rightarrow mP \tag{8}$$

for each $m \in Z$. The negation map [-1]: $E \rightarrow E$ defined by $P \rightarrow -P$ is a special case from *m*- multiplication map.

Lemma 6. Let *E* be an elliptic curve over prime field F_p. And let *P* be a point lies on *E* has large prime order *n*. Assume that $\psi(P)$ is a non trivial endomorphism of *E*. Then $\psi(P) = \lambda P$, where λ is a root of its characteristic polynomial.

Definition 7. Kim and Lim (2003) ISD generators are two sets $\{v_3,v_4\}$ and $\{v_5,v_6\}$ of the linearly independent vectors v_3, v_4 and v_5, v_6 in the kernel of the homomorphism

$$T\text{: Z×Z} \rightarrow Z/n \text{ defined by } (i, j) \rightarrow (i + j\lambda_m) \pmod n, \tag{9}$$

where m=1,2. It is called ISD generators if each component of v_3, v_4 and v_5, v_6 are bounded by $\sqrt{n}$.

Definition 8. Karypis *et al.* (1994); Jones (1999); Barney *et al.* (2010) Parallel computing is a mode of computation which carries out many of repeated calculations simultaneously, or it is operating on the basic dividing the large problems into smaller ones to solve them in parallel. On the other words, parallel computing is the employ of two or more processors in combination to solve computational problem such that this problem can be broken into discrete portions which can solve them concurrently.

3. WINDOW METHODS

Definition 9. Hankerson *et al.* (2004) Suppose $w \geq 2$ is a positive integer. A width-*w* NAF of a positive integer *k* is an expression

$$z = \sum_{i=0}^{l-1} G_i 2^i \tag{10}$$

where each nonzero coefficient G_i is odd, $|G_i| < 2^{w-1}$, $G_{l-1} \neq 0$, and at most one of any *w* consecutive digits is nonzero. The length of the width-*w* NAF is *l* .

Theorem 10 (Properties of width-*w* NAFs). Hankerson *et al.* (2004) Let *k* be a positive integer, then

(i) *k* has a unique width-*w* NAF denoted $NAF_w(k)$.

(ii) NAF$_2$(*k*)=NAF(*k*).

(iii) The length of $NAF_w(k)$ is at most one more than the length of the binary representation of *k*.

(iv) The average density of nonzero digits among all width-*w* NAFs of length *l* is approximately 1/(*w*+1).

3.1 Interleaving Method

For more efficiency, it should speed of the computation of $kP + lQ$ that was used in some kinds of elliptic curve cryptosystems such as in digital signature scheme. This speeding can be achieved through using a simultaneous multiple point multiplication that is also named Shamir trick. The simultaneous method depends on combinations of the points $iP + jQ$, *i, j* =0,1,2,3,...., that precomputed in the precomputation stage. When the precomputed value has only a single point like iP_j then the simultaneous method is called interleaving.

The interleaving, in computation $\sum z_j P_j$ for points P_j and integers z_j , *j*=1,2, permits to apply various methods for each $z_j P_j$ such that the doubling step can be carried out simultaneously. For instance, using width wNAF method with different width values *w*. The computational cost of doubling can be determined through the determination of the maximum number of required doublings for each computing $z_j P_j$. The interleaving method to compute

$$\sum\nolimits_{j=1,2} z_j P_j \tag{11}$$

is given by algorithm (3.51) of Hankerson *et al.* (2004). The representation of integers z_j that written in equation (11) can be done by using w_j NAF expansions that is illustrated in algorithm (3.35) of Hankerson *et al.* (2004). In algorithm (3.51), computing the points iP_j for odd $i < 2^{w_j - 1}$ was performed in precomputation stage. The processing through expansions NAF $w_j(z_j)$ can be carried out simultaneously from left to right with a single doubling of the accumulator at each phase Hankerson *et al.* (2004).

4. PARALLEL COMPUTATION OF W-NAF EXPANSIONS ANDEFFICIENTLY COMPUTABLE ENDOMORPHISMS

4.1 Proposed Model of The Parallel Computation of *w*NAF Expansions in Two Parallel Lines

In the widely of working, it is possible to generalize the computation of wNAF expansion that was introduced in the Definition (9) and Theorem (10) and was enhanced the implementations by the Algorithm (3.51) in Hankerson *et al.*,(2004). The generalization produces through finding the representation of wNAF expansions for four integers simultaneously through the parallel computation which consists from two parallel lines. Each parallel line works through two parallel sub-lines. First parallel sub-line takes (z_1, w_1) as an input to compute $w_1 NAFk_{11}$ expansion, second parallel sub-line takes input (z_2, w_2) to output $w_2 NAFk_{12}$ expansion. Whereas, on the second parallel line, the first parallel sub-line takes (z_3, w_3) as an input to obtain $w_3 NAFk_{21}$ expansion and second parallel sub-line takes input (z_4, w_4) to result $w_4 NAFk_{22}$ expansion.

In the context of the Definition (9) that presented to explain wNAF expansion, it can be generalized for parallel computation concept to represent more than one of integers and compute the wNAF expansions of them as follows:

Definition 11. Let $w_j > 2$, j=1,2,3,4 be positive integers. The width w_j *NAFs* of positive integers z_j are expansions

$$z_j = \sum_{i=0}^{l_j-1} G_{i,j}\, 2^i \quad , \quad \text{with} \quad j=1,2,3,4. \tag{12}$$

such that each nonzero coefficient $G_{i,j}$ is odd and satisfies $\left|G_{i,j}\right| < 2^{w_j-1}-1.$ The leftmost significant bit $G_{l_j-1,j} \neq 0.$ For each expansion, any w_j consecutive bits, at most one of them is nonzero. The length of each w_j NAF expansion is l_j for j=1,2,3,4. The implementation of algorithm (1) gives results of w_j NAF expansions in two parallel lines.

Algorithm 1 of Parallel Computation of the Width-wNAFs of Four Positive Integers in Two Parallel Lines

```
Input: Window width $w_j$, positive integer $z_j$, $j$=1,2,3,4.
Output: The $w_j$ NAF expansions of positive integer $z_j$, $j$=1,2,3,4.
Computation:
$i \leftarrow 0$
First parallel Line:
While ( $z_j \geq 1$, $j$ =1,2) do
    If ( $z_j$ is odd ) then
        $G_{i,j} \leftarrow z_j \bmod 2^w$,
        $z_j \leftarrow z_j - G_{i,j}$
    Else
        $G_{i,j} \leftarrow 0$
    Endif
    $z_j \leftarrow z_j / 2$, $i \leftarrow i+1$.
Endwhile
Second parallel Line:
While ( $z_j \geq 1$, $j$ =3,4) do
    If ( $z_j$ is odd ) then
        $G_{i,j} \leftarrow z_j \bmod 2^w$,
        $z_j \leftarrow z_j - G_{i,j}$
    Else
        $G_{i,j} \leftarrow 0$
    Endif
    $z_j \leftarrow z_j / 2$, $i \leftarrow i+1$.
Endwhile
Return: From first parallel line:
    $\{G_{i-1,1}, G_{i-2,1}, ..., G_{1,1}, G_{0,1}\}$
    and
    $\{G_{i-1,2}, G_{i-2,2}, ..., G_{1,2}, G_{0,2}\}$
From second parallel line:
    $\{G_{i-1,3}, G_{i-2,3}, ..., G_{1,3}, G_{0,3}\}$ and
    $\{G_{i-1,4}, G_{i-2,4}, ..., G_{1,4}, G_{0,4}\}$
```

Remark 12. To make the computation of *w*NAF expansions of the integers k_{11}, k_{12}, k_{21} and k_{22} and later on computing the interleavings $k_{11}P+k_{12}\,\psi_1(P)$ and $k_{21}P+k_{22}\,\psi_2(P)$ or $k_{11}P+k_{12}\,\psi_1(P) + k_{21}P+k_{22}\,\psi_2(P)$ more clearly, it is possible to write z_1, z_2, z_3 and z_4 rather than k_{11}, k_{12}, k_{21} and k_{22} respectively.

4.2 Proposed Parallel Computation of *wNAF* Expansions in One Parallel Line

On the other side, the parallel computations to find *w*NAF expansions can take another proposed model. This model is formed through another type of the parallel designs. It consists of one parallel line contains of four parallel sub-lines. On each parallel sub-line take place processing each value z_j for j=1,2,3,4 that based on the generalization of the original *w*NAF idea which presented in the Definition (11). For instance, the first parallel sub-line takes z_1 and w_1 as inputs and the processing performs to output the result of w_1NAF(z_1) expansion. For all others three parallel sub-lines, the processing occurs in the same way through computing same operations in each parallel sub line. All the computations carries out simultaneously to output results. So, such these computations save a lot of execution time. The parallel implementation results can be obtained by applying the next Algorithm (2).

Algorithm 2 of Parallel Computation of the Width-wNAFs Positive Integers in One Parallel Line

```
Input: Window width $w_j$, positive integer $z_1$, $j$=1,2,3,4.
Output: The $w_j$ $NAF$ expansions of positive integer $z_j$.
Computation:
$i \leftarrow 0$
While ( $z_j \geq 1$, $j$ =1,2,3,4) do
    If ( $z_j$ is odd ) then
        $G_{i,j} \leftarrow z_j \bmod 2^w$,   $z_j \leftarrow z_j - G_{i,j}$
    Else
        $G_{i,j} \leftarrow 0$
    Endif
$z_j \leftarrow z_j / 2$, $i \leftarrow i$+1.
Endwhile
Return: On one line: from
    parallel sub-line1: $\{G_{i-1,4}, G_{i-2,4}, ..., G_{1,4}, G_{0,4}\}$,
    parallel sub-line 2: $\{G_{i-1,2}, G_{i-2,2}, ..., G_{1,2}, G_{0,2}\}$
    parallel sub-line 3: $\{G_{i-1,3}, G_{i-2,3}, ..., G_{1,3}, G_{0,3}\}$
    parallel sub-line 4: $\{G_{i-1,4}, G_{i-2,4}, ..., G_{1,4}, G_{0,4}\}$
```

4.3 Parallel Computing For Efficiently Computable Endomorphisms

Recall the definition of endomorphism (4) and how to compute it though Definition (5) and Lemma (6). Such endomorphism has been computed as a multiplication by λ, where $\lambda \in [1,n\text{-}1]$. For $\lambda_1, \lambda_2 \in [1,n\text{-}1]$, it is possible using Lemma (6) to compute two endomorphisms ψ_1 and ψ_2 such that $\psi_1(P)= \lambda_1 P$ and $\psi_2(P)= \lambda_2 P$, where $\lambda_1 \neq \pm \lambda_2$ and P is a point lies on E over prime field F_p. Since the computation of endomorphisms form as multiplication by λ's then, it is easy to use any algorithm that computes the doubling $\lambda_1 P$ and $\lambda_2 P$. The parallel computation of these doublings is more efficient, because it saves (min (λ_1, λ_2) -1) I+2 (min(λ_1, λ_2)-1)M+ (min(λ_1, λ_2)-1) S from the executing time in comparison with the time that needs for computing two endomorphisms separately, where I, M and S are field operation, inversion, multiplication and squaring respectively.

5. THE PROPOSED INTERLEAVING METHODS TO COMPUTE ISD ELLIPTIC SCALAR MULTIPLICATION

In this section, we proposed two interleaving methods to compute ISD elliptic scalar multiplication kP as follows.

5.1 The Interleaving Method to Compute $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$

Let $E : y^2 = x^3+ax+b$ be an elliptic curve defined over a prime field F_p. And let P be a point on E which has prime order n. The curve E has two efficiently computable endomorphisms $\psi_1(P)$ and $\psi_2(P)$ that computed as shown in Definition (5) and Lemma (6). The computation of $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ can be carried out through the applying of the conception of interleaving method. But, it is not reasonable to compute these interleavings one by one because this procedure needs further of the executing time. So, it requires to propose new model based on the concept of the parallel computation of these interleavings $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ simultaneously.

The basic idea of this model depends on the parallel implementation of wNAF expansions that proposed and implemented by Algorithm (1). This model consists of two parallel lines, each parallel line contains on two parallel sub-lines. On these parallel sub-lines take place the processing of wNAF expansions to represent four sub-scalars k_{11}, k_{12}, k_{21} and k_{22} on the same time. The parallel computation in two lines of interleavings is performed in two stages. The precomputation stage and evaluation stage. The precomputation stage implements to compute the points iP_j for $i \in \{1, 3, ..., 2^{w_j - 1} - 1\}$ for j=1,2,3,4. There are two cases of computing iP_j when j=1,3. Since P_1=P_3=P, so when w_1= w_3 then the computation of iP_1 and iP_3 carries out for one of them. Whereas, if $w_1 \neq w_3$ the computation performs also for one of them that has maximum value between w_1 and w_3.

In evaluation stage takes place the computing of w_j NAF for j=1,2 on the first parallel line. And, the computing of w_j NAF for j = 3,4 on the second parallel line. Furthermore, on each parallel line, the applying of the interleaving method can be performed. The implementation of the proposed model gives by Algorithm (3).

Algorithm 3 of the interleaving method based on *w*NAF expansions in two parallel lines.

```
Input: Integers $z_j$, widths $w_j$ and points $P_j$ for $j$=1,2,3,4.
Output: An interleavings points $\sum_{j=1,2} z_j P_j$ and $\sum_{j=3,4} z_j P_j$.
1.  Precomputation stage:
2.  Compute $iP_j$ for $i \in \{1, 3, ..., 2^{w_j - 1} - 1\}$ where
      $j$=1,2,3,4.
3.  Computation of the endomorphisms $\psi_1(P)$
      and $\psi_2(P)$.
4.  Parallel computation stage:
5.  First parallel line:
6.  Set $P_1 \leftarrow P$, $P_2 \leftarrow \psi_1(P)$.
7.  Run parallel computations width w NAF of
      positive integers Algorithm (1) to compute
      $NAF_{w_j}(|z_j|) = \sum_{i=1}^{l_j - 1} G_{i,j} 2^i$
      for $j$ from 1 to 2 do.
8.  Set $l = max\{ l_j , j=1,2\}$.
9.  Define $G_{i,j}$ = 0 for $i$ from $l_j$ to $l$-1, and for $j$
      from 1 to 2 do
      do
10. If ( $z_j$ < 0) then
11.     Set $G_{i,j} \leftarrow - G_{i,j}$ , $i$ = 0:$l$, $j$=1,2.
12. Else
13.     $G_{i,j} \leftarrow G_{i,j}$ , $i$ = 0:$l$, $j$=1,2.
19.             If ( $G_{i,j} \neq 0$) then
20.                 If ( $G_{i,j} > 0$ ) then
21.                     $Q \leftarrow Q + G_{i,j} P_j$
22.                 Else
23.                     $Q \leftarrow Q - |G_{i,j}| P_j$
24.                 Endif
25.             Else
26.                 $Q \leftarrow Q$.
27.             Endif
28.         Endfor
29. Endfor
30. Second parallel line:
31. Set $P_3 \leftarrow P$, $P_4 \leftarrow \psi_2(P)$.
32. Run parallel computations width-w NAF of
      positive integers algorithm (2) to compute
      $NAF_{w_j}(|z_j|) = \sum_{i=1}^{l_j - 1} G_{i,j} 2^i$ for $j$ from 3 to 4
      do.
33. Set $l' = max\{ l_j , j=3,4 \}$.
34. Define $G_{i,j}$ = 0 for $i$ from $l_j$ to $l'$-1, and for $j$
      from 3 to 4 do
35. If ( $z_j$ < 0) then 36.    Set $G_{i,j} \leftarrow - G_{i,j}$ , $i$ =
          0 : $l'$, $j$=3,4.
37. Else
38.     $G_{i,j} \leftarrow G_{i,j}$, $i$ = 0 : $l'$, $j$=3,4.
39. Endif
40. $Q \leftarrow \infty$.
41. For $i$ from $l'$-1 down to 0 do
42.     $Q \leftarrow 2Q$.
43.     For $j$ from 3 to 4 do
```

```
14. Endif
15. $Q \leftarrow \infty$.
16. For  $i$ from $l$-1 down to 0 do
17.     $Q \leftarrow 2Q$.
18.     For  $j$ from 1 to 2 do
48.                 $Q \leftarrow Q - |G_{i,j}|\ P_j$.
49.             Endif
50.           Else
51.               $Q \leftarrow Q$.
```

```
44.         If ($G_{i,j} \neq 0$) then
45.             If ( $G_{i,j} > 0$ ) then
46.                 $Q \leftarrow Q + G_{i,j}\ P_j$.
47.             Else
52.         Endif
54. Endfor
55. Return $Q$.
```

5.2 Interleaving Method to Compute $k_{11}P+k_{12}\psi_1(P)+ k_{21}P+k_{22}\psi_2(P)$

For speeding up computation of elliptic scalar multiplication *kP*, another model of the interleaving method which consists of four sub-scalar multiplications can be used. The idea is to modify the computation proposed that has been accomplished on two parallel lines to compute interleavings $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ into calculation of the interleaving that defined in equation (4) on one line that comprises of four parallel sub-lines. For each term in equation (4), the interleaving permits to use different methods. For instance, using width *w*NAF with various window widths or other methods. The doubling at each term can be carried out simultaneously. So the cost of the doubling is given by the maximum number of doublings required to compute equation (4).

In computation of the interleaving that defined in equation (4), the processing takes place firstly for all four parallel sub-lines simultaneously to determine the wNAF expansions for integers k_{11}, k_{12}, k_{21}, k_{22} through w_j with j=1,2,3,4. Follows these the determinations, the computation of interleaving that also based on the saved points iP_j for odd $i < 2^{w_j-1}$, j=1,2,3,4 which are obtained from the precomputation stage. Since P_1=P_3=P, in the precomputation stage, then sets of the points iP_1 and iP_3 are equal when $w_1 = w_3$, so the precomputation does for one of them. On the other hand, it is possible determining the maximum number between w_1 and w_3, max (w_j), j =1,3, so the precomputation is performed for
$i < 2^{\max(w_j)-1}$, j=1,3.

Algorithm (4) can be used to compute the interleaving that defined in equation (4). On this algorithm, the computation of the steps (32-47) carries out from left to right and at each phase, there is one doubling operation 2*Q* of the accumulator.

Algorithm 4 of the interleaving method based on wNAF expansions in one parallel lines. Part 1 (Precomputation stage)

Input: Integers k^j, widths w_j and points P_j , j=1,2,3,4.
Output: The interleaving point $\sum_{j=1:4} z_j P_j$.

```
1. Set $P_1 = P_3 = P$.
2. If ( $j$ = 2,4) then
3.    For $i = 1,3,5 : 2^{w_j - 1} - 1$ then
4.           Compute $iP_j$
5.   Endfor
6. Endif
7. If ( $j$=1,3 ) then
8.     If ($w_1$=$w_3$) then
9.          For $i = 1,3,5 : 2^{w_j - 1} - 1$    then
10.               Compute $iP_1$.
11.          Endfor
12.    Else ($w_1 \neq w_3$)
13.    For $i = 1,3 : 2^{\max(w_j) - 1} - 1$  then
14.              Compute $iP_j$.
15.           Endfor
16.    Endif
17.  Return $iP_j$.
18. Computation of the  endomorphism $\psi_1(P)$ and $\psi_2(P)$.
```

Algorithm 4 of the interleaving method based on wNAF expansions in one parallel line. Part 2 (Parallel computation stage)

```
19. Set $P_1 = P_3 = P$, $P_2 = \psi_1(P)$ and $P_4 = \psi_2(P)$.
20. Run parallel computations width-w NAF of
      positive integers Algorithm (2) to compute
      $NAF_{w_j}\left(\left|z_j\right|\right) = \sum_{i=1}^{l_j-1} G_{i,j} 2^i$ for $j$ from 1 to 4 do.
21. Set $l=max\{ l_j ,\ j=1,2,3,4\}$.
22. Define $G_{i,j} = 0$ for $i$ from $l_j$ to $l$-1 and for $j$ from
      1 to 4 do
23. For $i$ from 0 to l do
24.     For $j$ from 1 to 4 do
25.         If ( $z_j < 0$) then
26.             Set $G_{i,j} \leftarrow - G_{i,j}$ .
27.         Else
28.             Set $G_{i,j} \leftarrow G_{i,j}$ .
29.         Endif
30.     Endfor
31. Endfor
32. $Q \leftarrow \infty$.
33. For $i$ from $l$-1 down to 0 do
34.     $Q \leftarrow 2Q$.
35.     For $j$ from 1 to 4 do
36.         If ( $G_{i,j} \neq 0$) then
37.             If ( $G_{i,j} > 0$ ) then
38.                 $Q \leftarrow Q + G_{i,j} P_j$ .
39.             Else
40.                 $Q = Q - |G_{i,j}| P_j$ .
41.             Endif
42.         Else
43.             $Q \leftarrow Q$.
44.         Endif
45.     Endfor
46. Endfor
47. Return $Q$.
```

6. INTERLEAVING METHOD TO COMPUTE PROPOSED ISD SCALAR MULTIPLICATION BASED ON WNAF EXPANSIONS

The idea of GLV method Gallant *et al.* (2001) is the main source on which the ISD method depends to obtain a faster scalar multiplication on an ordinary elliptic curve E defined in equation (7).

This method primarily aims to sub-decompose the values k_1 and k_2 when one or both values are not bounded by $\pm\sqrt{n}$. The sub-decomposition from equation (1) is expressed by these formulas that defined in equation (2).

To accomplish sub-decomposition, one should first find a GLV generator $\{v_1,v_2\}$ by using a GLV generator algorithm in Kim and Lim (2003) for a given n and λ, where n is a large prime order of elliptic curve point P and λ is a root of the characteristic polynomial of endomorphism ψ of E. Consequently, $k \in [1, n\text{-}1]$ is decomposed into k_1 and k_2. This decomposition can be performed using the balanced length-two representation of a multiplier k algorithm in Hankerson *et al.* (2004). Our modified algorithm can then be used to generate the ISD generators $\{v_3,v_4\}$ and $\{v_5,v_6\}$ such that each component of v_3,v_4,v_5 and v_6 is bounded by $\sqrt{n}$ and relatively prime to each other. These generators can be easily computed by solving the closest vector problem in a lattice that is involved in using an extended Euclidean algorithm in Gallant *et al.* (2001); Hankerson *et al.* (2004) k_1 and k_2 can be decomposed again into integers k_{11} k_{12}, k_{21} and k_{22} which means that the sub-decomposition of k as follows:

$$k \equiv k_{11} + k_{12}\lambda_1 + k_{21} + k_{22}\lambda_2 (\mathrm{mod}\ n) \tag{13}$$

with $-\sqrt{n} < k_{11}, k_{12}, k_{21}, k_{22} < \sqrt{n}$ from any ISD generators $\{v_3,v_4\}$ and $\{v_5,v_6\}$. Finally, the scalar multiplication kP can be computed by formula (4).

The formula of ISD elliptic scalar multiplication kP defined in equation (4) can be carried out through the applying of two proposed models to compute the interleaving based on wNAF expansions.

The processing, on the first model that consists from two parallel lines, each line contains on two parallel sub-lines, needs computing $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ separately in two parallel lines but in the same time. The final result of kP comes through the sum of two elliptic points resulting from the computing $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$. Whereas, the second proposed interleaving model to compute kP consists of one parallel line contains on four parallel sub-lines. The processing takes place to compute one interleaving $k_{11}P+k_{12}\psi_1(P) + k_{21}P+k_{22}\psi_2(P)$ to find the final result of kP . The computation with this model is more efficient in comparison with the first one because it gives more speeding up in computation. The ISD elliptic scalar multiplication kP can be compute by using Algorithm (5).

Algorithm 5 of ISD Elliptic Scalar Multiplication

```
Input: The integers $p, n, \lambda, P, w_j$ ,   $j$=1,2,3,4.
Output: $kP$.
1. Precomputation stage:
2. Compute two endomorphisms $\psi_1(P)= \lambda_1 P$ and $\psi_2(P)= \lambda_2 P$.
3. Computation stage:
4. Run GLV generator Algorithm (1) of Kim and Lim (2003) to find the generator $\{v_1, v_2\}$ such that $v_1 \leftarrow (r_{m+1}, -t_{m+1})$ and $v_1 \leftarrow (r_m, -t_m)$ or $v_1 \leftarrow (r_{m+2}, -t_{m+2})$.
5. Run balanced length-two representation of a multiplier Algorithm (3.74) of Hankerson et al. (2004) to decompose $k$ into $k_1$ and $k_2$.
6. Choose randomly $\lambda_1, \lambda_2 \in [1, n\text{-}1]$ such that $\lambda_1 \neq \pm \lambda_2$.
7. Run ISD generators Algorithm (1) of Ajeena and Kamarulhaili (2014 a, b) to find $\{v_3, v_4\}$ and $\{v_5, v_6\}$ such that $v_3 \leftarrow (r_{m_1+1}, -t_{m_1+1})$, $v_4 \leftarrow (r_{m_1+2}, -t_{m_1+2})$ or $(r_{m_1}, -t_{m_1})$, $v_5 \leftarrow (r_{m_2+1}, -t_{m_2+1})$ and $v_6 \leftarrow (r_{m_2+2}, -t_{m_2+2})$ or $(r_{m_2}, -t_{m_2})$.
8. Use Algorithm (2) of Ajeena and Kamarulhaili (2014) to sub-decompose $k_1$ and $k_2$ into $k_1 \equiv k_{11}+ k_{12}\lambda_1 \ (mod\ n)$ an $k_2 \equiv k_{21}+ k_{22}\lambda_2 \ (mod\ n)$ such that $k \equiv k_{11}+ k_{12}\lambda_1 + k_{21}+ k_{22}\lambda_2 \ (mod\ n)$
9. Set $P_1=P_3=P$, $P_2= \psi_1(P)$ and $P_4= \psi_2(P)$.
10. Use parallel computing $w$NAF Algorithm (1) or (2) to compute $w_j$NAF expansions for $j$=1,2,3,4 of integers $k_{11}$, $k_{12}$, $k_{21}$ and $k_{22}$.
11. Use interleaving Algorithm (3) or (4) to compute $kP= k_{11}P+ k_{12}\psi_1(P) + k_{21}P+ k_{22}\psi_2(P)$.
12. Return $kP$.
```

7. CONCLUSION

The computation of ISD elliptic scalar multiplication kP requires the computation of the terms $k_{11}P$, $k_{12}\psi_1(P)$, $k_{21}P$ and $k_{22}\psi_2(P)$. The process to compute those terms one by one separately needs more execution time. So, it is reasonable to find new ways to compute these terms simultaneously. In this work, we proposed two new algorithms to compute these terms jointly through the concept of parallel computations. For the first proposed model, the parallel computation of two interleavings $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ can be carried out on two parallel lines. The implementation on this model firstly takes place on proposed parallel computation of wNAF expansions that also consists of two parallel lines. Each parallel line contains two parallel sub-lines. Each parallel sub-line performs the computations of w_j NAF expansion for j=1,2,3,4. Thereafter, the computation of interleavings $k_{11}P+k_{12}\psi_1(P)$ and $k_{21}P+k_{22}\psi_2(P)$ are determined by two parallel lines. The sum of the final results from the interleavings, that was represented as two elliptic points, give the final result of kP.

On the other hand, it is possible to use another model to compute the terms in ISD elliptic scalar multiplication kP simultaneously. This model uses the parallel computation to compute the interleaving $k_{11}P+k_{12}\psi_1(P) + k_{21}P+k_{22}\psi_2(P)$ on one parallel line. The performance was based on the computation of wNAF expansions in one parallel line. This line consists of four parallel sub-lines that on them are determined the wj NAF expansions.

The processing on the one parallel line model that consists of four parallel sub-lines is more efficient because it implements with less executing time. It provides four times the time used for the implementation. The cost of doubling here determines on the basic the maximum number among k_{11}, k_{12},

k_{21} and k_{22} in comparison with the computation that performed of the terms $k_{11}P$, $k_{12}\psi_1(P)$, $k_{21}P$ and $k_{22}\psi_2(P)$ individually.

8. ACKNOWLEDGMENTS

The authors would like to express their gratitude to the Fundamental Research Grant funded by the Ministry of Higher Education and the School of Mathematical Sciences, Universiti Sains Malaysia.

REFERENCES

Ajeena, R. K. K., and Kamarulhaili, H. 2013. Analysis On The Elliptic Scalar Multiplication Using Integer Sub-Decomposition Method. *International Journal of Pure and Applied Mathematics*, *87*(1), 95-114.

Ajeena, R. K. K., and Kamarulhaili, H. 2014. Point Multiplication using Integer Sub-Decomposition for Elliptic Curve Cryptography. *Applied Mathematics & Information Sciences*, 8(2).

Ajeena, R. K. K., and Kamarulhaili, H. 2014. Comparison Studies on Integer Decomposition Method for Elliptic Scalar Multiplication. *Advanced Science Letters*, 20(2): 526-530.

Barney, B. 2010. Introduction to parallel computing. *Lawrence Livermore National Laboratory*, *6*(13), 10.

Gallant, R. P., Lambert, R. J., and Vanstone, S. A. 2001, January. Faster point multiplication on elliptic curves with efficient endomorphisms. In *Advances in Cryptology—CRYPTO 2001* (pp. 190-200). Springer Berlin Heidelberg.

Hankerson, D., Vanstone, S., and Menezes, A. J. 2004. *Guide to elliptic curve cryptography*. Springer.

Hao, Y., Ma, S., Chen, G., Zhang, X., Chen, H., and Zeng, W. 2008. Optimization Algorithm for Scalar Multiplication in the Elliptic Curve Cryptography over Prime Field. In *Advanced Intelligent Computing Theories and Applications* : 904-911.

Jones, J. E. 1999. A parallel multigrid tutorial. In *Proceedings of the Ninth Copper Mountain Conference on Multigrid Methods, Copper Mountain, CO, April*(pp. 11-16).

Kim, D., and Lim, S. 2003, January. Integer decomposition for fast scalar multiplication on elliptic curves. In *Selected Areas in Cryptography* (pp. 13-20). Springer Berlin Heidelberg.

Koblitz, N. 1987. Elliptic curve cryptosystems. *Mathematics of computation*,*48*(177), 203-209.

Kumar, V., Grama, A., Gupta, A., and Karypis, G. 1994. *Introduction to parallel computing* (Vol. 110). Redwood City: Benjamin/Cummings.

Longa, P., and Miri, A. 2011. *U.S. Patent No. 7,991,162*. Washington, DC: U.S. Patent and Trademark Office.

Miller, V. S. 1986, January. Use of elliptic curves in cryptography. In *Advances in Cryptology—CRYPTO'85 Proceedings* (pp. 417-426). Springer Berlin Heidelberg.

Washington, L. C. (2008). *Elliptic curves: number theory and cryptography*. CRC press.

Pseudo τ - Adic Non Adjacent Form for Scalar Multiplication on Koblitz Curves

[1]Faridah Yunos, [2]Kamel Ariffin Mohd Atan, [3]Muhammad Rezal Kamel Ariffin and [4]Mohamad Rushdan Md Said

Institute for Mathematical Research,
Universiti Putra Malaysia,
43400 Serdang, Selangor, Malaysia.
E-mail:[1]faridahy@.upm.edu.my , [2]kamel@upm.edu.my,[3]rezal@upm.edu.my,
[4] mrushdan@upm.edu.my

ABSTRACT

In ECC, scalar multiplication is the dominant operation, namely computing nP from a point P on an elliptic curve where the multiplier n is an integer, defined as the point resulting from adding $\boldsymbol{P}+\boldsymbol{P}+\cdots+\boldsymbol{P}$, n times. The $\boldsymbol{\tau}$-NAF proposed by Solinas, is one of the most efficient algorithms to compute scalar multiplications on Koblitz curves. In this paper, we introduced an equivalent multiplier to $\boldsymbol{\tau}$-NAF namely pseudoTNAF. It is based on the idea of transforming the $\boldsymbol{\tau}$-NAF expression to a reduced $\boldsymbol{\tau}$-NAF that has been done by some researchers. It can eliminate the elliptic doublings in scalar multiplication method, and double the number of elliptic additions. We provide the formula for obtaining a total of lattice points in Voronoi region of modulo $\boldsymbol{r}+\boldsymbol{s\tau}$ where $\boldsymbol{r}+\boldsymbol{s\tau}$ an element of ring $\boldsymbol{Z(\tau)}$. This helps us to find all the multipliers $\boldsymbol{n}$ that based on $\boldsymbol{\tau}$-NAF. We also discuss the estimation of operational costs when using pseudoTNAF as a multiplier of **scalar multiplication**

Keywords: Scalar multiplication, Koblitz curve, density, Voronoi region, Hamming weight.

1. INTRODUCTION

The Koblitz curves are a special type of curves for which the Frobenius endomorphism can be used for improving the performance of computing a scalar multiplication (Koblitz, 1987). The Koblitz curves are defined over F_2 as follows

$$E_a\colon y^2+xy=x^3+ax^2+1$$

where $a\in\{0,1\}$ (Koblitz (1992)). The Frobenius map $\tau\colon E_a(F_{2^m})\mapsto\ E_a(F_{2^m})$ for a point $P=(x,y)$ on $E_a(F_{2^m})$ is defined by

$$\tau(x,y)=(x^2,y^2)\ ,\qquad \tau(O)=O$$

where O is the point at infinity. It stands that $(\tau^2\ \ +2)P=t\tau(P)$ for all $P\in E_a(F_{2^m})$, where the trace, $t=(-1)^{1-a}$. Thus, it follows that the Frobenius map can be considered as a multiplication with complex number $\tau=\frac{t+\sqrt{-7}}{2}$ (Solinas (2000)).

In the ensuing discussion, the following definitions will be applied.

Definition 1 (Yunos and Mohd Atan, 2013**).** A τ-adic Non-Adjacent Form of nonzero $\bar{n}$ an element of $Z(\tau)$ is defined as $\tau\text{-}NAF(\bar{n})=\sum_{i=0}^{l-1}c_i\tau^i$ where l is the length of an expansion of $\tau\text{-}NAF(\bar{n})$, $c_{l-1}\neq 0$, $c_i\in\{-1,0,1\}$ and $c_ic_{i+1}=0$.

Definition 2 (Yunos and Mohd Atan, 2013**).** A Hamming weight is defined as the number of elements -1 and 1 of an expansion of an element of $Z(\tau)$

Definition 3 (Hankerson *et al.*, 2004). Let $N\colon Z(\tau)\to Z$ as a function of norm and $\alpha=x+y\tau$ an element of $Z(\tau)$. The norm of α is $N(\alpha)=x^2+txy+2y^2$ where and $t=(-1)^{1-a}$.

Definition 4. An operational costs is defined as the cost in terms of running time to compute the scalar multiplication of the number of doubling and addition operations.
Definition 5 (Solinas, 2000). Let $\lambda \in\ Q(\tau)$, and $\lambda = \lambda_0 + \lambda_1\tau$ with λ_0, $\lambda_1 \in R$. U is a region in the (λ_0, λ_1)-plane by the inequalities below.

$$-1 \leq 2\lambda_0 + t\lambda_1 < 1$$
$$-2 \leq\ \lambda_0 + 4t\lambda_1 < 2$$
$$-2 \leq \lambda_0 - 3t\lambda_1 < 2.$$

Definition 6. A Voronoi region of $\psi\, Z(\tau)$ is denoted by

$$V = \{\lambda\, \psi : \lambda \in\ U, \psi \in\ Z(\tau)\}.$$

In this paper, we introduced an equivalent multiplier to τ-NAF namely pseudoTNAF. This is based on the idea of transforming the τ-NAF to a reduced τ-NAF developed by some researchers for example Solinas (2000) and Joye and Tymen (2001). We begin in Section 1 with the concept of reduction in the ring Z (τ). In Section 2, we prove the equivalence of both expansion of τ-NAF and pseudoTNAF, also refined one of the properties of ρ so that the scalar multiplication is not heading to infinity. Two other properties have been discussed by Yunos *et al.* (2014). In Section 3, we give the formula to find the number of elements in Voronoi region of of $\rho\frac{\tau^m -1}{\tau-1}Z(\tau)$ and produced the algorithm for finding all points in mod $(r + s\tau)$. This algorithm is important to facilitate the process of getting all pseudoTNAF for all elements in $\rho\ \frac{\tau^m -1}{\tau-1}$ developed in Section 4. The discussion concluded with the estimating of average Hamming weight of pseudoTNAF with maximum length.

2. MODULO REDUCTION IN Z(τ)

The region U that was mentioned in Definition 5 form a hexagon with six vertices with their norms $\frac{4}{7}$ respectively. If the vertices is represented by $\lambda = \lambda_0 + \lambda_1\tau$ then $\lambda_0^2\ +\ t\,\lambda_0\lambda_1\,\tau + 2\,\lambda_1^2\ =\ \frac{4}{7}$ form an ellipse. However, if λ a point in the ellipse, then the norm is less than $\frac{4}{7}$. Thus, we have

$$N(\lambda) \leq\ \frac{4}{7}.$$

The rounding process of $\lambda \in\ Q(\tau)$ is done via

$$\text{Round}(\lambda) =\ \left\lfloor \lambda + \frac{1}{2} \right\rfloor \tag{1}$$

so that $\lambda \in Z\ (\tau)$. The value of $\left\lfloor\lambda + \frac{1}{2}\right\rfloor$ is the largest integer that does not exceed $\lambda + \frac{1}{2}$.

The reduction concept in the field of rational integer has been discussed by Solinas (2000). The reduction of $x'\,mod\ z'$ is expressed as $x' \equiv y'\,mod\ z'$ where y' and $z' > 1$ are integers, $-\frac{z'}{2} \leq\quad x\,' < \frac{z'}{2}$ and $N(x') < \frac{1}{2}N\ (z\ ')$. This reduction is then expanded to the ring of $Z(\tau)$ i.e.

$$x'' \equiv\ y''\,\text{mod}\,z''$$

where y'' and z'' are elements of $Z(\tau)$. Division y'' by z'' produces the residue x'' and it can be written as

$$y'' = \kappa\, z'' + x''$$

where $\kappa \in Z(\tau)$. Suppose that $\lambda = \frac{y''}{z''}$. It generates λ an element of $Q(\tau)$. The rounding process of λ to an element of $Z(\tau)$ is done via (1) so that κ an element of $Z(\tau)$. Therefore, the residue x'' is obtained from equation

$$x'' = y'' - \kappa\, z''$$

where $\kappa = Round(\lambda)$. Now, the above expression becomes

$$x'' = z''(\lambda - Round(\lambda))$$

where $N\big(\lambda - Round(\lambda)\big) \leq \frac{4}{7}$. To avoid scalar multiplication towards to infinity, the residue x'' must have a norm as small as possible i.e.

$$N(x'') \leq \frac{4}{7} N(z''). \tag{2}$$

ROUTINE 62 in Solinas (2000) for division in $Z(\tau)$ provides a detail reduction steps for x'' mod z''. This algorithm has been used by Solinas in the reduction of x'' mod $(\tau^m - 1)$ and x''mod $\frac{\tau^m -1}{\tau - 1}$..

3. EQUIVALENCE OF τ-ADIC NAF

The main purpose of obtaining an equivalence of τ –adic NAF is to maintain the situation so that the doubling operation in elliptic scalar multiplication method can be eliminated and the number of elliptic additions can be doubled. Let G be a set of points on Koblitz Curve. Let γ and β are element of $Z(\tau)$ such that $\gamma\, P = \beta P$ for all $P \in G$. Therefore τ –NAF(γ) is equivalent to τ –NAF(β) with respect to G. The following proposition provides a guideline on how the equivalence of two τ –NAF occur on the entire set $G = E_a\, F_{2^m}$.

Proposition 1. If γ, β and ρ are elements of $Z(\tau)$ where $\gamma \equiv \beta$ mod $\rho\, \frac{\tau^m -1}{\tau - 1}$ then $\gamma\, P = \beta P$ for all $P \in E_a\, F_{2^m}$. τ – NAF(γ) is equivalent to τ –NAF(β) in set $E_a\, F_{2^m}$.

Proof

Given γ, β and ρ are elements of $Z(\tau)$ and $\gamma \equiv \beta$ mod $\rho\, \frac{\tau^m -1}{\tau - 1}$. Therefore,

$$\gamma\ = \beta + \rho\, \frac{\tau^m\ - 1}{\tau - 1} \cdot \kappa$$

for $\kappa \in Z(\tau)$. Thus,

$$\gamma\, P\ = \beta\, P + \rho\, \frac{\tau^m\ - 1}{\tau - 1} \cdot \kappa \cdot\ P.$$

Since $(\tau^m - 1)P = O$ (refer Proposition 65 on page 221 Solinas(2000)), then

$$\begin{aligned} &\gamma\, P\ = \beta\, P + O \\ &\gamma P = \beta\, P. \end{aligned}$$

Hence, τ – NAF(γ) is equivalent to τ –NAF(β) in set $E_a\, F_{2^m}$. ■

In this paper, γ as a product of reduction modulo $\rho\ \frac{\tau^m-1}{\tau-1}$ as shown in Proposition 1 can be used as a multiplier for P. If the multiplier of P is equal to $\rho\ \frac{\tau^m-1}{\tau-1}$, then the scalar multiplication getting towards infinity due to

$$\rho\ \frac{\tau^m-1}{\tau-1}(P)=(\tau^m\ -1)\ P\cdot\frac{\rho}{\tau-1}\ =O.$$

To avoid this situation, $\gamma\in\ Z(\tau)$ with the norm as small as possible should be selected. Solinas(2000) has given condition (2) of $N(x'')$ for $x''\equiv\ y''\bmod z''$ for any $z''\in\ Z(\tau)$. With this guideline, the following condition must be chosen. That is,

$$N(\gamma)\leq\ \frac{4}{7}N\left(\rho\frac{\tau^m\ -1}{\tau-1}\right). \tag{3}$$

This study named the τ –NAF(γ) with condition (3), $\rho\neq\ 1$ and $\rho\neq\tau-1$ as pseudo τ –adic non adjacent form of γ and abbreviated as pseudoTNAF(γ).

By substituting γ with $\bar{n}$, pseudoTNAF($\bar{n}$) can be used in place of τ-NAF($\bar{n}$) for elliptic scalar multiplication on the set $E_a\ F_{2^m}$. The elliptic operational costs with pseudoTNAF($\bar{n}$) can be calculated by estimating an average Hamming weight of it's expansion. Such average is a product of multiplying an average density among pseudoTNAF($\bar{n}$) that have the maximum length $\bar{l}$ by the size of maximum length.

4. VORONOI REGION OF $\rho\frac{\tau^m-1}{\tau-1}Z(\tau)$

In this section, we give a geometric description of element n that is a result from the modulo reduction of $\rho\frac{\tau^m-1}{\tau-1}$. The following theorem is important in order to get the Voronoi region of $\rho\frac{\tau^m-1}{\tau-1}Z(\tau)$ where $\rho\frac{\tau^m-1}{\tau-1}\neq 0$.

Theorem 1. Suppose that λ is in the interior of region U, $\psi\ =\ r+s\ \tau$ and $\omega\ =\ N(\psi)$. Then the following properties are true for every nonzero $\psi\in\ Z(\tau)$.

i. If $N(\lambda)<\ N(\lambda\pm\psi)$ then $|(2r+ts)\lambda_0+(tr+4s)\lambda_1|<\omega$.
ii. If $N(\lambda)<N(\lambda\pm\tau\psi)$ then $|(r-3ts)\lambda_0+(4tr+2s)\lambda_1|<2\omega$.
iii. If $N(\lambda)<N(\lambda\pm\bar{\tau}\psi)$ then $|(r+4ts)\lambda_0-(3tr-2s)\lambda_1|\ <2\omega$.

Proof

Suppose that $\lambda=\lambda_0+\lambda_1\tau$ and $\psi\ =\ r+s\ \tau$.

i. If $N(\lambda)<\ N(\lambda+\psi)$ then we have

$$\begin{aligned}\lambda_0^2\ +\ t\lambda_0\,\lambda_1\ +\ 2\lambda_1^2 &< \lambda_0^2+2\lambda_0\,r+r^2+t(\,\lambda_0\,\lambda_1\ +\lambda_0\,s+r\,\lambda_1+rs)\\ &\quad +2\,(\lambda_1^2+2\lambda_1 s+s^2)\\ 0 &< \lambda_0(2r+ts)+\lambda_1(tr+4s)+(r^2+trs+2s^2)\\ -(r^2+trs+2s^2) &< \lambda_0(2r+ts)+\lambda_1(tr+4s)\\ -N(\psi) &< \lambda_0(2r+ts)+\lambda_1(tr+4s).\end{aligned}$$

Take $N(\psi)=\omega$, then we have

$$-\omega\ <\lambda_0(2r+ts)+\lambda_1(tr+4s). \tag{4}$$

And also by using the similar way as above, if $N(\lambda)\ <\ N(\lambda-\psi)$, then

$$\omega\ >\lambda_0(2r+ts)+\lambda_1(tr+4s). \tag{5}$$

From (4) and (5), we obtain

$$-\omega \ < \lambda_0(2r + ts) + \lambda_1(tr + 4s) < \omega.$$

ii. If $N(\lambda) < \ N(\lambda + \tau\psi)$, then

$$\begin{aligned} \lambda_0^2 + t\lambda_0\,\lambda_1 + 2\lambda_1^2 &< \lambda_0^2 - 4\lambda_0\,s + 4s^2 + \\ & \quad t(\,\lambda_0\,\lambda_1 + \lambda_0\,r + \lambda_0\,st - 2\lambda_1 s - 2rs - 2s^2 t\,) \\ & \quad +2\,(\,\lambda_1^2 + 2\lambda_1 r + 2\lambda_1 st + r^2 + 2rst + s^2) \\ 0 &< \lambda_0(tr - 3s) + \lambda_1(4r + 2st) + \ 2(r^2 + trs + 2s^2) \\ -2\omega &< \lambda_0(tr - 3s) + \lambda_1(4r + 2st). \qquad (6) \end{aligned}$$

And also by using the similar way as above, if $N(\lambda) < \ N(\lambda - \tau\psi)$, then we get

$$2\omega > \lambda_0(tr - 3s) + \lambda_1(4r + 2st). \qquad (7)$$

From (6) and (7), we obtain

$$-2\omega \ < \lambda_0(tr - 3s) + \lambda_1(4r + 2st) \ < \ 2\omega.$$

Since $\frac{1}{t} = t$ for $t = \pm 1$ then

$$-2\omega \ < \lambda_0(r - 3ts) + \lambda_1(4tr + 2s) \ < \ 2\omega.$$

iii. If $N(\lambda) < \ N(\lambda + \bar{\tau}\psi)$, then

$$\begin{aligned} \lambda_0^2 \ + \ t\lambda_0\lambda_1 + 2\lambda_1^2 &< \lambda_0^2 + 2t\lambda_0\,r + 4\lambda_0\,s + 4s^2 + 4trs \\ & \quad +t(\lambda_0\,\lambda_1 - \lambda_0\,r + t\lambda_1\,r - tr^2 + 2\lambda_1 s - 2r\,s\,) \\ & \quad +2\,(\,\lambda_1^2 - 2\lambda_1 r + r^2) \\ 0 &< \lambda_0(tr + 4s) + \lambda_1(-3r + 2ts) + 2(r^2 + trs + 2s^2) \\ -2\omega &< \lambda_0(tr + 4s) + \lambda_1(-3r + 2ts) \qquad (8) \end{aligned}$$

And also by using the similar way as above,, if $N(\lambda) < \ N(\lambda - \ \bar{\tau}\psi)$, then we have

$$2\omega > \lambda_0(tr + 4s) + \lambda_1(-3r + 2t) \qquad (9)$$

From (8) and (9), we obtain

$$-2\omega \ < \lambda_0(tr + 4s) + \lambda_1(-3r + 2ts) < \ 2\omega.$$

Since $\frac{1}{t} = t$ for $t = \pm 1$ then

$$-2\omega \ < \lambda_0(r + 4ts) + \lambda_1(-3tr + 2s) \ < \ 2\omega.$$

■

As a result, the Voronoi region of $\psi\, Z(\tau)$ is given by the inequalities

$$\begin{aligned} -\omega &\le (2r + ts)\lambda_0 + (tr + 4s)\lambda_1 < \omega \\ -2\omega &\le (r + 4ts)\lambda_0 - (3tr - 2s)\lambda_1 < 2\omega \\ -2\omega &\le (r - 3ts)\lambda_0 + (4tr + 2s)\lambda_1 < 2\omega. \end{aligned}$$

The above result is similar to the definition of region V made by Solinas(2000) with the assumption of the variable w is the norm of $\psi = \frac{\tau^m - 1}{\tau - 1}$. Wheareas, in our study, w is the norm of any element in $Z(\tau)$. However, this study has confirmed that the definition of V made by him can be apply to the case of ψ is any element in $Z(\tau)$. There is suggestion in Theorem 6 of Gordon(2008) say that the elements in the

Voronoi region with $\tau^m - 1$ can be obtained from the distribution of elements in $L = \{0,1,2,\dots,N\,(\tau^m\ - 1)\ - 1\}$ by $\tau^m - 1$. In other words, the reduction of L mod $\tau^m - 1$ produces a total of $N(\tau^m - 1) - 1$ distinct lattice points in the Voronoi region of $(\tau^m - 1)\,\mathrm{Z}\,(\tau)$. Solinas(2000) also follow the same suggestion i.e. the Voronoi region with $\frac{\tau^m - 1}{\tau - 1}$ can be derived from the division of each element in $\left\{0,1,2,\dots,N\left(\frac{\tau^m - 1}{\tau - 1}\right) - 1\right\}$ by $\frac{\tau^m - 1}{\tau - 1}$. This division also generates a total of $N\left(\frac{\tau^m - 1}{\tau - 1}\right)$ distinct lattice points. This is reinforced with Proposition 75 of Solinas(2000) which says that the lattice points in the region Voronoi is exactly $n\ mod\ \psi$ for any $\psi \in\ Z\ (\tau)$ where $0 \le\ n\ < N(\psi)$.

The question now, does this proposition could be applicable in the case of $\psi\ = \rho\ \frac{\tau^m - 1}{\tau - 1}$? Suppose that Voronoi region with $\rho\ \frac{\tau^m - 1}{\tau - 1}$ and written as $V = \left\{\rho\ \frac{\tau^m - 1}{\tau - 1}\ \lambda : \lambda\ \in\ U\right\}$. The following describes the Voronoi region of $2\frac{\tau^3 - 1}{\tau - 1}\ Z(\tau) = (-2 + 4\tau)Z(\tau)$ with $t = 1$.

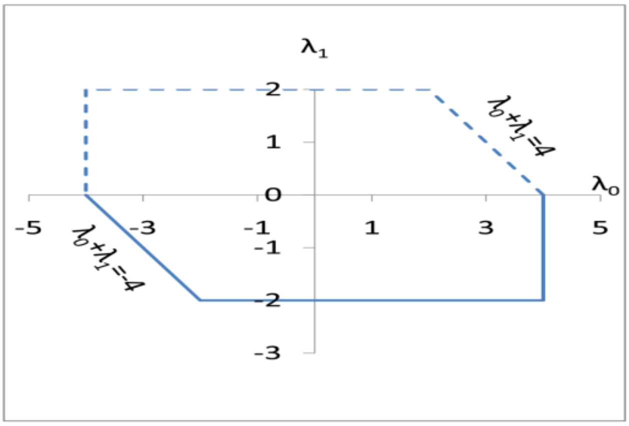


Figure 1: A Voronoi Region of $(-2 + 4\tau)Z(\tau)$ with $t = 1$.

In the above figure, the division of every element in $\{0,1,2,\dots,27\}$ by $-2\ + 4\tau$ produces $(0,0), (1,0), (2,0), (3,0), (-2,-2), (-1,-2), (0,-2), (1,-2), (2,-2), (3,-2), (4,-2), (-3,0), (-2,0),$ $(-1,0), (0,0), (1,0), (2,0), (3,0), (-2,-2), (-1,-2), (0,-2), (1,-2), (2,-2), (3,-2), (4,-2), (-3,0),$ $(-2,0)$ and $(-1,0)$ respectively. That is , there exist 14 distinct lattice points with 14 pairs of the same points that satisfy the region V. In this case, the number of this distinct points is a total of $2N(-1 + 2\tau) = 14$ (i.e. supposed to be $N(\psi) = 28$ by Proposition 75 of Solinas(2000)). The actual points in the above figure have been produced from the reduction of $\{0,1,2,\dots,13\}\ mod\ (-2 + 4\tau)$. Its show us that the Proposition 75 of Solinas(2000) is not applicable for the case of $\psi\ = 2\ \frac{\tau^3 - 1}{\tau - 1}$. Now, we observing another case by studying first the property of $r + s\tau$.

Theorem 2. Suppose that $r + s\tau\ = \rho'(\ r' + s'\tau)$ where $\rho' \in\ Z$ and $r' + s'\tau \in\ Z(\tau)$ then $Z \cap (r + s\tau)Z(\tau) = \rho' N(r' + s'\tau)Z$.

Proof

Let $r + s\tau\ = \rho'(\ r' + s'\tau)$, then we obtain

$$\begin{aligned} Z \cap (r + s\tau)Z(\tau) &= Z \cap \rho'(r' + s'\tau) \\ &\quad \{\dots, -2(r' + s'\overline{\tau}), -(r' + s'\overline{\tau}), 0, r' + s'\overline{\tau}, 2(r' + s'\overline{\tau}) \\ &\quad \dots, (i + j\tau)(r' + s'\overline{\tau}), \dots \mid\ i,j \in\ Z\,\} \\ &= Z \cap \rho' N(r' + s'\tau)\{\dots, -2, -1, 0, 1, 2, \dots, i + j\tau, \dots \mid\ i,j \in\ Z\} \\ &= Z \cap \rho' N(r' + s'\tau)\{\,Z \cup\ Z(\tau)\} \end{aligned}$$

$$= \quad \rho' N(r' + s'\tau)Z. \qquad \blacksquare$$

As a result, the number of lattice points in the interior of Voronoi derived from the division of each element in $\{0,1,2,\dots,N(r+s\tau)-1\}$ by $r+s\tau$ can be obtained by the following corollary.

Corollary 1. Let Voronoi region V is define as Definition 6 and $\psi = r + s\tau$ an element of Z(τ). The number of lattice points in V can be obtained from formula $|\rho'|\, N(r'+s'\tau)$ such that $r + s\tau = \rho'\ (r'{+}s'\tau)$ where $\rho' \in Z$ and $r' + s'\tau \in Z(\tau)$.

Proof

Let $r + s\tau = \rho'\ (r'{+}s'\tau)$ where $\rho' \in Z$ and $r' + s'\tau \in Z(\tau)$. Since $Z \cap (r+s\tau)Z(\tau) = \rho' N(r' + s'\tau)Z$ from Theorem 2, then the number of lattice points in V can be obtained from $|\rho'|\, N(r' + s'\tau)$. ■

For the case that $\psi = \rho\, \frac{\tau^m - 1}{\tau - 1}$, this expression needs to be converted into $r + s\tau$ via Lucas sequence before factoring ψ into $\rho'\ (r' + s'\tau)$. Refer Figure 1, $r + s\tau = -2 + 4\tau$ can be factorized into $2(-1 + 2\tau)$. Hence, $Z \cap (-2 + 4\tau)\, Z(\tau) = 2N(-1 + 2\tau)Z = 14Z$ where the coefficient of Z which is 14 is the number of points in the interior of Voronoi region of $(-2 + 4\tau)Z(\tau)$. The following is an algorithm for obtaining the number of lattice points in the region V of $(r + s\tau)Z(\tau)$, which uses Corollary 1. $N(P_u)$ in the following algoritma is the norm for every lattice points in the Voronoi region. They must be less than or equal to $\frac{4}{7}N(r + s\tau)$ so that the scalar multiplication will not approaches to infinity.

Algorithm 1. (Finding all points in mod $(\boldsymbol{r + s\tau})$)

```
Input: Integers $r, s, \rho', r'$ and $s'$ such that $r + s\tau = \rho'\ (r' + s'\tau)$.
Output: All points $x_u + y_u\tau \in mod\ (r + s\tau)$ and their norms respectively.
Computation:
(1) $N(r' + s'\tau) \leftarrow (r')^2 + tr's' + 2(s')^2$.
(2) $N' \leftarrow |\rho'|\, N(r' + s'\tau)$.
(3) For $u$ from 0 to $N' - 1$ do
        $k \leftarrow r'u + ts'u$.
        $l \leftarrow - s'u$.
        $h \leftarrow (r')^2 + tr's' + 2(s')^2$.
        $\lambda_0 \leftarrow \frac{k}{h}$.
        $\lambda_1 \leftarrow \frac{l}{h}$.
      Use Algorithm 3.63 of Hankerson(2004) for rounding of $w$ and $z$ in $Q(\tau)$ to get elements in $Z(\tau)$.
        $x_u \leftarrow u - r'w + 2s'z$.
      . $y_u \leftarrow -s'w - r'z - ts'z$.
        $P_u \leftarrow x_u + y_u\tau$.
        $N(P_u) \leftarrow x_u^2 + tx_uy_u + 2y_u^2$.
(4) Return $(P_u, N(P_u))$.
```

Once all elements in mod $\rho\, \frac{\tau^m - 1}{\tau - 1}$ are known via Algorithm 1, so now it is easy to get the pseudoTNAF expansion for each element. The algorithm is presented in the next section.

5. DENSITY FOR SOME ELEMENTS IN MODULO $\boldsymbol{\rho\, \frac{\tau^m - 1}{\tau - 1}}$

In this section, we discuss a method for obtaining the density of some elements in mod $\rho\, \frac{\tau^m - 1}{\tau - 1}$. Firstly, we find the Hamming weight of pseudoTNAF expansions together with their lengths. After that, we calculate the density of each element by dividing the Hamming weight by its length. This density are very important to determine the operating costs of elliptic scalar multiplication with the multiplier for P is based on pseudoTNAF. We developed the following algorithm to obtain all pseudoTNAF for all elements in $\rho\, \frac{\tau^m - 1}{\tau - 1}$.

Algorithm 2.

```
Input: Integers $x_u, y_u$ for $u \in \{0,1,2,\dots,N'-1\}$
Output: pseudoTNAF($x_u + y_u\tau$)
Computation:
        $(c_0, c_1) \leftarrow (x_0, y_0)$
        $pseudoTNAF_0 \leftarrow 0$
        For $u$ from 1 to $N' - 1$ do
                $(c_0, c_1) \leftarrow (x_u, y_u)$
                $i \leftarrow 0$
                While $c_0 \neq 0$ or $c_1 \neq 0$ do
                        If $c_0$ is odd then
                            $v_i \leftarrow 2 - (c_0 - 2\,c_1 \bmod 4)$
                            $c_0 \leftarrow c_0 - v_i$
                        else
                            $v_i \leftarrow 0$
                Endwhile
                $R \leftarrow c_0$
                $(c_0, c_1) \leftarrow (c_1 + \frac{t \cdot c_0}{2}, -\frac{R}{2})$
                $i \leftarrow i + 1$
                $j \leftarrow i$
        Output $pseudoTNAF_u(v_0, v_2, \dots, v_{j-1})$
```

Through the above algorithm, the number of bits of N' can be used as a guide to find an integer u which is a multiplier of scalar multiplication. For example, if $a = 0, \rho_0 = 1, \rho_1 = -1$ and $m = 163$, then $N' = 11692013098647223345629473816263631617836683539492$. The maximum number of bits available for integer u in $\bmod(-3334746503586958025881129 - 1824026374634505274957943\,\tau)$ is about 163 bits. In other words, we can get all integers u with their sizes between 1 to 163 bits. The question now, does all integers u from 1 to $N' - 1$ suitable to be used as the multipliers for scalar multiplication? According to Solinas(2000), the sizes of the practical multiplier for ECC is between 96 to 128 bits for $m = 163$. It is not necessary to examine all points u from 1 to $N' - 1$.That is, the integers u should be in element of [39614081257132168796771975168,340282366920938463463374607431768211455]. That means, there are 340282366881324382206242438634996236288 choice of multipliers that might be used in the scalar multiplication. Several options u can be made randomly by doing part by part looping. For example, to obtain the pseudoTNAF for u in Table 1, the command ‘ for u from 0 to $N' - 1$do’ and ‘for u from 1 to $N' - 1$ do’ can be replaced by the command ‘for u from 79228162514264337593543950335 to 79228162514264337593543950339 do'.

u	Size of bits	Length of pseudoTNAF	Hamming weight	Density (5decimal places)
79228162514264337593543950335	96	157	29	0.18471
79228162514264337593543950338	96	157	28	0.17834
79228162514264337593543950336	97	157	30	0.19108
79228162514264337593543950337	97	157	29	0.18471
79228162514264337593543950339	97	157	27	0.17197

Table 1: Density of pseudoTNAF for some integers u modulo $(1 - \tau)\frac{\tau^{163} - 1}{\tau - 1}$ with their sizes are 96 and 97 bits

From the above table, the average Hamming weight among integer u of length 157 is $\frac{29+28+30+29+27}{5} = 28.6$. This value is equal to $\frac{0.18471+0.17834+0.19108+0.18471+0.17197}{5} = 0.182162$ (i.e. the average density among integer u of length 157) multiplied by the length 157. With a few multiplier u that randomly chosen will not be able to give an estimation of the actual average Hamming weight of pseudoTNAF with maximum length. The question remains how such estimation can be made? This will be discussed in detail in the next topic.

6. AVERAGE HAMMING WEIGHT AMONG PseudoTNAF OF MAXIMUM LENGTH

Gordon(1998) has shown that the average Hamming weight of TNAF of all length m integers of $[1, N\ (\tau^m - 1) - 1]$ is approximately $\frac{m}{3}(1\ +\ o\ (1))$ when $m \rightarrow\ \infty$. This estimation is a product of multiplying the average density $\frac{1}{3}\ +\ o\ (1)$ with the maximum length (i.e. m). By taking the same average density, Solinas(2000) has shown that the average Hamming weight of RTNAF of all length $m + a$ integers of $[1, N(\frac{\tau^m-1}{\tau-1}) - 1]$ is about $\frac{m}{3}$ when $m \rightarrow\ \infty$. Such average is a product of multiplying the average density $\frac{1}{3}\ +\ o\ (1)$ with the maximum length (i.e. $m + a$). Now, our study provides an estimation of average density of pseudoTNAF via the following proposition. The average is estimated to be equal to $\frac{1}{3}\ +\ o\ (1)$ as Gordon(1998) with some modifications against the arguments given by him.

Proposition 2. The average density of pseudoTNAF is approximately $\frac{1}{3}\ +\ o\ (1)$.

Proof
Integer reduction of $\{0,1, \ldots, |\rho'|\ N(r' + s'\tau) - 1\}$ cover all congruence classes modulo $\rho(\frac{\tau^m-1}{\tau-1})$. Integer $|\rho'|\ N(r' + s'\tau)$ is the number of lattice points in the Voronoi region of $\rho N(\frac{\tau^m-1}{\tau-1})Z(\tau)$ as in Corollary 1. Every integers of $\{0,1, \ldots, |\rho'|\ N(r' + s'\tau) - 1\}$ are divided by $\rho(\frac{\tau^m-1}{\tau-1})$ to get a total of $|\rho'|\ N(r' + s'\tau)$ lattice points that can be obtained through the Algorithm 1. Each lattice points of length $\bar{l}$ are complete distributed to some Voronoi region of $\tau^{\bar{l}+3}\ Z\ (\tau)$ that overlaps with the Voronoi region of $\rho\left(\frac{\tau^m-1}{\tau-1}\right)$. Then, pseudoTNAF of each points obtained from the Algorithm 2. The number of pseudoTNAF of $\bar{l} > 2$ is greater than 4 and less than $N(\tau^{\bar{l}+3})$. Thus, the average density of the maximum length can be identified and it is approximately $\frac{1}{3}\ +\ o\ (1)$. ■

Finally, the average Hamming weight among pseudoTNAF of maximum length can be estimated. This is explained in the following theorem.

Theorem 2. The average Hamming weight among pseudoTNAF of all integers modulo $\rho\left(\frac{\tau^m-1}{\tau-1}\right)$ with maximum length is approximately $\left(\frac{1}{3}\ +\ o\ (1)\right)(log_2\ N(\rho) + m + a)$.

Proof
The maximum length of the pseudoTNAF expansion can be obtained from Theorem 2.7 of Yunos *et al.* (2014) i.e. $log_2\ N(\rho) + m + a$. Whereas the average density of pseudoTNAF is around $\frac{1}{3}\ +\ o\ (1)$ obtained from Proposition 2. Therefore, the average Hamming weight is $\left(\frac{1}{3}\ +\ o\ (1)\right)(log_2\ N(\rho) + m + a)$. ■

7. CONCLUSION

Proposition 1 has proved that pseudoTNAF is equivalent to TNAF in set $E_a\ (F_{2^m})$. Therefore, the pseudoTNAF can be used as a multiplier to the scalar multiplication with condition (3) so that scalar multiplication is not towards to infinity. Now, the operational costs when using pseudoTNAF can be estimated via the average Hamming weight of pseudoTNAF. That is approximately $\left(\frac{1}{3} + o\ (1)\right)(log_2\ N(\rho) + m + a)$. We also developed one algorithm for finding all pseudoTNAF in modulo $\rho\left(\frac{\tau^m - 1}{\tau - 1}\right)$. Retrieval from this algorithm is important to get the lowest operational costs of scalar multiplication for a certain ρ and m that will be the subject of our future discussion.

REFERENCES

Avanzi, R. M., Heuberger, C. and Prodinger, H. 2005. *Minimality of the Hamming Weight of the τ-NAF for Koblitz Curves and Improved Combination with Point Halving*. http://eprint.iacr.org/2005/ 225.pdf

Brumley, B.B. and Jarvinen, K. 2007. Koblitz Curves and Integer Equivalents of Frobenius Expansions. *Lecturer Notes in Computer Science*. 4876: 126-137. Springer.

Gordon, D.M. (1998). A Survey of Fast Exponentiation Methods. *Journal of Algorithms 27, Article no AL970913*, 129-146.

Hankerson, D., Menezes, A., and Vanstone, S. (2004). Guide to Elliptic Curve Cryptography. Springer-Verlag.

Hakuta, K., Sato, H. and Takagi, T. 2010. Explicit Lower bound for the Length of Minimal Weight τ-adic Expansions on Koblitz Curves. *Journal of Math-for-Industry.* 2 (2010A-7): 75-83.

Joye, M. and Tymen, C. 2001. Protection against Differential Analysis for Elliptic Curve Cryptography: An Algebraic Approach, in *Cryptography Hardware and Embedded Systems-CHES'*01, *Lecturer Notes in Computer Science.* 2162:377-390. Springer-Verlag.

Koblitz, N. 1987. Elliptic curve cryptosystem, *in Mathematics Computation.* 48 (177): 203-209.

Koblitz, N. 1992. CM curves with good cryptographic properties. *Proc. Crypto*'91: 279-287. Springer-Verlag.

Li, M., Qin, B., Kong, F. and Li, D. 2007. Wide-W-NAF Method for Scalar Multiplication on Koblitz Curves. *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/ Distributed Computing*:143-148.

Lin, T. C. 2009. Algorithm on Elliptic Curves over fields of Characteristic Two with Non-Adjacent Forms. *International Journal of Network Security.* 9(2): 117-120.

Ratsimihah, J.R. and Prodinger, H. 2005. *Redundant Representation of Numbers*. *http://resources.aims.ac.za/archive/2005/joel.ps*

Roy, S. S., Robeiro, C., Mukhopadhyay, D., Takahashi, J. and Fukunaga, T. 2011. *Scalar Multiplication on Koblitz Curves Using* τ^2 *-NAF*. http://eprint.iacr.org/2011/318.pdf

Solinas, J. A. 1997. An Improved Algorithm for Arithmetic on a Family of Elliptic Curves, in B. Kaliski, editor, *Advance in Cryptology-CRYPTO*'97. *Lecture Notes in Computer Science*. 1294: 357-371. Springer-Verlag.

Solinas, J. A. 2000. Efficient Arithmetic on Koblitz Curves, in Kluwer Academic Publishers, Boston, Manufactured in the Netherlands, *Design*, *Codes*, *and Cryptography*. 19:195-249.

Yunos, F. and Mohd Atan, K.A. 2013. An Average Density of τ-adic Naf (τ -NAF) Representation: An Alternative Proof. *Malaysian Journal of Mathematical Sciences*. 7(1): 111 -- 124.

Yunos, F., Mohd Atan, K.A., Md Said, M.R. and Kamel Ariffin, M.R . 2014. A Reduced τ -NAF (RTNAF) Representation for Scalar Multiplication on Anomalous Binary Curves (ABC). *Pertanika Journal of Science and Technology*. Accepted on 17 December 2012. Publication in JST Vol. 22(2) Jul. 2014.

Improved S-Box Construction from Binomial Power Functions*

[1, 2]Herman Isa, [1]Norziana Jamil and [2]Muhammad Reza Z'aba
[1]College of Information Technology, Universiti Tenaga Nasional (UNITEN), Selangor, Malaysia.
[2]Cryptography Lab, MIMOS Berhad, Technology Park Malaysia, Kuala Lumpur, Malaysia.
Email: herman.isa@mimos.my, Norziana@uniten.edu.my, reza.zaba@mimos.my

ABSTRACT

Substitution boxes with strong cryptographic properties are commonly used in block ciphers to provide the crucial property of nonlinearity. This is important to resist standard attacks such as linear and differential cryptanalysis. A cryptographically-strong s-box must have high nonlinearity, low differential uniformity and high algebraic degree. In this paper, we improve previous s-box construction based on binomial operation on two power functions over the finite field $\mathbb{F}_{2^8}$. By widening the scope of the power function and introducing new manipulation techniques, we managed to obtain cryptographically-strong s-boxes which are better than the previous construction.

Keyword: s-box construction, binomial power functions, nonlinearity, bijective, substitution boxes.

1. INTRODUCTION

In his seminal work in 1949, Shannon defined the property of confusion which should exist in an encryption system (Shannon, 1949). Basically confusion is required so that the ciphertext is related to both the plaintext and secret key, in a complex way. In modern block ciphers, this property can be provided by a component called a substitution box (s-box). Since an s-box plays an important role in a block cipher, it must be cryptographically strong to resist various attacks such as differential (Biham and Shamir, 1991) and linear cryptanalysis (Matsui, 1994). A cryptographically strong s-box should have high nonlinearity (NL), low differential uniformity (DU) and high algebraic degree (AD).

Generally, the construction of an s-box can be categorized into three generic methods which are random search, evolutionary or heuristic method and lastly mathematical function approaches. In (Isa, Jamil, and Z'aba, 2013) the authors use the combination of mathematical function approach and heuristic method in their proposed s-boxes construction. In detail, they proposed the construction of an s-box using binomial operation on a non-permutation power function with another power function in the finite field $\mathbb{F}_{2^8}$. The resulting function's codomain is analyzed to determine elements which are mapped by more than one input in its domain. These are referred to as *redundant* elements. If these elements exist, then the function is further manipulated using a heuristic method. The final s-box is produced if it exhibits strong cryptographic properties. They obtained an s-box which has a NL of 106, DU of 6 and AD of 7. We denote this as the tuple (106, 6, 7). Furthermore, the s-box is ranked sixth out of 20 where s-boxes are sorted according to their NL, then DU and AD. The best known s-box (e.g. AES (Daemen and Rijmen, 2002)) has a property of (112, 4, 7).

Inspired by the uniqueness of cryptographic properties exhibits from the binomial power functions, we improve Isa *et al.*'s construction (Isa, Jamil, and Z'aba, 2013) by widening the scope of the power functions over the finite field $\mathbb{F}_{2^8}$ to include both permutation and non-permutation. Furthermore, in analyzing the redundant elements, we introduce two methods which are addition and multiplication. Using these approaches, we obtained three different s-boxes which have the cryptographic properties of (108, 4, 7), (108, 6, 4) and (106, 6, 7) respectively. One of these s-boxes is better than the one proposed by Isa, Jamil, and Z'aba (2013).

The rest of the paper is organized as follows. In the second section, the main cryptographic properties of an s-box are discussed. In the third section, we present and discuss our s-box construction and its findings. The paper is concluded in the last section.

*This research project is supported by Ministry of Higher Education (MOHE), Malaysia

2. S-BOX PROPERTIES

An s-box needs to have at least three strong cryptographic properties which are high nonlinearity (NL), low differential uniformity (DU) and high algebraic degree (AD). In this paper, our focus is bijective s-boxes over the finite field $\mathbb{F}_{2^8}$.

Let $\mathbb{F}_2$ and $\mathbb{F}_{2^n}$ be a finite field with 2 and 2^n elements, respectively. An $n \times n$ s-box is a Boolean map:

$$F: \mathbb{F}_{2^n} \to \mathbb{F}_{2^n} = \big(f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_n(x_1, \dots, x_n)\big)$$

Nonlinearity. Let $c = (c_1, c_2, \dots, c_n)$ be a nonzero elements in $\mathbb{F}_{2^n}$. Let$c \cdot F = c_1 f_1 + c_2 f_2 + \cdots + c_n f_n$ be a linear combination of the coordinate Boolean functions $f_1, f_2, \dots, f_n$ of F. The nonlinearity (NL) for an s-box is defined as:

$$\mathrm{NL}(F) = \min_{c \in \mathbb{F}_{2^n}, c \neq 0} \mathrm{NL}(c \cdot F)$$

The NL of F is the Hamming distance between the set of all non-constant linear combinations of component functions of F and the set of all affine functions over $\mathbb{F}_{2^n}$. The known highest NL value is 112 as obtained by AES’s s-box (Daemen and Rijmen, 2002) and Li *et al.*’s proposed s-box (Li and Wang, 2012). As suggested by (Piret, Roche and Carlet, 2012), the NL must be close to the best known nonlinearity (i.e. NL of AES s-box) to thwart linear cryptanalysis (Matsui, 1994). Therefore in this study, we set the value of NL > 100 for the s-box to be considered as cryptographically strong.

Differential Uniformity. The Differential Uniformity (DU) of an s-box is the largest value present in its difference distribution table by omitting the trivial entry case, $a = b = 0$. The DU is defined as:

$$\mathrm{DU}(F) = \max_{a,b \in \mathbb{F}_{2^n}, a \neq 0} |\{x \in \mathbb{F}_{2^n}: F(x+a) + F(x) = b\}|$$

Better s-box has smaller value (i.e. $2 \leq \mathrm{DU} \leq 6$) as preferred in (Piret, Roche, and Carlet, 2012) to resist against differential cryptanalysis (Biham and Shamir, 1991).

Algebraic Degree. The Algebraic Degree (AD) of an s-box can be determined by the maximum degree between all component functions:

$$\mathrm{AD}(F) = \max\{\deg(f_1), \deg(f_2), \dots, \deg(f_n)\}$$

where $\deg(f)$ is the number of variables in the largest monomial of an s-box. Preferable measurement of AD ≥ 4 is suggested in (Piret, Roche, and Carlet, 2012) in order to resist higher order differential cryptanalysis (Knudsen, 1995).

3. S-BOX CONSTRUCTION AND FINDINGS

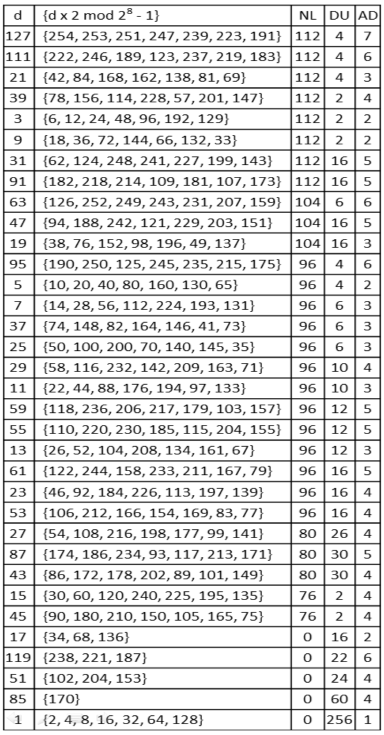
In the works of (Isa, Jamil and Z’aba, 2013) and (Mamadolimov, Isa and Mohamad, 2013), the authors proposed a construction of an s-box using binomial power function approach. However, they only focus on non-permutation power functions that carry high cryptographic properties as one of the two seed functions. In this study, we do thorough analysis on all power functions (permutation and non-permutation) over the finite field $\mathbb{F}_{2^8}$. We study the cryptographic properties exhibited from the binomial operation on the two power functions. If the resulting function is shown to be non-bijective, then additional operations are performed which are:

1) *Addition* with another power function, and

2) *Multiplication* of the second function with a coefficient.

Let x^d denotes a power function in $\mathbb{F}_{2^8}$ with the irreducible polynomial $x^8 + x^4 + x^3 + x^2 + 1$, where $d = \{1, 2, \dots, 2^8 - 2\}$ and $x \in \mathbb{F}_{2^8}$. All these functions can be classified into linearly non-equivalent functions using the squaring method (Aslan, Sakalli and Bulus, 2008) as shown in Table 1.

The first column of Table 1 represents the powers d that are non-equivalent to each other. The second column lists all the equivalent power functions for each of power d. For instance, the power x^{127} is equivalent to x^{223}. Other columns give the values of nonlinearity (NL), differential uniformity (DU) and algebraic degree (AD) of the s-box produced using the underlying power function.



d	{d x 2 mod 2^8 - 1}	NL	DU	AD
127	{254, 253, 251, 247, 239, 223, 191}	112	4	7
111	{222, 246, 189, 123, 237, 219, 183}	112	4	6
21	{42, 84, 168, 162, 138, 81, 69}	112	4	3
39	{78, 156, 114, 228, 57, 201, 147}	112	2	4
3	{6, 12, 24, 48, 96, 192, 129}	112	2	2
9	{18, 36, 72, 144, 66, 132, 33}	112	2	2
31	{62, 124, 248, 241, 227, 199, 143}	112	16	5
91	{182, 218, 214, 109, 181, 107, 173}	112	16	5
63	{126, 252, 249, 243, 231, 207, 159}	104	6	6
47	{94, 188, 242, 121, 229, 203, 151}	104	16	5
19	{38, 76, 152, 98, 196, 49, 137}	104	16	3
95	{190, 250, 125, 245, 235, 215, 175}	96	4	6
5	{10, 20, 40, 80, 160, 130, 65}	96	4	2
7	{14, 28, 56, 112, 224, 193, 131}	96	6	3
37	{74, 148, 82, 164, 146, 41, 73}	96	6	3
25	{50, 100, 200, 70, 140, 145, 35}	96	6	3
29	{58, 116, 232, 142, 209, 163, 71}	96	10	4
11	{22, 44, 88, 176, 194, 97, 133}	96	10	3
59	{118, 236, 206, 217, 179, 103, 157}	96	12	5
55	{110, 220, 230, 185, 115, 204, 155}	96	12	5
13	{26, 52, 104, 208, 134, 161, 67}	96	12	3
61	{122, 244, 158, 233, 211, 167, 79}	96	16	5
23	{46, 92, 184, 226, 113, 197, 139}	96	16	4
53	{106, 212, 166, 154, 169, 83, 77}	96	16	4
27	{54, 108, 216, 198, 177, 99, 141}	80	26	4
87	{174, 186, 234, 93, 117, 213, 171}	80	30	5
43	{86, 172, 178, 202, 89, 101, 149}	80	30	4
15	{30, 60, 120, 240, 225, 195, 135}	76	2	4
45	{90, 180, 210, 150, 105, 165, 75}	76	2	4
17	{34, 68, 136}	0	16	2
119	{238, 221, 187}	0	22	6
51	{102, 204, 153}	0	24	4
85	{170}	0	60	4
1	{2, 4, 8, 16, 32, 64, 128}	0	256	1

Table 1: Classification of power function, x^d based on maximum nonlinearity in $\mathbb{F}_{2^8}$.

The construction starts by adding two different power functions F_1 and F_2 to produce F:

$$F = F_1 + F_2.$$

Then, we count how many elements in the resulting function's codomain which are mapped by more than one input in its domain. We denote this number as $\mathrm{R_{EL}}$ and refer these elements as *redundant* elements. If $\mathrm{R_{EL}}$ is greater than zero, then the function F will be sent to an algorithm called *Algo 2*. Otherwise, the cryptographic properties generated by the function are recorded and we move to the next power function. Our construction is illustrated in Figure 1.

The results of performing binomial operation on all power functions are summarized in Table 2. $\mathrm{N_{EL}}$ refers to the number of elements that does not exist in the codomain of the resulting binomial function. We refer to these elements as *non-existent* elements. There are a total of $\mathrm{C}_2^{254} = 32131$ possible combinations of binomial power functions. These functions can be categorized into 130 categories based on the combination of ($\mathrm{N_{EL}}$, $\mathrm{R_{EL}}$) for each generated binomial function. The FREQ column denotes the number of binomial function in that particular category. As an example, there are 1024 binomial functions that have 15 non-existent elements and one redundant element (i.e. one element of the function's codomain is mapped by more than one input in its domain).

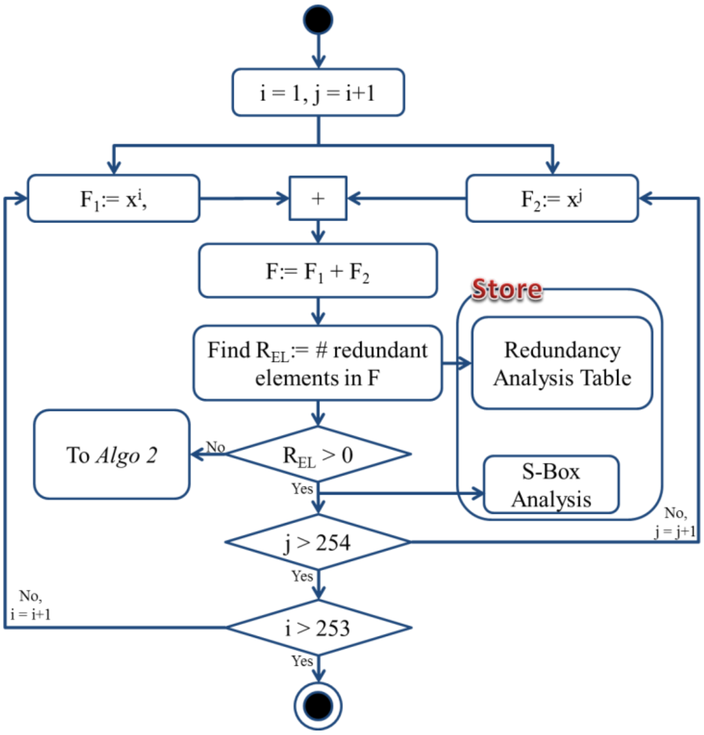


Figure 1: Our Proposed S-box Construction

As mentioned earlier, if the generated binomial function gives $R_{EL} > 0$, then the output will be sent to *Algo 2* for further analysis. *Algo 2* consists of two methods to manipulate the output so that a nearly bijective function is obtained. The methods are 1) *Addition* with another power function, and 2) *Multiplication*, where the second power function is multiplied with a coefficient. This is illustrated in Figure 2.

Note that before we perform the *Addition* or *Multiplication* methods in *Algo 2*, we first perform equivalence check on the involved functions. This equivalence check is intended to ensure that not all involved power functions in each method is from linearly equivalent power function. If this happens, then the output of the generated functions will likely to have the same cryptographic properties as in Table 1.

N_{EL}	R_{EL}	FREQ	N_{EL}	R_{EL}	FREQ	N_{EL}	R_{EL}	FREQ	N_{EL}	R_{EL}	FREQ	N_{EL}	R_{EL}	FREQ	N_{EL}	R_{EL}	FREQ
15	1	1024	64	22	256	206	50	128	88	64	384	84	72	128	105	83	256
17	1	128	51	25	512	85	51	256	96	64	256	92	72	256	97	85	256
51	1	256	230	26	16	125	51	128	192	64	128	75	73	640	105	85	128
85	1	128	69	31	512	205	51	224	89	65	640	93	73	256	171	85	256
16	2	128	85	35	256	72	52	256	191	65	128	97	73	256	90	86	128
254	2	1	221	35	64	84	52	640	94	66	256	99	73	640	102	86	128
253	3	2	75	37	256	204	52	64	85	67	256	123	73	256	120	86	128
252	4	4	217	39	64	77	53	256	93	67	256	183	73	320	170	86	64
251	5	4	45	41	128	85	55	256	97	67	256	92	74	256	96	88	256
40	6	128	81	41	512	80	56	256	189	67	64	93	75	128	104	88	256
250	6	4	85	41	256	90	58	512	84	68	384	181	75	320	102	90	256
248	8	12	214	42	64	89	59	1024	92	68	768	84	76	256	160	96	64
247	9	8	85	43	512	88	60	256	77	69	128	108	76	256	99	97	128
246	10	8	213	43	256	92	60	384	85	69	384	179	77	1024	100	98	128
245	11	32	50	46	128	75	61	256	97	69	128	178	78	64	108	100	128
244	12	8	82	46	256	99	61	768	187	69	192	177	79	128	144	112	64
243	13	32	90	46	512	195	61	64	78	70	128	96	80	256	113	113	128
80	16	128	210	46	64	84	62	128	92	70	768	100	80	256	120	120	128
240	16	96	209	47	64	85	63	256	94	70	128	85	81	896	136	120	192
239	17	112	75	49	128	89	63	768	96	70	128	125	81	256	128	128	576
68	18	128	99	49	256	193	63	64	93	71	256	175	81	512			
238	18	16	207	49	256	84	64	128	185	71	64	104	82	256			

Table 2: Redundancy Analysis Table

For *Multiplication* method, we only examine the multiplication of coefficient on the second power function while for *Addition* method, the coefficients of all involved power functions is set to 1. The purpose of this technique is to study the degree of generated output likelihood towards bijective function in addition to measuring the strength of the exhibited cryptographic properties.

Both methods (i.e. *Addition* and *Multiplication*) will perform equivalence check on the given binomial function, $F = x^i + x^j$. An additional equivalence check will be performed in *Addition* method which is between F and new power function, $x^k, k \neq \{i, j\}$. If all equivalence checks give linearly equivalent function, then the process is discarded. Otherwise, the process continues with either addition with another power function, (i.e. $F = x^i + x^j + x^k, i \neq j \neq k$) or multiplication of the second power function with a coefficient, (i.e. $F = x^i + \alpha x^j, \alpha \in \{1,2,\dots,2^8\}$). If no redundant elements found in F (i.e. $R_{EL} = 0$), the s-box properties will be measured on that output. Then, the output will be stored as a new s-box if the desired value is achieved.

Using this method, we obtained three cryptographically strong s-boxes. Two of them generated from the *Addition* method and the other one from the *Multiplication* method. These s-boxes are shown in Tables 3, 4 and 5. The first column in Tables 3, 4 and 5 denotes the first four bits of the input while the first row in each table denotes the remaining four bits of the 8-bit input to the s-box. For example, in Table 3, the input 63 gives the output A6. i.e. F(63) = A6.

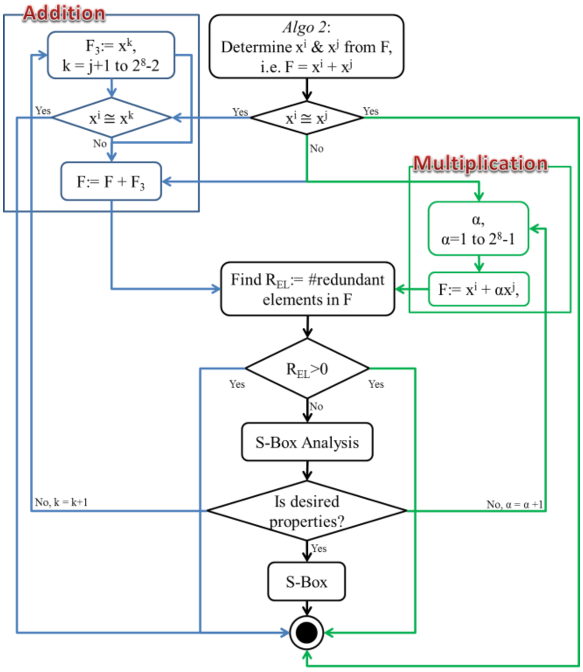


Figure 2: *Algo 2*

Table 3 gives us the first s-box, S-Box1, generated from *Addition* method, with function $F_{S-Box1} = x^{19} + x^{70} + x^{223}$. This function exhibits (108, 4, 7) for its (NL, DU, AD).

Table 4 also is an s-box generated from *Addition* method but with different function, $F_{S-Box2} = x^{29} + x^{89} + x^{164}$. We denote it as S-Box2. Its s-box properties are (108, 6, 4) for its (NL, DU, AD) respectively.

Another proposed s-box denoted as S-Box3 is shown in Table 5. This was generated using the *Multiplication* method with function $F_{S-Box3} = x^{21} + 211x^{191}$. Its s-box properties of (NL, DU, AD) are (106, 6, 7).

Empirically, all three proposed s-box functions (i.e. F_{S-Box1}, F_{S-Box2} and F_{S-Box3}) were identified based on smallest combination of (N_{EL}, R_{EL}) from Table 2. As an example, the binomial operation of any two elements in F_{S-Box1} will give us the combination of (51, 1), (i.e. $(x^{19} + x^{70})$ or $(x^{70} + x^{223})$ or

$(x^{19} + x^{223})$ will generated a function with (51, 1) for its (N_{EL}, R_{EL}) combination). The F_{S-Box2} is identified from (15, 1) combination while F_{S-Box3} is generated from the combination of (85, 1) of its (N_{EL}, R_{EL}) pair.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	00	01	0D	AC	51	37	2F	68	90	65	4E	44	F6	75	D4	55
1	D1	7C	40	D9	21	F9	25	58	82	CD	59	4A	F4	C2	8E	EA
2	E2	B0	56	77	64	10	60	9D	D5	B7	A1	F7	24	04	FF	76
3	66	A2	38	FD	D2	6F	48	1C	F0	53	02	2C	88	7E	B2	4B
4	0E	A8	E3	83	99	DD	DE	B8	EE	47	D0	B3	DA	A7	45	93
5	A3	B5	DB	F1	D3	C1	3F	34	89	91	41	6D	8F	17	A0	06
6	B9	5F	69	A6	A9	1D	95	05	63	AD	6C	AA	E6	29	8C	12
7	C4	B1	84	9F	AE	7A	E1	8A	EC	C7	22	C5	4D	8D	18	94
8	46	08	1B	2D	67	14	31	BF	7F	C3	28	BB	D8	FE	61	C0
9	52	0F	98	0A	CC	E8	1A	9A	0B	DC	50	F2	72	87	09	32
A	03	5B	F5	30	CE	7B	C9	5E	4C	26	A4	8B	11	1F	6A	3D
B	AB	57	13	5A	33	E0	49	81	85	3B	F3	9B	73	F8	EF	A5
C	C6	DF	FC	BD	3C	79	15	ED	16	AF	0C	FB	27	54	39	E5
D	62	6B	6E	07	86	B6	D6	D7	3A	97	E7	2B	92	4F	CA	96
E	70	CF	B4	3E	FA	CB	EB	71	35	BE	78	2E	19	43	E4	20
F	5D	1E	9E	80	9C	42	2A	7D	36	BC	5C	C8	23	E9	BA	74

Table 3: S-Box1 from *Addition* Method

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	00	01	17	6C	08	E9	35	CB	39	4A	DD	98	4B	E4	F1	D1
1	40	EB	A5	E0	78	9B	9A	3C	AA	76	14	F2	EF	CD	0E	61
2	43	EC	32	1F	6A	12	BE	80	A6	9D	7B	8E	B4	81	8C	6F
3	E1	47	B5	57	73	5F	37	55	4C	F5	5D	74	7D	1D	10	36
4	28	89	2D	DE	92	4E	E5	D3	B8	54	3B	15	8F	FF	45	93
5	88	BA	24	50	B6	C2	F8	BC	B3	0D	58	A8	B0	23	AE	D9
6	2E	13	41	77	FD	19	67	91	D8	AB	5A	B2	C5	6B	BD	4D
7	6D	48	1A	72	C8	FA	42	EA	85	A1	9F	FE	C1	7C	05	F6
8	7A	71	3F	CA	33	82	C7	29	0C	2C	63	69	C0	3A	D4	79
9	7F	51	44	4F	90	27	06	FB	0B	DC	B9	07	E2	46	1B	65
A	59	A0	A4	09	E8	8D	A7	96	5C	E6	95	AF	0F	AC	8A	75
B	C9	EE	03	F9	9E	F3	62	1C	18	20	04	AD	B1	CC	49	B7
C	6E	DA	3D	D5	F7	2A	26	DB	97	25	60	86	52	34	A2	30
D	53	3E	C3	A9	64	D0	D7	D6	C6	BB	38	D2	99	0A	ED	87
E	5B	E3	CE	21	02	66	1E	F0	FC	CF	E7	8B	9C	2B	BF	56
F	11	A3	68	84	DF	F4	16	70	22	83	5E	31	7E	94	2F	C4

Table 4: S-Box2 from *Addition* Method

Table 6 summarize and compares our obtained s-boxes with the existing 8×8 cryptographically strong s-boxes in literature. As we mention in the earlier section, to be considered as cryptographically strong s-boxes, the following cryptographic properties condition must be satisfied: i) $\text{NL} > 100$, ii) $2 \leq \text{DU} \leq 6$ and iii) $\text{AD} \geq 4$.

As a result, there are a total of 21 proposed s-boxes with several different techniques that include multiplicative inverse in $\mathbb{F}_{2^8}$, conversion function from $\mathbb{F}_{2^9}$ to $\mathbb{F}_{2^8}$, gray s-box, linear fractional transformation, theorem of polynomial permutation, bit permutation, 4-step tweaking and manipulation of power functions in $\mathbb{F}_{2^8}$ as a based function.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	00	D2	58	EB	17	ED	C5	9E	EC	69	67	C7	F8	F5	5F	0D
1	8C	03	38	66	4A	EF	40	2B	77	19	12	64	A6	01	C0	91
2	82	49	FC	50	32	05	93	B0	96	B1	2A	0F	8E	62	4D	F3
3	6E	C8	E7	BF	AA	1F	A5	29	1C	42	1B	30	F1	04	B7	4F
4	37	2D	F0	1A	D1	21	23	AB	76	56	F7	75	9F	A7	E5	4C
5	F6	47	87	78	60	A4	99	E0	31	DA	BD	FB	27	7D	E6	4E
6	4B	D7	6B	CA	89	D4	72	44	35	63	6A	94	7C	51	84	48
7	CC	39	0C	BB	FF	E9	B2	88	7E	DD	95	80	D6	DE	8F	22
8	81	EA	E8	14	8A	24	43	0B	FE	CE	5C	41	7A	A8	61	28
9	3A	A2	79	25	3B	10	13	5A	57	2C	B6	DC	BA	36	73	3F
A	11	68	15	A3	E4	59	BC	AE	C9	E1	1E	6D	6C	D8	2E	9B
B	2F	D0	6F	FD	AD	E3	90	DF	06	EE	CB	C1	46	F9	DB	9A

C	34	16	AF	20	3C	5B	C3	52	02	B9	09	07	D3	5D	E2	33
D	9D	92	0A	D5	C4	70	AC	7F	08	9C	D9	53	C2	1D	3E	C6
E	26	B3	B5	F2	B4	0E	74	F4	A1	B8	7B	A0	86	45	CF	CD
F	71	54	8D	A9	65	BE	55	83	3D	97	FA	85	98	8B	18	5E

Table 5: S-Box3 from *Multiplication* Method

The most used technique in the early construction of an s-box is using multiplicative inverse in $\mathbb{F}_{2^8}$ (Daemen and Rijmen, 2002; Aoki *et al.*, 2001; Kwon *et al.*, 2004; Hirata, 2010; Ohkuma, 2001) and (Shirai *et al.*, 2007)). This technique gives the best known cryptographic properties for an s-box and ranked first in Table 6. There are also proposed s-boxes by (Tran, Bui and Duong, 2008) and (Hussain *et al.*, 2013) that give the same s-box properties as the best known s-box.

Our proposed s-box is ranked sixth, seventh and eighth after the proposed s-box by (Li and Wang, 2012) and three different s-boxes by (Tran, Bui and Duong, 2008). Two of our proposed s-boxes are better than the s-boxes proposed by (Isa, Jamil and Z'aba, 2013) which were ranked eighth and ninth. There is also the proposed s-box by Fuller (Fuller and Millan, 2003) at rank number 8 and Mamadolimov's s-box (Mamadolimov, Isa and Mohamad, 2013) at rank number 10.

4. CONCLUSION

In this paper, we manage to improve the s-box construction based on binomial operation on power functions proposed by (Isa, Jamil and Z'aba, 2013). By widening the scope of the power function and introducing new manipulation techniques, we managed to obtain a stronger s-box than the previous construction. All the s-boxes are the results of manipulating binomial power functions which have one redundant element. Two of these s-boxes are produced using the addition method and the other one using the multiplication method.

Rank	S-Box	NL	DU	AD	*Techniques*
1	AES (Daemen and Rijmen, 2002)	112	4	7	Multiplicative Inverse, x^{-1} in $\mathbb{F}_{2^8}$
	Camellia (Aoki *et al.*, 2001)				
	ARIA (Kwon *et al.*, 2004)				
	HyRAL (Hirata, 2010)				
	Hierocrypt-HL (Ohkuma, 2001)				
	CLEFIA-S$_1$ (Shirai *et al.*, 2007)				
	Tran *et al.* (Tran, Bui and Duong, 2008)				Gray S-Box
	Hussain *et al.* (2013)				Linear Fractional Transformation
2	Li and Wang, (2012)	112	4	5	Conversion $\mathbb{F}_{2^9} \rightarrow \mathbb{F}_{2^8}$
3	Yang *et al.* (2011)	112	6	7	Theorem of Permutation Polynomials
4		110	4	7	
5		110	6	7	
6	**S-Box1**	**108**	**4**	**7**	**Trinomial Power Functions (*Addition*)**
7	**S-Box2**	**108**	**6**	**4**	
8	**S-Box3**	106	6	7	**Binomial Power Function (*Multiplication*)**
	Hierocrypt-LL (Ohkuma, 2001)				Bit Permutation from x^{-1} in $\mathbb{F}_{2^8}$
	Fuller *et al.* (Fuller and Millan, 2003)				4-Step tweaking on AES s-box
	Isa, Jamil and Z'aba (2013)				Binomial Power Function
9	Isa, Jamil and Z'aba (2013)	104	6	7	Binomial Power Function + Heuristic Techniques

10	Mamadolimov, Isa and Mohamad (2013)	102	8	7	Binomial Power Functions

Table 6: Comparison of Cryptographically Strong S-Boxes

5. ACKNOWLEDGMENT

The authors would like to thank you Ministry of Higher Education (MOHE), Malaysia for supporting and financing this research project under the Exploratory Scheme Research Grant 2012/13.

REFERENCES

Aoki, K., Ichikawa, T., Kanda, M., Matsui, M., Moriai, S., Nakajima, J., and Tokita, T. 2001. Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms - Design and Analysis. *Selected Areas in Cryptography,* 39-56.

Aslan, B., Sakalli, M. T. and Bulus, E. 2008. Classifying 8-bit to 8-bit S-Boxes based on Power Mappings from the Point of DDT and LAT Distributions, *Arithmetic of Finite Fields*, 123-133.

Biham, E. and Shamir, A. 1991. Differential Cryptanalysis of DES-Like Cryptosystems, *Advances in Cryptology (CRYPTO '90)*, 2-21.

Daemen, J. and Rijmen, V. 2002. The Design of Rijndael, AES – The Advanced Encryption Standard.

Fuller, J. and Millan, W. 2003. Linear Redundancy in S-Boxes, *Fast Software Encryption*, 74-86.

Hirata, K. 2010. The 128-bit Block Cipher HyRAL (Hybrid Randomization Algorithm): Common Key Block Cipher. *International Symposium on Intelligence Information Processing and Trusted Computing*, 9-14.

Hussain, I., Shah, T., Gondal, M.A., and Mahmood , H. 2013. An Efficient Approach for the Construction of LFT S-boxes using Chaotic Logistic Map, *Nonlinear Dynamics*, 133-140.

Isa, H., Jamil, N. and Z'aba, M. R. 2013. S-Box Construction from Non-Permutation Power Functions, *Proceeding of the 6th International Conference on Security of Information and Networks (SIN '13)*, 46-53.

Knudsen, L. R. 1995. Truncated and Higher Order Differentials, *Fast Software Encryption*, 196-211.

Kwon, D., Kim, J., Park, S., Sung, S., Sohn, Y., Song, J., Yeom, Y., Yoon, E.-J., Lee, S., Lee, J., Chee, S., Han, D., and Hong, J. 2004. New Block Cipher: ARIA. *Information Security and Cryptology*, 432 - 445.

Li, Y. and Wang, M. 2012. Constructing Differentially 4-uniform Permutations over GF(2^{2m}) from Quadratic APN Permutations over GF(2^{2m+1}), *Designs, Codes and Cryptography*, 1-16.

Mamadolimov, A., Isa, H. and Mohamad, M. S. 2013. Practical Bijective S-Box Design. Accessed 21st Jan 2013. Sourced from http://arxiv.org/abs/1301.4723

Matsui, M. 1994. Linear Cryptanalysis Method for DES Cipher, *Advances in Cryptology (EUROCRYPT '93)*, 386-397.

Ohkuma, K., Muratani, H., Sano, F., and Kawamura, S. 2001. The Block Cipher Hierocrypt, *Selected Areas in Cryptography*, 72-88.

Piret, G., Roche, T. and Carlet, C. 2012. PICARO - A Block Cipher Allowing Efficient Higher-Order Side-Channel Resistance, *Applied Cryptography and Network Security*, 311-328.

Shannon, C. E. 1949. Communication Theory of Secrecy Systems, *Bell System Technical Journal 28*, vol 7, 656-715.

Shirai, T., Shibutani, K., Akishita, T., Moriai, S., and Iwata, T. 2007. The 128-bit Blockcipher CLEFIA. *Fast Software Encryption*, 181-195.

Tran, M. T., Bui, D. K. and Duong, A. D. 2008. Gray S-Box for Advanced Encryption Standard, *Computational Intelligence and Security, CIS* '08, 253-258.

Yang, M., Wang, Z., Meng, Q., and Han, L. 2011. Evolutionary Design of S-box with Cryptographic Properties, *Parallel and Distributed Processing with Applications Workshops (ISPAW)*, 12-15.

Analysis on Lightweight Block Cipher, SIMON

[1]Isma Norshahila Mohammad Shah, [2]Liyana Chew Nizam Chew, [3]Nor Azeala Mohd Yusof, [4]Nik Azura Nik Abdullah, [5]Norul Hidayah Lot@Ahmad Zawawi and [6]Hazlin Abdul Rani

Cyber Technology Research Department,
CyberSecurity Malaysia,
Kuala Lumpur, Malaysia
Email: [1]isma@cybersecurity.my, [2]liyana@cybersecurity.my, [3]azeala@cybersecurity.my, [4]azura@cybersecurity.my, [5]norul@cybersecurity.my, [6]hazlin@cybersecurity.my

ABSTRACT

In this paper, we present a randomness analysis on the members of the lightweight block cipher, Simon. Simon and Speck are two families of block ciphers proposed by the National Security Agency on June 2013. It consist varieties of block and key size. Block size starts from 32-bit to 128-bit while key size starts from 64-bit to 256-bit. In this paper, analysis was performed on Simon32/64, Simon48/72, Simon64/96, Simon96/96 and Simon128/128 by using NIST Statistical Test Suite. The algorithms are tested on the output sequence generated using nine data categories. From the analysis conducted, we can conclude that the output from the algorithm tested are not random based on 0.1% significance level.

Keywords: lightweight block cipher, Statistical analysis, Simon block cipher, significance level, NIST Statistical Test Suite

1. INTRODUCTION

Lightweight cryptography is being used in response to typical constraints in the hardware used in sensitive application such as RFID systems, sensor networks, the Internet of Things and many more. The hardware used in these applications will likely be constrained in computational power, battery, as well as memory. Lightweight cryptography is tailored for such constrained devices, with the goal of balancing the trade-offs between low resource requirements, performance, and cryptographic strength.

Recently, National Security Agency (NSA) has released two new families of lightweight block cipher, SIMON and SPECK on June 2013 (Ray *et al.*, 2013). It comes in a variety of widths and key sizes. Both families can perform well in hardware and software but SIMON family is optimized for hardware platforms while Speck is optimized for software platform.

SIMON block cipher operates in classical Feistel scheme with *n* bit block will be referred as SIMON2*n* where *n* is required to be 16, 24, 32, 48 and 64. For SIMON with 48 bit block and 72 bit key will be represented as Simon48/72. SIMON family consists of SIMON32/64, SIMON48/72, SIMON48/96, SIMON64/128, SIMON96/96, SIMON96/144, SIMON128/128, SIMON128/192 and SIMON128/192.

One of the important criteria when evaluating an algorithm is to demonstrate its suitability as a random number generator. Statistical analysis can be used to determine whether the algorithm can fulfill this criterion. One of the techniques used to evaluate the randomness properties of a finite sequence is by using the NIST Statistical Test Suite (Andrew *et al.*,2010). For the evaluation of block ciphers presented for AES competition in 1997, Soto (2000) proposed nine different ways to generate large sequences of data from block ciphers and tested these streams using the NIST Statistical Test Suite.

This paper will illustrate the statistical analysis conducted on selected size of algorithms in SIMON family of lightweight block cipher. The analysis reported here focused on SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128. Future works will address the remaining size of the algorithm. Section II gives a brief description of SIMON algorithm in general. Section III and IV discussed further on the randomness test and data description, respectively. Section V explained on the analysis framework while Section VI discussed on the result of the experiment. Conclusion is demonstrated in Section VII.

2. BRIEF DESCRIPTION OF SIMON

SIMON algorithm, which was designed by NSA, aims to fill the need for secure, flexible, and analyzable lightweight block ciphers. SIMON was designed to be flexible on a variety of platforms but it is tuned for optimal performances in hardware (Ray B. *et al*, 2013). SIMON supports block sizes of 32, 48, 64, 96 and 128 bits, with up to three key sizes to go along with each block size.

A. Round Function

SIMON has a classical Feistel structure as shown in Figure 1. In general, each round of SIMON uses three operations on its *n*-bit words; bitwise XOR, ⊕, bitwise AND, & and left circular shift, S^j by *j* bits. It includes a non-linear and non-invertible function, F. Given X ϵ $\{0, 1\}^n$, F(X) is calculated as follows:

$$F(X) = (X \lll 2) \oplus ((X \lll 1) \,\&\, (X \lll 8))$$

SIMON operates on two *n*-bit halves in each round. The output of F is added using XOR to the right half along with a round key, and then the two halves are swapped.

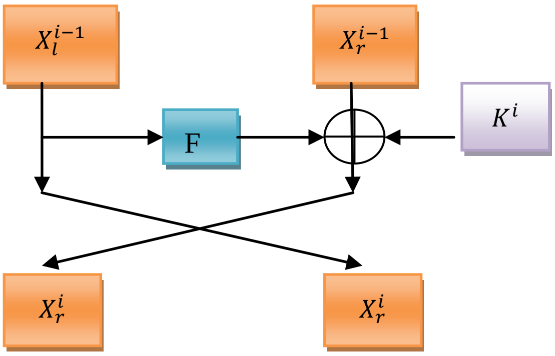


Figure 1: Round Function of SIMON

B. Key Schedule

Depending on the size of the algorithm, the key sizes vary. It was operating on two, three and four *n*-bit words registers as shown in Table 1. It performed two rotations to the right by x >>> 3 and x >>>1 and XOR the results together with a fixed constants, *c* and five constant sequences, z_j which are dependent. Table 2 shows the value of z_j. Assuming that the number of word for key, K is m, the first subkeys are driven from K, i.e. $K^0 \ldots K^{m-1}$. The subkey for round i where $m \le i \le r\text{-}1$, is calculated as follows:

$$m=2;\ K^i = K^{i-2} \oplus (K^{i-1} >>> 3) \oplus (K^{i-1} >>> 4) \oplus \mathrm{c} \oplus (\mathrm{z_j})_{\mathrm{i-m}}$$
$$m=3;\ K^i = K^{i-3} \oplus (K^{i-1} >>> 3) \oplus (\mathrm{K}^{\mathrm{i-1}} >>> 4) \oplus \mathrm{c} \oplus (\mathrm{z_j})_{\mathrm{i-m}}$$
$$m=4;\ K^i = K^{i-4} \oplus K^{i-3} \oplus (K^{i-1} >>> 3) \oplus ((\mathrm{K}^{\mathrm{i-3}} \oplus (\mathrm{K}^{\mathrm{i-1}} >>> 3)) >>> 1) \oplus \mathrm{c} \oplus (\mathrm{z_j})_{\mathrm{i-m}}$$

where $c = 2^{n-1} \oplus 3$ is a constant value, $(z_j)_{i-m}$ denotes the i^{th} bit of z_j where $m \le i \le r\text{-}1$. The value of $i\text{-}m$ is modulo by 62. z_j are five constant values as shown in Table 2 while j is the parameter for the constant, z as listed in Table 1.

Algorithm	**Block Size**	**Key Size**	**Number of Rounds**	**Input size of one word, *n***	**Number of word for key, *m***	***j***
SIMON32/64	32	64	32	16	4	0
SIMON48/72	48	72	36	24	3	0
SIMON48/96	48	96	36	24	4	1
SIMON64/96	64	96	42	32	3	2
SIMON64/128	64	128	44	32	4	3
SIMON96/96	96	96	52	48	2	2
SIMON96/144	96	144	54	48	3	3
SIMON128/128	128	128	68	64	2	2
SIMON128/192	128	192	69	64	3	3
SIMON128/256	128	256	72	64	4	4

Table 1: Details of variants in SIMON, *j* is the parameter for the constant, *z*.

j	z_j
0	11111010001001010110000111001101111101000100101011000011100110
1	10001110111110010011000010110101000111011111001001100001011010
2	10101111011100000011010010011000101000010001111110010110110011
3	11011011101011000110010111100000010010001010011100110100001111
4	11010001111001101011011000100000010111000011001010010011101111

Table 2: Value of Constant Sequences, z_j used in SIMON Key Schedule

3. RANDOMNESS TEST

Randomness test was conducted using the NIST Statistical Test Suite (StsGui MFC Application). The tool was developed by the National Institute of Standards and Technology, USA (NIST) to evaluate the statements that the stream is generated by a truly random source. The NIST Statistical Test Suite (Juan Soto *et al.* (2000)) consists of 15 tests. Descriptions of each test are explained in Appendix A. These 15 tests are divided into two types of categories which are Parameterized Test Selection and Non-Parameterized Test Selection. Users need to identify parameter value for each test in Parameterized Test Selection. The lists of tests for Parameterized Test Selection are Block Frequency Test, Overlapping Template Test, Non-overlapping Templates Test, Serial Test, Approximate Entropy Test, Linear Complexity Test and Maurer's Universal Test. Furthermore, the lists of tests for Non-Parameterized Test Selection are Cumulative Sums (Forward/Reverse) Test, Runs Test, Longest Runs of Ones Test, Binary Matrix Rank Test, Spectral Test, Random Excursion Test, Random Excursion Variants Test and Frequency Test.

For each test, there is a recommended minimum bit length required as explained in Juan Soto *et al* (2000). As informed earlier, Parameterized Test Selection requires some parameter input before we run the statistical test. The parameter(s) selection characteristics for each test can be found in Juan Soto *et al.* (2000).

4. DATA DESCRIPTION

Five algorithms being tested are SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128. Nine different sets of data for each of these algorithms are generated and analyzed. The nine different sets of data are Strict Key Avalanche (SKA), Strict Plaintext Avalanche (SPA), Plaintext Ciphertext Correlation (PCC), Cipher Block Chaining (CBC), Random Plaintext Random Key (RPRK), Low Density Keys (LDK), High Density Keys (HDK), Low Density Plaintext (LDP) and High Density Plaintext (HDP). Each data set was selected due to their specific function (Juan Soto *et al.* (2000)). The process to generate the sample of each data types is in accordance with the paper of Juan Soto *et al.* (2000). The description of the data types is in Appendix B.

Table 3 shows the length of each sample generated according to the algorithm and data category.

Data Categories	**SIMON32/64**	**SIMON48/72**	**SIMON64/96**	**SIMON96/96**	**SIMON128/128**
SKA	1001472	1002240	1001472	1004544	1015808
SPA	1000448	1002240	1003520	1004544	1015808
PCC	1000000	1000032	1000000	1000032	1000064
CBC	1000000	1000032	1000000	1000032	1000064
RPRK	1000000	1000032	1000000	1000032	1000064
LDK	66592	126192	298048	447072	1056896
HDK	66592	126192	298048	447072	1056896
LDP	16928	56496	133184	447072	1056896
HDP	16928	56496	133184	447072	1056896

Table 3: The length in bits of each sample generated according to the algorithm and data category

5. ANALYSIS FRAMEWORK

The randomness testing was performed to five variations of SIMON family, which are SIMON 32/64, SIMON 48/72, SIMON 64/96, SIMON 96/96 and SIMON 128/128. The statistical analysis conducted was based on significance level of 0.1%. This is in accordance with NIST recommendation, where the value of significance level is to be at least 0.1%. A sample size is proportional to the significance level. Thus, for a level of 0.001, a sample of at least 1000 sequences has to be generated.

The list of parameter used in each of test contained in the parameterized test selection are 20,000 for Block Frequency Test, 10 for Overlapping Template Test and Non-overlapping Template Test, 2 for Serial Test and Approximate Entropy Test and 2,000 for Linear Complexity Test. For Universal Test, the input parameter depends on the length of the bit sequences of the sample.

All tests except for the Cumulative Sums Test, Serial Test, Non-overlapping Templates Test, Random Excursion Test and Random Excursion Variants Test, produce 1 p-value for each sample. Cumulative Sums and Serial Test produce 2 p-values for each test. Non-overlapping Template Test produces 148 p-values for each sample. As for the Random Excursions and Random Excursion Variants Test, the p-value produced will depend on the sample accepted. Only samples with number of cycle exceeding 500 were evaluated. Samples with insufficient number of cycles are not applicable. Note that all p-values are collected and analyzed. The randomness of the algorithm based on data categories is determined by this value. The sequence for each sample will be considered as random if the p-value is more or equal to 0.001. If the p-value of the sample is less than 0.001, the sample is rejected.

Since this analysis used 1000 samples and the significance level was fixed at 0.001, the rejection rate for most of the test in all data categories for all algorithms should not exceed 3 samples. The formula used to compute the maximum number of rejections is as follows:

$$s\left(\alpha + 3\sqrt{\frac{\alpha(1-\alpha)}{s}}\right)$$

where s is the sample size and α is the significance level (Juan Soto Jr., 2000). For example, Non-overlapping Template Test produces 148,000 p-values for each data categories in all algorithms. Therefore, the rejection rate for this test should not exceed 184 sequences.

The p-values produced in Random Excursion Test and Random Excursion Variants Test varied as it depended on the sample accepted. The rejection rates for these tests were different according to data categories and algorithm. The rejection rate for Random Excursion Test for all algorithms in all data categories was 11 except for Strict Key Avalanche data category in SIMON32/64 and SIMON64/96. The rejections for these two algorithms were 9 and 12 respectively. The rejection rate for Random Excursion Variants test for all algorithms in all data categories were 21 except in Strict Key Avalanche data category of SIMON32/64, SIMON48/72 and SIMON64/96. The rejection rates for these algorithms were 17, 20 and 22 respectively. Furthermore, the rejection rates in this test for SIMON48/72 and SIMON64/96 of Strict Plaintext Avalanche data category were also different. The rejection rate was 20 for each algorithm. The rejection rate for SIMON48/72 in Cipher Block Chaining data category and SIMON96/96 in Random Plaintext Random Key data category was 20 as well.

6. ANALYSIS RESULT

This statistical test was conducted on five chosen algorithms under nine data categories with each having 1000 samples. A total of 45,000 binary sequences were evaluated. Each sample in all data categories of SIMON128/128 has an output sequence length of at least 10^6 bit. Due to this reason, the algorithm was evaluated using all 15 NIST tests under all data categories. Five data categories on SIMON32/64, SIMON48/72, SIMON64/96 and SIMON96/96 that having a sample size of at least 10^6 is Strict Key Avalanche, Strict Plaintext Avalanche, Plaintext Ciphertext Correlation, Cipher Block Chaining and Random Plaintext Random Keys. Therefore, sample from these data categories were evaluated using all 15 NIST Statistical Test on these four algorithms.

Table 4 explained a case-by-case description according to algorithm. In this table, only the tests that exceeded the maximum number of rejection rate are discussed. As explain earlier, the rejection rate for most of the tests is 3 except for Non-overlapping Templates Test, Random Excursion Test and Random Excursion Variants Test. The rejection rates for these three tests are informed in Analysis Framework section.

Data Categories	Statistical Test	Number of rejection
SIMON32/64		
Strict Key Avalanche	Block Frequency	38
Strict Plaintext Avalanche	Approximate Entropy	4
	Random Excursion Variants	24
Plaintext Ciphertext Correlation	Runs	4
	Serial (p-value 2)	4
	Random Excursion Variants	23
Cipher Block Chaining	Cumulative Sums (Reverse)	4
	Linear Complexity	4
Random Plaintext Random Keys	Linear Complexity	4
	Random Excursion Variants	22
Low Density Key	Spectral	4
	Non-overlapping Templates	306
High Density Key	Non-overlapping Templates	282
Low Density Plaintext	Non-overlapping Templates	777
High Density Plaintext	Non-overlapping Templates	717
SIMON48/72		
Strict Key Avalanche	Cumulative Sums (Forward)	5
	Approximate Entropy	5
Strict Plaintext Avalanche	Maurer’s Universal	4
Plaintext Ciphertext Correlation	Overlapping	4
	Random Excursion	13
Cipher Block Chaining	Maurer’s Universal	4
Random Plaintext Random Keys	Frequency Test	4
	Cumulative Sums (Forward)	4

	Cumulative Sums (Reverse)	4
	Serial (p-value 1)	4
Low Density Keys	Non-overlapping Templates	220
High Density Keys	Longest Runs of Ones	5
	Non-overlapping Templates	253
Low Density Plaintext	Longest Runs of Ones	6
	Non-overlapping Templates	326
High Density Plaintext	Non-overlapping Templates	366
SIMON64/96		
Strict Key Avalanche	Linear Complexity	5
	Non-overlapping Templates	188
Random Plaintext Random Keys	Runs	5
	Serial (p-value 2)	5
Low Density Keys	Longest Runs of Ones	4
High Density Keys	Longest Runs of Ones	4
	Non-overlapping Templates	186
Low Density Plaintext	Non-overlapping Templates	212
High Density Plaintext	Longest Runs of Ones	4
	Non-overlapping Templates	219
SIMON 96/96		
Strict Key Avalanche	Non-overlapping Templates	210
Plaintext Ciphertext Correlation	Random Excursion Variants	26
Cipher Block Chaining	Linear Complexity	5
SIMON128/128		
Strict Key Avalanche	Non-overlapping Templates	223
Cipher Block Chaining	Overlapping	4
Random Plaintext Random Keys	Overlapping	4
Low Density Keys	Longest Runs of Ones	4
	Maurer’s Universal	4
	Random Excursion Variants	26
Low Density Plaintext	Low Density Plaintext	5

Table 4: Number of sample which exceeded the maximum number of rejection rate = 3 for most test except Non-overlapping Templates Test, Random Excursion Test and Random Excursion Variants Test.

7. CONCLUSION

In this research paper, we have examined the randomness of selected algorithm of SIMON Lightweight Block Cipher, namely SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128. This analysis was conducted to 1000 samples under nine data categories based on a significance level fixed to 0.001. From the analysis conducted, we have found that each algorithm have failed significantly in the NIST Statistical Tests. Although there are 15 main tests, due to the various length of output sequence in each algorithm under all nine data categories conducted, the total number of tests is more than 15. Therefore, the total number of tests performed for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128 are 113, 115, 115, 119 and 135 tests respectively. Based on the results shown in Table 4, we can see that SIMON32/64 have failed 15 tests out of 113 tests while SIMON48/72 have failed 15 tests out of 115 tests conducted. Furthermore, SIMON64/96, SIMON96/96 and SIMON128/128 have failed 10, 3 and 7 tests out of 115, 119 and 135 tests conducted respectively. Therefore, we conclude that the outputs from the algorithms tested are non-random on 0.1% significance level. Future works should cover the remaining algorithms in order to see if other key sizes of the same block size of algorithm are random or non-random.

REFERENCES

Abdullah, N.A.N., Zawawi, N.H.L.A. and Rani, H.A. 2011. Analysis on Lightweight Block Cipher, KTANTAN. *7th International Conference on Information Assurance and Security(IAS). 46-51. Dec 2011.*

Andrew R., Juan S., James N., Miles S., Elaine B., Stefan L., Mark L., Mark V., David B., Alan H., James D. and San V. 2010. A *Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.* NIST Special Publication 800-22.

Juan S. 1999. *Randomness Testing of the AES Candidate Algorithms*. NIST IR 6390, September 1999.

Juan S. and Lawrence B. 2000. *Randomness Testing of the Advanced Encryption Standard Finalist Candidates.* NISTIR 6483.

Lot, N.H, Abdullah, N.A.N. and Rani, H.A. 2011. Statistical Analysis on KATAN Block Cipher. *International Conference on Research and Innovation in Information Systems(ICRIIS), 2011. 1-6. Nov 2011.*

Norul H. L., Kamaruzzaman S. and Nurzi J. M. Z. 2013. Randomness analysis on Grain-128 Stream Cipher. *International Conference on Mathematical Sciences and Statistics 2013 (ICMSS2013): Proceedings of the International Conference on Mathematical Sciences and Statistics 2013. AIP Conference Proceedings, Volume 1557, pp. 15-20(2013). Sept 2013.*

Ray B., Douglas S., Jason S., Stefan T., Bryan W., and Louis W. 2013. *The SIMON and SPECK Families of Lightweight Block Ciphers*. Cryptology ePrint Archive, Report 2013/404, 2013.

APPENDIX A: Description of the Statistical Tests

Parameterized Test Selection

1. *Block Frequency Test*: To determine whether the number of ones in an M – bit block is approximately M/2 where M is the length of each block.

2. *Overlapping Templates Test* : To reject sequences that shows deviations from the expected number of runs of ones of a given length.

3. *Non – Overlapping Templates Test*: To reject sequences that exhibit too many occurrences of a given non – periodic pattern.

4. *Serial Test*: To determine whether the number of occurrences of *m*-bit overlapping patterns is approximately the same as would be expected for a random sequence (*m*-bit is referred to the length in bits of each block).

5. *Approximate Entropy Test*: To compare the frequency of overlapping blocks of two consecutive/adjacent lengths *(m and m+1)* against the expected result for a normally distributed sequence (*m*-bit is referred to the length of each block).

6. *Linear Complexity Test*: To determine whether the sequence is enough to be random or not.

7. *Universal Test*: To detect whether the sequence can be significantly compressed without loss of information or not.

Non – Parameterized Test Selection

1. *Cumulative Sums Test*: To determine whether the sum of the partial sequences occurring in the tested sequence is too large or too small.

2. *Runs Test*: To determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence.

3. *Longest Runs of Ones Test*: To determine whether the longest run of ones is consistent with the longest run of ones that would be expected in a random sequence.

4. *Rank Test*: To check for linear dependence among fixed length substrings of the original sequence.

5. *Spectral DFT Test*: To detect periodic features in the tested sequence that would indicate a deviation from the assumption of randomness.

6. *Random Excursions Test*: To determine if the number of visits to a particular state within a cycle deviates from what one would expect for a random sequence.

7. *Random Excursions Variant Test*: To detect deviations from the distributions of the number of visits of a random walk to a certain state.

8. *Lempel – Ziv Complexity Test*: To determine how far the tested sequence can be compressed.

9. *Frequency Test*: To determine whether the number of zeros and ones in a sequence are approximately the same as would be expected for a truly random sequence

APPENDIX B: Description of the Data Types

A. *Strict Key Avalanche (SKA) and Strict Plaintext Avalanche (SPA)*

To examine the Strict Key Avalanche and Strict Plaintext Avalanche data category, 1000 samples are generated for each algorithm.

The samples for Strict Key Avalanche are constructed as follows: given a number of base-key blocks and a set of all zeroes plaintext block. The total bit of each base-key block and plaintext block are relying on the size of the algorithm. The base-key is encrypted with the all zero plaintext block to derive a block of base-ciphertext. Then, every bit of the base-key is flipped and encrypted with the respective length of all zero plaintext blocks to get the perturbed-ciphertext. Every block of perturbed-ciphertext is then XORed with the base-ciphertext and concatenated to produce a derived block at least 10^6–bit output for each sample. The numbers of base-key blocks for this data category are 489, 218, 163, 109 and 62 for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128 respectively.

In the case of Strict Plaintext Avalanche, The numbers of base-key blocks are 977, 435, 245, 109 and 62 for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128 respectively. Furthermore, substitute “base-key” for “base-plaintext” in the above description and “plaintext” for “key”.

B. *Plaintext Ciphertext Correlation (PCC)*

In order to study the correlation of plaintext/ciphertext pairs, 1000 samples are generated for each algorithm. Each sequence consists of at least 10^6–bits. 31250, 20834, 15625, 10417 and 7813 blocks of random plaintext and keys for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/18 respectively are used to produce the concatenated ciphertext block of 1000000, 1000032, 1000000, 1000032 and 1000064 bits of output sequences. The output sequences are constructed by XORed operation between the plaintext block and its corresponding ciphertext block which is computed in ECB mode.

D. *Cipher Block Chaining (CBC)*

In this data category, 1000 samples are generated using 31250, 20834, 15625, 10417 and 7813 blocks of random keys for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128 respectively for each sample. The derived block of 1000000, 1000032, 1000000, 1000032 and 1000064 bit for each respectively algorithm from this data categories are constructed using CBC mode. The first ciphertext block (*CT1*) is defined by $CT1 = Ek$ (IV $\oplus$ *PT*). Subsequent ciphertext blocks were defined by $CTi+1 = Ek$ ($CTi \oplus PT$) for $1 \leq i \leq N$ (*N* is 31250, 20834, 15625, 10417 or 7813 blocks of random keys). These derived blocks are then concatenated to produce an output of at least 10^6 bits.

E. *Random Plaintext/Random Key (RPRK)*

To study this data category, 1000 samples are generated. Each sample was a result of the concatenation of 31250, 20834, 15625, 10417 and 7813 blocks of ciphertext using 31250, 20834, 15625, 10417 and 7813 blocks of random plaintexts and random keys for SIMON32/64, SIMON48/72, SIMON64/96, SIMON96/96 and SIMON128/128 algorithm respectively. Each of these algorithms will produce at least 10^6 bits of sequences.

F. *Low Density Keys (LDK) and High Density Keys (HDK)*

In Low Density Keys data category, five sets of data consisted 1000 sequences are created according to the algorithm tested. For each key size, a random plaintext block is used. The first ciphertext blocks are obtained using a block of all zeroes keys. Please be informed that the lengths of the blocks of plaintext in this test are relying on the size of the algorithm. Ciphertext block 2-65/ 2-75/ 2-97/ 2-97/ 2-129 are obtained using a block of keys with a single one in each of the possible bit positions. Then, for ciphertext block 66-2081/ 76-2629/ 98-4657/ 98-4657 / 130-8257 are obtained using a block of keys with two ones and 62/70/94/94/126 zeroes (the two ones appear in each combination of two bit position within the length of the key). The derived block of ciphertext is concatenated.

In the case of High Density Keys, substitute “zeroes” for “ones” in the above description and “ones” for “zeroes”.

H. *Low Density Plaintext (LDP) and High Density Plaintext (HDP)*

In Low Density Plaintext data category, a sets of data consisted 1000 sequences are created for each algorithm. For each block size, a random key block is used. The first ciphertext blocks are obtained using a block of all zeroes plaintext. Please be informed that the lengths of the blocks of keys in this test are relying on the size of the algorithm. Ciphertext block 2-33/ 2-49/ 2-65/ 2-97/ 2-129 are calculated using a block of plaintext with a single one in each of the possible bit positions. Then, for ciphertext block 34-529/ 50-1177/ 66-2081/ 98-4657 / 130-8257 are obtained using a block of plaintext with two ones and 62/70/94/94/126 zeroes (the two ones appear in each combination of two bit position within the length of the plaintext). The derived block of ciphertext is concatenated.

In the case of High Density Plaintext, substitute “zeroes” for “ones” in the above description and “ones” for “zeroes”.

Linear Extension Cube Attack on Stream Ciphers

[1]Liren Ding [2]Yongjuan Wang [3]Zhufeng Li

[1,2,3]Language Engineering Department, Luo yang University for Foreign Language, Luo yang city, He nan Province, 471003, P. R. China

Email: [1,2,3]willssponge@icloud.com

ABSTRACT

Basing on the original Cube attack, this paper proposes an improved method of Cube attack on stream ciphers, which makes improvement on the pre-processing phase of the original attack. The new method can induce maxterms of higher-order from those of lower-order by the trade-off between time and space, thus recovering more key bits and reducing the search complexity on higher-dimension. In this paper, the improved attack is applied to Lili-128 algorithm and reduced variants of Trivium algorithm. We can recover 88 key bits of Lili-128 algorithm within time complexity of $O(2^{14})$ and 48 key bits of Trivium algorithm can be recovered by cubes with dimension no larger than 8 when the initialization round is 576, the results are much better than those of the original attacks.

Keywords: Cube Attack, Stream Cipher, Linear Extension, Pre-processing, Trivium, Lili-128

1. INTRODUCTION

In 2008, Dinur and Shamir (Dinur and Shamir, 2009) introduced a deterministic analysis method basing on algebraic theory, i.e. Cube Attack. Cube attack is a kind of chosen plaintext attack, it makes use of the fact that the output of a cryptosystem can be represented as a polynomial of public variables and key bits. Theoretically, Cube attack can be applied to all cryptosystems as long as they can be represented as tweakable polynomials. The algebraic degree of the master polynomial can be decreased by choosing arbitrary values for tweakable variables. By doing so, the attacker can obtain linear equations about the key bits. Hence, the security of a cryptosystem is reduced to the problem of solving a linear equation system. Although the specific representation of a cryptosystem is unknown, the attack can still be applied to a "black box". The performance of Cube attack against stream ciphers, block ciphers and hash functions is quite well ((Pierre, 2013), (Zhao, Wang and Guo, 2011), (Aumasson, Meier and Dinur, 2009)). Therefore, Cube attack has achieved lots of attention since its announcement.

However, there being few limitations of Cube attack. First of all, the algebraic degree of a certain cryptosystem should not be too large, otherwise it would be difficult to search for linear expressions on the pre-processing phase. As a result, the performance of original Cube attack against NFSR based stream ciphers and block ciphers is limited. Second, choosing the right cubes is time-consuming. If the choice is not proper or the search has not been done thoroughly, the final result of Cube attack would be weakened.

With the research of Cube attack going deeper, cryptologists have come up with several extended versions of Cube attack against different algorithms. In 2010, Dinur and Shamir (Dinur and Shamir , 2011) proposed Dynamic Cube attack, which is based on the analysis of the internal structure of a certain cipher so that the proper choice for the values of such dynamic variables can lower the algebraic degree of the master polynomial. P. Mroczkowki (Mroczkowski and Szmidt, 2012) improved the linearity test of Cube attack by extracting quadratic equations. Yu Sun (Sun and Wang, 2012) made improvements on the search algorithm and linearity test so that more than one linear expressions can be extracted from only one cube with the help of the trade-off between time and space.

Our work is based on the phenomenon pointed out by P. Mroczkowki in (Mroczkowski and Szmidt, 2012) that although the cubes were chosen randomly, there was a way to obtain new cubes from another one with linear expression. In fact, for stream ciphers with shift register, whose transitivity enables the lower-round to influence the higher-round so that d and $d+1$ dimensional cubes will have some variables in common. Therefore, more cubes can be derived from one with linear expression, thus obtaining more linear expressions.

Motivated by the above observations, this paper proposes an improved attack on stream ciphers on the basis the original Cube attack, i.e. Linear Extension Cube Attack, which makes use of those common variables in two different dimensional cubes and the trade-off between time and space enables the attacker to induce maxterms of higher-order from those of lower-order, thus recovering more key bits and reducing the the search complexity on higher-dimension.

To demonstrate the application of the extended method, this correspondence provides Linear Extension Cube Attack against Lili-128 algorithm and two reduced variants of Trivium algorithm and the results are much better than those of the original attacks. For Lili-128 algorithm, only $O(2^9)$ key streams are needed to recover 88 key bits and the attack has time complexity less than $O(2^{14})$, which is better compared to $O(2^{16})$ of the original Cube attack (Li, Wang and Wang, 2010). And it is the best attack against Lili-128 to the best of our knowledge. For Trivium algorithm, when the initialization round is 525, the original Cube attack on the 525th output bit with 4 dimensional cubes can only recover 6 key bits, while the improved attack can directly recover 31 key bits. When the initialization round is 576, applying the improved Cube attack to the 576th output bit, 48 key bits can be recovered by cubes with dimension no larger than 8, while the original Cube attack on several output bits in (Bedi and Pillai, 2009) can only recover 36 key bits by cubes with dimension no larger than 8.

The organization of this paper is as follows, in section 2, the preliminaries and basic steps of Cube attack are reviewed. Section 3 contains the main contribution of this paper, where the notion of Linear Extension Cube Attack and its details are provided. To testify the performance of the improved attack, it is applied to Lili-128 and two reduced variants of Trivium algorithm in section 4. At last, section 5 is a brief summary of this paper.

2. REVIEW ON CUBE ATTACK

Cube attack regards the investigated cryptosystem as a polynomial $P(v,k)$ about the public variables $v=(v_1,...,v_m)$ and secret key bits $k=(x_1,...,x_n)$. Let $I=\{i_1,...,i_k\}\subseteq\{1,2,...,n\}$ be the indexes of the public variables and $t_I=v_{i_1}\cdots v_{i_k}$ be the product of these public variables, through factoring the master polynomial by the monomial t_I, we have:

$$P(v_1,...,v_m,x_1,...,x_n)=t_I\cdot P_{S(I)}+Q(v_1,...,v_m,x_1,...,x_n).$$

where $P_{S(I)}$, which is called the superpoly of t_I, does not have any common variables with t_I, and each monomial term in the residue polynomial $Q(v_1,...,v_m,x_1,...,x_n)$ misses at least one variable from t_I.

Denote $C_I=\{(v_1,...,v_m)\in F_2^m \mid v_i\in F_2, i\in I; v_i=0, i\notin I\}$ as a ***Cube***, apparently , we have $|C_I|=2^{|I|}$. Summing the polynomials when the public variables $v=\{v_1,...,v_m\}$ walking over the cube, we have:

$$\sum_{(v_1,...v_m)\in C_I}P(v,k)=\sum_{(v_1,...v_m)\in C_I}t_I\cdot P_{S(I)}+\sum_{(v_1,...v_m)\in C_I}Q(v_1,...,v_m,x_1,...,x_n).$$

The feasibility of Cube attack mainly depends on the following observations:

Theorem 1((Dinur and Shamir, 2009) For any polynomial $P(v,k)$ and public variable $v=(v_1,...,v_m)$, we have $\sum\limits_{v\in C_I} P(v,k)=P_{S(I)}(v,k) \bmod 2$.

Proof: Each monomial term in the residue polynomial of $P(v,k)$,that is, $Q(v_1,...,v_m,x_1,...,x_n)$ misses at least one variable from t_I , the value of each term from $Q(v,k)$ after 2^k times computation is 0, thus $\sum\limits_{v\in C_I} Q(v,k)=0$. Next, if and only if $v_{i_1},...,v_{i_k}$ are all set to 1, the coefficient of $P_{S(I)}(v,k)$ would be 1. #

A term t_I is called a ***maxterm*** if its superpoly is a linear polynomial while not a constant.

Example 1: Let $P(v_1,v_2,v_3,x_1,x_2,x_3)=v_1v_2x_1+v_1v_2x_3+v_1v_3x_1+v_1v_2+x_2+1$, where (v_1,v_2,v_3) being the public variables and (x_1,x_2,x_3) being the key. And define *I*={1,2}, through factoring the master polynomial by the monomial t_I , we have:

$$P(v_1,v_2,v_3,x_1,x_2,x_3)=v_1v_2(x_1+x_3+1)+v_1v_3x_1+x_2+1.$$

$C_I=\{(0,0,0),(1,0,0),(0,1,0),(1,0,0)\}$, by summing the polynomial over this cube, we have:

$$\sum_{v\in C_I} P(v_1,v_2,v_3,x_1,x_2,x_3)=P_{S(I)}(v_3,x_1,x_2,x_3)=x_1+x_3+1.$$

Cube attack consists of two phases, the preprocessing phase and the on-line phase. During the preprocessing phase, the attacker can arbitrarily assign values to both the public variables and the key bits, and choose appropriate cubes to do the computation. Then the linearity test is applied to test whether the superpoly is linear or not. The main purpose of this phase is to extract linear expressions about the key bits as many as possible. During the on-line phase, the attacker can only control the public variables, through which he can conduct the cube sum over the same cube in order to obtain the value of the right side of a certain expression. The main purpose of this phase is to establish linear equations. Due to the selection of different cubes, he can establish a system of linear equations. At last, the key bits are recovered by solving the equation system.

3. LINEAR EXTENSION CUBE ATTACK

3.1 The Main Observation

When it comes to the shift register, a higher-position of a lower-round will become a lower-position of a higher-round with the extension of round itself. Therefore, the existence of common variables between the master polynomials before and after the extension is certain. Using these common variables to construct new cubes, the attacker can analysis the target cryptosystem of higher rounds on the basis of the attack results of a lower round. Motivated by the above observation, this paper proposes Linear Extension Cube Attack against stream ciphers. The main idea of this extended attack is to make improvement on the pre-processing phase of the original attack. Thanks to the transitivity of the shift register, by the trade-off between time and space, maxterms of degree $d+1$ can be derived from those maxterms of degree d , thus obtaining more linear expressions so that more key bits can be recovered.

To demonstrate the main contribution, definitions and examples are given as follows.

Definition 1 Let $N=\{1,2,...,n\}$, $I=\{i_1,i_2,\cdots,i_k\}$, $I\subset U$, $|I|=k$, define the maxterm as $t_I=v_{i_1}v_{i_2}\cdots v_{i_k}$, through factoring the master polynomial by t_I , we have: $P=t_IP_{S(I)}+Q$. Let $t_{I+1}=t_Iv_s$, where $s\in N\setminus I$, $P^{'}=t_{I+1}P_{S(I+1)}+Q^{'}$ denotes the master polynomial after extension, monomial t_{I+1} is called the **1-time extended maxterm** of t_I if $\deg(P_{S(I+1)})=1$.

Example 2:

$$P(v_1,v_2,v_3,v_4,x_1,x_2,x_3,x_4) \quad =v_1v_2v_3x_4+v_1v_2v_3+v_2v_3x_1+v_1v_3x_2+v_1v_2x_3+x_1x_2$$

$$= v_1v_2v_3(x_4+1)+v_2v_3x_1+v_1v_3x_2+v_1v_2x_3+x_1x_2 \tag{1}$$

$$P^{'}(v_1,v_2,v_3,v_4,x_1,x_2,x_3,x_4) \quad = v_1v_2v_3v_4x_1+v_1v_2v_3+v_2v_3x_1+v_1v_3x_2+v_1v_2x_3+x_1x_2 \tag{2}$$

$P^{'}$ denotes the master polynomial after extension, let $I=\{1,2,3\}$, we have maxterm as $t_I = v_1v_2v_3$ according to (1), let $I^{'}=\{1,2,3,4\}$, then $t_{I^{'}} = t_{I+1} = v_1v_2v_3v_4$ is also a maxterm according to (2), so t_{I+1} is called the 1-time extended maxterm of t_I.

Note that, to conduct the Linear Extension Cube Attack, the target cryptosystem should be extended first so that the algebraic degree of master polynomial would increase. For extension here, it implies one round as well as several rounds, the existence of 1-time extended maxterm holds true for both conditions. The following of this section discusses the existence of 1-time extended maxterm under the extension of round.

Theorem 2 $P(x)$ denotes the master polynomial of a cryptosystem, through factoring $P(x)$ by maxterm t_I, we have: $P(x)=t_IP_{S(I)}+Q$, $P^{'}(x)$ denotes the polynomial after extension by one round, then the 1-time extended maxterm of t_I is existed by the possibility of 1 if t_I satisfies that $P^{'}(x)=t_IP^{'}{}_{S(I)}+Q^{'}$, where $\deg(P^{'}{}_{S(I)})=2$, and can be represented as $v_a \cdot l(x)+g(x)$, where $\deg(l(x))=1$ and $a \notin I$.

Proof: Since t_I is a maxterm of $P(x)$, we have: $P(x)=t_IP_{S(I)}+Q$. Assuming that $t_{I+1}=t_I \cdot v_a$ is a 1-time extended maxterm of t_I, where $a \notin I$ according to definition 1, then we have $P^{'}(x)=t_{I+1}P^{'}{}_{S(I)}+Q^{'}$, where $\deg(P^{'}{}_{S(I)})=1$. Obviously, the following equations hold true:

$$P^{'}(x)=t_I \cdot v_aP^{'}{}_{S(I)}+Q^{'} \tag{3}$$

$$P^{'}(x)=t_IP^{''}{}_{S(I)}+Q^{''} \tag{4}$$

Note that, $t_I \cdot v_a \nmid Q^{'}$, $t_I \nmid Q^{''}$ and $t_I \mid t_I \cdot v_a$, then we have:

$$Q^{'}=t_Ig(x)+Q^{''} \tag{5}$$

Subsitute (5) into (3), then combine it with (4), we have:

$$\begin{aligned} t_IP^{''}{}_{S(I)}+Q^{''} &= t_I \cdot v_aP^{'}{}_{S(I)}+Q^{'} \\ &= t_I \cdot v_aP^{'}{}_{S(I)}+t_Ig(x)+Q^{''} \\ &= t_I(v_aP^{'}{}_{S(I)}+g(x))+Q^{''} \end{aligned}$$

In conclusion, $P^{''}{}_{S(I)}=v_a \cdot P^{'}{}_{S(I)}+g(x)$ meets the criteria for the existence of t_{I+1}. #

Corollary 1 $P(x)$ denotes the master polynomial of a cryptosystem and t_I denotes one of its maxterms, let $P^{'}(x)$ be the polynomial after the extension of $r(1<r<n)$ round, then the 1-time extended maxterm of t_I is existed by the possibility of 1 if t_I satisfies the criteria in Theorom 2.

Proof: According to the proof of Theorom 2, the times of extension would not make any differences on the existence of t_{I+1}. #

Note that, although the existence of 1-time extended maxterm is not strictly related to the times of extension, it is impossible to find maxterm through increasing the cubes by one when the algebraic degree of master polynomial goes too large after several rounds of extension.

Example 3:

$$\begin{aligned} P(x_1,\ldots,x_5) \quad &= x_1x_2x_3+x_2x_3x_4+x_1x_4x_5 \\ &= x_2x_3(x_1+x_4)+x_1x_4x_5 \end{aligned}$$

$$\begin{aligned} P^{'} &= P(x_2,...,x_5,x_6) \\ &= x_3x_4(x_2+x_5)+x_2x_5x_6 \\ &= x_3x_4(x_2+x_5)+x_2x_5(x_1x_2x_3+x_2x_3x_4+x_1x_4x_5) \\ &= x_3x_4(x_2+x_5)+x_1x_2x_3x_5+x_2x_3x_4x_5+x_1x_2x_4x_5 \\ &= x_2x_3x_4(1+x_5)+x_3x_4x_5+x_1x_2x_3x_5+x_1x_2x_4x_5 \\ &= x_2x_3[x_4(x_1+x_5)+x_1x_5]+x_3x_4x_5+x_1x_2x_4x_5 \end{aligned}$$

$P^{'}$ denotes the master polynomial after extension, note that $t_I = x_2x_3$ is a maxterm of P, there exists $t_{I+1} = t_I \cdot x_4$, which satisfies the criteria in Theorom 2, being the 1-time extended maxterm of t_I.

3.2 Attack Procedure

Compared to the original Cube attack, the Linear Extension Cube Attack makes improvement on the pre-processing phase. Cube attack consists of two phases, pre-processing phase and on-line phase. The improved attack, which makes use of the cubes with linear expressions, consists of the original pre-processing phase, the improved pre-processing phase and on-line phase. The following figures reveal the difference between the two versions of Cube attack:

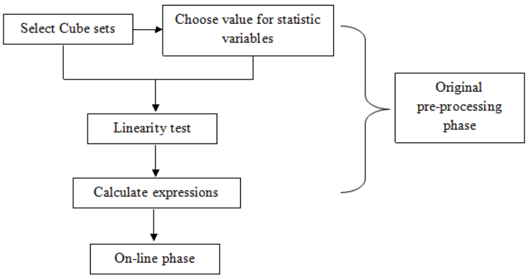

Select Cube sets
Choose value for statistic variables
Linearity test
Calculate expressions
On-line phase
Original pre-processing phase

Figure 1: Procedures of Original Cube Attack

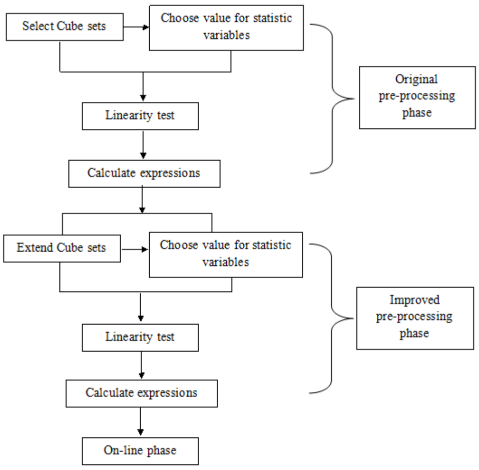

Select Cube sets
Choose value for statistic variables
Linearity test
Calculate expressions
Original pre-processing phase
Extend Cube sets
Choose value for statistic variables
Linearity test
Calculate expressions
Improved pre-processing phase
On-line phase

Figure 2: Procedures of Linear Extension Cube Attack

As shown above, the details of the improved attack display as follows:

Phase 1, the original pre-processing phase. In this phase, the attacker can arbitrarily choose values for both public variables and key bits in order to find proper cubes.

(1) Choose the initial dimension as d and walk over all of the d dimensional cubes;

(2) Sum over each cubes and use linearity test to find maxterms;

(3) Calculate algebraic normal forms for each linear expression.

Phase 2, the improved pre-processing phase. The attack can search $d+1$ dimensional cubes on the basis of the results of the original pre-processing phase.

(1) Store all of the d dimensional cubes obtained via phase 1;

(2) Extend the investigated cryptosystem by extension of round or choosing the next output bit;

(3) Extend each maxterm of degree d respectively. Firstly, modify each variable of the d dimensional cube according to the regulation of the target algorithm. The modification here means increase by one (or *r*), decrease by one (or *r*) or stay the same for each variable. Secondly, add a new variable from the public variables to set I to form a new cube;

(4) Sum over the new $d+1$dimensional cube and use linearity test to find maxterms;

(5) Calculate algebraic normal forms for each linear expression;

(6) Testify the linear dependence of linear expressions by Gaussian elimination.

Phase 3, the on-line phase. According to the cubes obtained by phase 1 and phase 2, the attacker can choose values for the public variables to obtain the value of the right side of each expression. At last, the key bits are recovered by solving the linear equations system.

Special attention should be paid to the following steps:

(1) The extension of the targeted algorithm is necessary after walking over all of the d dimensional cubes since there is no 1-time extended maxterms under the same polynomial;

(2) There is no need for Gaussian elimination after walking over all of the d dimensional cubes since linearly independent expressions can be derived from linearly dependent ones. Therefore, Gaussian elimination is introduced after the extension in the improved attack.

The pseudo-code of the improved pre-processing phase is as follows:

```
Algorithm 1:The Improved Pre-processing Phase
Input: $V$ ; // set of public variables, $v_i$ denotes each variable
    $T$ ; // set of maxterms of degree $d$ , $I_i$ denotes each maxterm
    $R$ ; // number of round extension
    $r = 0$ ; // initialization of round extension
Output: linear expressions
repeat
  $r++$ ;
  repeat
    Choose an $I_i$ which has not been chosen from $T$ ;
    repeat
      Modify each variable in $I_i$ ;
      Choose a $v_i$ which has not been chosen from $V$ , i.e. $I_i^{'} = I_i \cup \{v_i\}$;
      Sum over $I_i^{'}$ ;
      Introduce linearity test;
    until
      Walk over $V$ ;
  until
    Walk over $T$ ;
until
  $r = R$;
```

Figure 3: The Pseudo-code of the Improved Pre-processing Phase

4. THE APPLICATION OF LINEARITY EXTENSION CUBE ATTACK

To demonstrate the application of the extended method, this correspondence provides Linear Extension Cube Attack against Lili-128 algorithm and two reduced variants of Trivium algorithm when the initialization round is 525 and 576 by using the personal computer equipped with i-5 CPU, 1.7GHz dominant frequency and 2GB RAM.

4.1 Attack 1: Lili-128

4.1.1 Brief Introduction of Lili-128 Algorithm

Lili-128 (Dawson, Clark and Golic, 2000) is one of the candidate algorithms of NESSIE. The clock-controlled algorithm is consists of two linear feedback shift registers, a linear function and a non-linear function, where one LFSR is used to clock control and the other to generate key stream. The following figure displays it in detail:

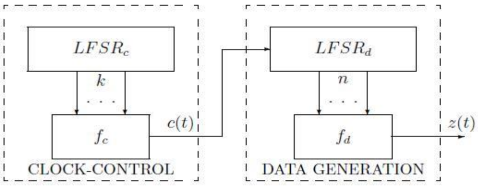


Figure 4: The Structure of Lili-128 Algorithm

$LFSR_c$ contains 39 bits for clock controlling, $k = 2$, $c(t)\,(1 \le c(t) \le 4)$ is generated by f_c, where $f_c(x_{12}, x_{20}) = 2x_{12} + x_{20} + 1$. $LFSR_d$ contains 89 bits for key stream generation, $n = 10$, and the 10 input positions of f_d, with the algebraic degree of 6, the nonlinearity of 480 and correlation immunity of 3, are (0, 1 , 3, 7, 12, 20, 30, 44, 65, 80).

4.1.2 Attack Procedure

Step 1, initialization. Since $LFSR_c$ does not directly control the bit generation, attacking against $LFSR_d$ is a common choice. To make the extended Cube attack possible, an initialization process is introduced to Lili-128 to make sure that the initial vector and key bits are blended thoroughly.

Step 2, choose annihilator. According to the result of algebraic attack in (Li, Wang and Wang, 2010), $g(x) = x_{44}x_{80}$ is one of the annihilators of f_d which render $\deg(f_d \cdot g) = 4$ instead of 8.

Step 3, choose initial dimension of cubes and the output bit. In this paper, the 20th output bit is attacked and we search cubes with the dimension of 2 to obtain linear expressions thanks to the annihilator which reduces the algebraic degree of master polynomial to 4.

Step 4, 1-time extension of 2-degree maxterms.
(1) Bound the extension of round as 2;
(2) choose one 2 dimensional cube a time until they are walked over;
(3) modify each variable in the cube according to the algorithm;
(4) choose one number a time from 0 to 87 to form a new 3 dimensional cube ;
(5) sum over the new cube and extract linear expression by linearity test;
(6) extend the algorithm by one round until it meets the bound.

4.1.3 Attack Result

We find 161 cubes with the dimension of 2 within seconds when the initialization round is bounded from 176 to 178. What's more, on the improved pre-processing phase, only 19 2-degree maxterms are needed to extend enough 3 dimensional maxterms, with which 88 linearly dependent expressions can be obtained. The cubes used for extension are as follows:

Round	Cube	Output bit	Round	Cube	Output bit
178	{23, 45}	20	177	{22, 44}	20
176	{1, 50}	20	176	{1, 65}	20
176	{2, 50}	20	176	{3, 50}	20
176	{4, 38}	20	176	{5, 56}	20
176	{8, 56}	20	176	{11, 12}	20
176	{12, 52}	20	176	{18, 57}	20
176	{38, 62}	20	176	{31, 47}	20
176	{31, 58}	20	176	{2, 65}	20
176	(3, 65}	20	176	{6, 65}	20
176	{8, 65}	20			

Table 1: 2-degree maxterms used for extension

Linear Extension Cube Attack searches 3 dimensional cubes on the basis of 2 dimensional cubes, which dramatically reduces the search scale on 3 dimensional cubes. Only 19×87 times computation are needed to extract enough expressions so that the time complexity of the improved attack on pre-processing phase is $O(2^{13})$ and the total time complexity of our attack is less than $O(2^{14})$. And its data complexity is $O(2^{9})$ since the 88 linear expressions are obtained from 5 cubes with the dimension of 2 and 83 cubes with the dimension of 3.

Compared to the previous attacks on Lili-128, our attack is the best as the following table displays:

	Time and Space Attack (Saarinen, 2002)	Algebra Attack (Courtouis and Meier, (2003)	Fast Algebra Attack (Courtois, 2003)	Cube Attack (Li, Wang and Wang, 2010).	Our Attack
Key stream	2^{46}	2^{18}	2^{60}	2^{12}	2^{9}
Pre-processing				2^{16}	2^{13}
On-line	2^{48} DES	2^{57}	2^{34}	2^{12}	2^{10}
Total complexity	2^{48} DES	2^{57}	2^{34}	2^{16}	2^{14}

Table 2: Complexity of Various Attacks on Lili-128

4.2 Attack 2: Reduced Variants of Trivium

4.2.1 Brief Introduction of Trivium Algorithm

Trivium (Dinur and Shamir, 2009) is one of the candidate algorithms of eSTREAM. The length of IV is 80 bits and the initial key is also 80 bits. During the initialization, the internal state is updated for 1152 rounds. The 288 bits of its internal state store in 3 nonlinear feedback shift register with different length. The following figure displays it in detail:

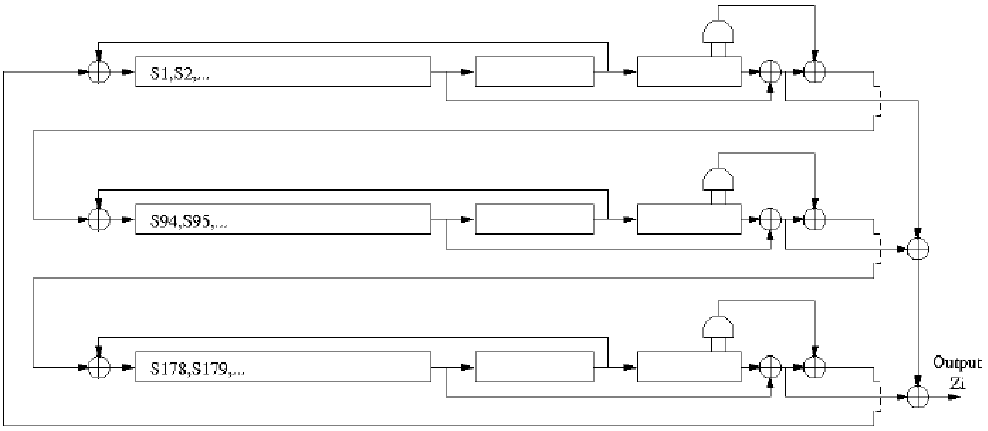


Figure 5: The Structure of Trivium Algorithm

The feedback to each register consists of a nonlinear combination of bits from different registers. The output bit of each round is a linear combination of six state bits, two taken from each register.

4.2.2 Attack Procedure

Step 1, choose initial dimension of cubes. When the initialization round is 525, we randomly search 4 dimensional cubes for maxterms and linear expressions. When the initialization round is 576, according to paper (Bedi and Pillai, 2009), we search cubes with the dimension no larger than 7.

Step 2, choose the output bit. When the initialization round is 525, we apply the improved attack to the 525th output bit. On the original pre-processing phase, 8 cubes with the dimension of 4 can be obtained ***without*** Gaussian elimination and they are extended on the improved pre-processing phase. The cubes are as follows:

Cube	Expression	Cube	Expression
{0,13,14,70}	x_{57}	{11,21,38,74}	x_{63}
{13,23,72,75}	x_{61}	{34,62,67,75}	x_{54}
{25,26,31,57}	x_{27}	{4,50,72,75}	x_{61}
{22,40,42,68}	x_{27}	{2,6,35,45}	x_{47}

Table 3: Original Results on the 525th Output Bit

When the initialization round is 576, we apply the improved attack to the 576th output bit. On the original pre-processing phase, 20 cubes with the dimension of 6 or 7 can be obtained ***without*** Gaussian elimination and they are extended on the improved pre-processing phase. The cubes are as follows:

Cube	Expression	Cube	Expression
{22,32,35,47,68,77}	x_{55}	{5,7,17,54,58,74}	x_{19}
{9,15,21,40,67,69}	x_{56}	{24,33,42,46,68,76}	x_{20}
{2,6,32,36,38,57}	x_6	{11,30,34,45,51,66}	x_{19}
{18,20,53,56,76,78}	$x_{22}+1$	{5,15,26,42,49,62}	x_4
{8,10,18,26,30,73}	x_{62}	{22,23,37,56,60,67,77}	x_{19}
{7,9,23,25,50,55,71}	x_{27}	{1,9,11,12,22,41,69}	$x_{61}+x_{67}+1$
{10,11,13,24,29,55,62}	x_{59}	{9,34,44,57,65,68,78}	x_{57}
{1,8,10,15,35,51,70}	x_{57}	{8,18,26,31,37,63,64}	x_{20}
{4,20,23,58,64,68,71}	x_{24}	{20,23,32,34,53,58,74}	x_{19}
{1,13,27,30,35,45,65}	$x_{61}+x_{67}+1$	{7,8,26,29,57,61,77}	x_{64}

Table 4: Original Results on the 576th Output Bit

Step 3, 1-time extension of 12-degree maxterms.

(1) Bound the extension of round as 5 and extend the algorithm by 1 round;
(2) choose one dimensional cube a time from table 2 (table 3) until they are walked over;
(3) modify each variable in the cube according to the algorithm;
(4) choose one number a time from 0 to 79 to form a new 13 dimensional cube ;
(5) sum over the new cube and conduct linearity;
(6) extend the algorithm by one round until it meets the bound.

4.2.3 Attack Result

When the initialization round is 525, 28 5-dimensional cubes can be obtained *after* Gaussian elimination by extending the 8 4-dimensional cubes in table 3. And *31* key bits can be directly recovered altogether. The extended cubes are as follows:

Cube	Expression	Bit	Cube	Expression	Bit
{2,15,16,72,27}	x_{54}	526	{3,16,17,73,4}	x_{62}	527
{5,18,19,75,7}	x_7	529	{5,18,19,75,36}	$x_{38}+x_{62}$	529
{13,23,40,76,39}	x_{65}	527	{14,24,41,77,43}	$x_{47}+1$	528
{14,24,41,77,44}	x_{46}	528	{14,24,41,77,67}	$x_{56}+1$	528
{14,24,41,77,68}	x_{55}	528	{15,25,42,78,56}	x_{22}	529
{14,24,73,76,3}	x_{60}	526	{14,24,73,76,41}	x_{26}	526
{15,25,74,77,1}	x_{64}	527	{15,25,74,77,1}	$x_{64}+x_{63}$	527
{36,64,69,77,10}	x_{58}	527	{36,64,69,77,12}	$x_{66}+1$	527
{38,66,71,79,10}	$x_{53}+x_{68}+1$	529	{38,66,71,79,13}	x_{23}	529
{35,63,68,76,44}	x_{48}	526	{29,30,35,61,17}	x_{37}	527
{30,31,36,62,38}	$x_{55}+x_{66}+1$	528	{30,31,36,62,39}	x_{51}	528
{8,54,76,79,78}	x_0	529	{23,41,43,69,14}	x_{10}	526
{24,42,44,70,56}	x_{11}	527	{25,43,45,71,13}	x_{65}	528
{6,10,39,49,51}	x_{53}	529	{7,11,40,50,8}	x_{52}	530

Table 5: The Extended Cubes on the 525th Output Bit

When the initialization round is 576, *26* key bits can be directly recovered and a linear equation about another 2 key bits can be obtained by extending those cubes in table 3, while the original attack on this output bit can only recover *13* bits and obtain a linear equation about another 2 key bits. The extended cubes are as follows:

Cube	Expression	Bit
{23,33,36,48,69,78,62}	x_{56}	577
{24,34,37,49,70,79,29}	x_{57}	578
{6,8,18,55,59,75,16}	$x_{56}+1$	577
{14,20,26,45,72,74,43}	x_{16}	581
{26,35,44,48,70,78,27}	$x_{39}+x_{57}$	578
{26,35,44,48,70,78,37}	x_{22}	578
{26,35,44,48,70,78,57}	x_{39}	578
{11,13,21,29,33,76,65}	x_{54}	579
{12,14,24,30,34,77,6}	x_{64}	580
{7,17,28,44,51,64,59}	x_{48}	578
{10,20,31,47,54,67,1}	x_{2}	581
{10,20,31,47,54,67,6}	x_{65}	581
{9,11,25,27,52,57,73,10}	x_{39}	578
{9,11,25,27,52,57,73,28}	$x_{39}+x_{41}+1$	578
{10,12,26,28,53,58,74,0}	x_{55}	579
{11,13,27,29,54,59,75,17}	$x_{15}+1$	580
{24,25,39,58,62,69,79,40}	x_{68}	578
{24,25,39,58,62,69,79,74}	x_{58}	578
{11,12,14,25,30,56,63,29}	x_{60}	577
{5,12,14,19,39,55,74,54}	x_{1}	580
{8,24,27,62,68,72,75,4}	x_{62}	580
{4,16,30,33,38,48,68,39}	x_{17}	579

Table 6: The Extended Cubes on the 576th Output Bit

Combined with Bedi's result in (Bedi and Pillai, 2009), we can improved the final result by applying the Linear Extension Cube Attack to only one output bit. 48 key bits can be directly recovered by cubes with dimension no larger than 8, while the original Cube attack can only recover 36 key bits on the same condition. With the help of 10-dimensional cubes, the original Cube attack can only recover 45 key bits. The following table displays the results in detail:

	Bedi's	Ours'	Bedi'	Ours'
Dimension	≤ 10	≤ 8	≤ 8	≤ 10
Key Bits	45bits	48bits	36bits	≥ 55bits

Table 7: Comparison of Results

Note that, this paper conduct the new method of Cube attack on the 576th output bit with boundary of cube dimension, more output bits and larger dimensional cubes can be implemented in future work and the more than 55 key bits should be recovered.

4.3 Analysis

Linear Extension Cube Attack can improve the complexity in two ways, one is that more key bits can be recovered so that the complexity of brute force attack is improved. The other is that the search scale on higher rounds is narrowed. Instead of walking over all of the $d+1$ dimensional cubes, the attacker only need to search on a relatively smaller scale. Assuming there are m public variables altogether, t maxterms of degree d can be obtained by the original attack, then the search scale on $d+1$ dimension is $O(t\times(m-d))$ instead of $O(C_m^{d+1})$.

5. CONCLUSION

Motivated by the observation in (Mroczkowski and Szmidt, 2012), this paper proposes an improved attack on stream ciphers basing on the original Cube attack, i.e. the Linear Extension Cube Attack, which makes improvement on the pre-processing phase of the original attack and the trade-off between time and space enables the attacker to induce maxterms of higher-order from those of lower-order, thus recovering more key bits and improving the search complexity on higher-dimension. This paper provides Linear Extension Cube Attack against Lili-128 algorithm and two reduced variants of Trivium algorithm. For Lili-128 algorithm, only $O(2^9)$ key streams are needed to recover 88 key bits and the attack has time complexity less than $O(2^{14})$. It is the best attack on Lili-128 to the best of our knowledge. For Trivium algorithm, 48 key bits can be recovered by cubes with dimension no larger than 8 when the initialization round is 576, the results are much better than those of the original attacks.

We also find two interesting phenomena during our experiments. First, the improved Cube attack is applied to Trivium algorithm when the initialization round is 672, 9 key bits can be directly recovered by the improved attack on the 672th output bit, which is 2 more than Dinur and Shamir's attack on the same output bit in (Dinur and Shamir, 2009). However, the performance is not sparkle compared to variants of lower round. Here we propose an open problem about enhancing the performance of the Linear Extension Cube Attack against ciphers with complex initialization.

Second, the algebraic degree of a certain monomial may increase by $a(a\geq 2)$ after 1-round extension. Hence, the extension from d to $d+1$ can also be improved to $d+a$, which extends the attack by adding a new indexes into a cube and searching for linear expressions. Therefore, the improvement of Linear Extension Cube Attack is also considerable in future research.

At last, according to our experiments and results, we should say that our new method of Cube attack is efficient and of certain importance, especially with the application of lightweight cryptography.

REFERENCES

Dinur I, Shamir A, 2009. Cube Attack on Tweakable Black box polynomials. *Advances in Cryptology-EUROCRYPT*. Springer Berlin Heidelberg: 278-299.

Pierre Alain Fouque, Thomas Vannet, 2013. Improving Key Recovery to 784 and 799 rounds of Trivium using Optimized Cube Attacks. *In Fast Software Encryption 2013*. Sourced from http://fse2013.spms.ntu.edu.sg:80.

Zhao, X., Wang, T. and Guo, T. 2011. Improved Side Channel Cube Attacks on PRESENT. Sourced from http://eprint.iacr.org/2011/165.

Aumasson, J. P., Meier W. and Dinur I., 2009. Cube Testers and Key Recovery Attacks on Reduced Round MD6 and Trivium. *Fast Software Encryption 2009*. Springer Berlin Heidelberg: 1-22.

Dinur, I., Shamir, A., 2011. Breaking Grain-128 with Dynamic Cube Attacks. *Fast Software Encryption 2011*. Springer Berlin Heidelberg: 167-187.

Mroczkowski P, Szmidt J, 2012. The Cube Attack on Stream Cipher Trivium and Quadraticity Tests. *Fundamenta Informaticae*. 114(3): 309-318.

Sun, Y. and Wang, Y. 2012. Cube Attack and Its Improvement. *Computer Science*. 39(100): 77-80.

Li, G., Wang, W. and Wang, Y. 2010. Cube Attack on Lili-128 Algorithm. *Advanced Cipher Study*. 2010(2): 46-54.

Bedi, S.S., Pillai, N.R, 2009. Cube Attacks on Trivium. Sourced from http://eprint.iacr.org/2009/015.

Dawson, E., Clark, A. and Golic, J. 2000. The Lili-128 Keystream Generator. *Proceedings of First NESSIE Workshop*.

Saarinen, M-J. O., 2002. A Time-Memory Trade-off Attack Against LILI-128. *Fast Software Encryption 2002*. Springer Berlin Heidelberg: 231-236.

Courtois, N. T. and Meier, W. 2003. Algebraic Attacks on Stream Ciphers with Linear Feedback. *Advances in Cryptology-EUROCRYPT 2003*. Springer Berlin Heidelberg: 345-359.

Courtois, N. T. 2003. Fast Algebraic Attacks on Stream Ciphers with Linear Feedback. *Advances in Cryptology-CRYPTO 2003*. Springer Berlin Heidelberg: 176-194.

Randomness Analysis on Speck Family Of Lightweight Block Cipher

[1]Liyana Chew Nizam Chew [2]Isma Norshahila Mohammad Shah [3]Nik Azura Nik Abdullah [4]Norul Hidayah Ahmad Zawawi [5]Hazlin Abdul Rani [6]Abdul Alif Zakaria
Cyber Technology Research Department
Cyber Security Malaysia
Kuala Lumpur, Malaysia
Email: [1]liyana@cybersecurity.my, [2]isma@cybersecurity.my, [3]azura@cybersecurity.my, [4]norul@cybersecurity.my, [5]hazlin@cybersecurity.my [6]alif@cybersecurity.my

ABSTRACT

Speck family of lightweight block cipher was publicly released by National Security Agency (NSA), USA in June 2013. Speck has been designed with ten instances which provides excellent performance in both hardware and software. Speck is optimized for performance on microcontrollers. This paper will present the result of randomness testing using NIST statistical test suite for SPECK cipher family, which are Speck128/128, Speck128/192, and Speck128/256. Nine data categories are applied to generate the input sequence (either plaintext or key) for each algorithm. Randomness is important for cryptography module to ensure that the cipher is unpredictable before it becomes available. From the analysis conducted, some failures were identified in some data categories.

Keywords: *Speck block cipher, NIST Statistical Test Suite, lightweight cryptography, statistical randomness testing, significance level.*

1. INTRODUCTION

Lightweight cryptography is a new field that applied specifically for highly constrained devices. Among the important design considerations of lightweight cryptography are reduced power consumption, sufficient encryption speed and small chip size. In highly constrained environments, hardware and software efficiency is becoming more important thus making lightweight cryptography an essential ongoing research. The standard usage of block cipher such as AES was deem to be not a right choice for extremely constrained environment.

Speck family of lightweight block ciphers is the algorithm that was introduced in June 2013 by National Security Agency (NSA). Speck family supports a total of ten instances of different block sizes and key sizes. There were several published research papers that discussed the attacks applied on Speck family since it was published (Alkhzaimi and Lauridsen, 2013; Abed *et al.*, 2013). Differential cryptanalysis is one of the attacks that have been applied to Speck family.

This paper will illustrate the randomness test conducted on the output of Speck algorithms. One of the techniques to check the randomness of the algorithm is by using the NIST statistical analysis. Nowadays, random number generator and pseudorandom number generator is important to cryptography since the cryptography sequence should not be guessed by unauthorized people any easier than a brute force. Therefore, it is necessary for an algorithm to be random and unpredictable.

Encryption is a cryptographic operation that is used to provide confidentiality for sensitive information. Several algorithms that were approved for encryption by the Federal government of USA and published in NIST publications are algorithms which have keys sizes larger than 112 bits (Barker and Roginsky, 2011). Therefore this paper will only discuss on Speck algorithms with a large key size. The analysis will focus on the following Speck family algorithm; Speck128/128, Speck128/192 and Speck128/256.

2. A BRIEF DESCRIPTION OF SPECK FAMILY OF BLOCK CIPHER

Speck Family consists of ten instances with difference block sizes and key sizes, each algorithms is applicable in various implementations. The algorithms provide excellent performance in hardware and software, and also optimized performance on microcontroller. Speck Family has a range of block and key sizes to match application requirement and security needs without sacrificing the performance (Beaulieu *et al.*, 2013).

Algorithm	Block size	Key size	Round
Speck128/128	128	128	68
Speck128/192	128	192	69
Speck128/256	128	256	72

Table 1: Block sizes, key sizes and Round of Speck algorithms.

3. ROUND FUNCTIONS OF BLOCK CIPHER

Speck cipher encryption is operated using Feistel network. Left word L_i of the input is rotated by $\alpha = 8$ bits to the left and the output is added with the right word R_i before modulo $2^n = 128$. The left output will be XORed with round key K_i and becomes the left input for next round. The right word is then rotated to the right by $\beta = 3$ bits, then is XORed with left output and the output will become the right input for next round. The process of Speck's round function and key expansion is shown in Figure 1 and Figure 2.

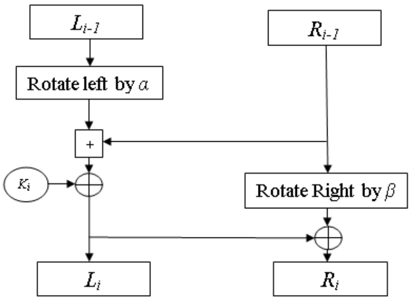


Figure 1: Round Function of SPECK; i steps of encryption.

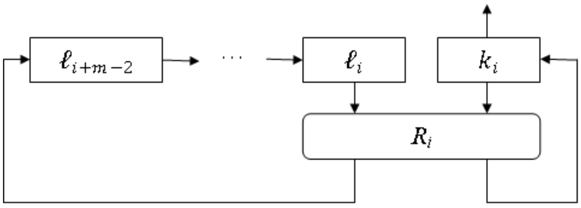


Figure 2: Key Expansion of SPECK.

4. KEY SCHEDULES OF BLOCK CIPHER

Round keys k_i are generated to be used in the round function of Speck. Round keys are written as $(K_0, \ell_0, \dots, \ell_{m-2})$ for a value of m in $\{2, 3,4\}$. The round keys are defined by the following procedure:

$$\ell_{i+m-1} = (k_i + S^{-\alpha}\ell_i) \oplus i$$

$$k_{i+1} = S^{\beta} k_i \oplus \ell_{i++m-1}$$

5. NIST STATISTICAL TEST SUITE

Output sequence of the algorithm can be applied on several statistical tests that attempts to compare and evaluate a random sequence. The properties of randomness of the sequence can be characterized and described in terms of probability (p-value).

Randomness test for the output of the Speck Family will be analyzed under full round consideration. All the randomness testing was based on the application of the NIST Statistical Test Suite that consists of 15 tests. The tests aim to test the randomness of binary sequences produced by either hardware or software based cryptographic random or pseudorandom number generators (Rukhin *et al.*, 2010). Each of these 15 tests is under different parameter input and number of p-values reported by each test listed in Table 2. Each p-value corresponds to an individual statistical test on one sample binary sequence. The tests differentiate into two categories, namely the Parameterized Test and the Non-Parameterized Test. Parameterized Test requires the parameter value(s) of block size, number of blocks and template length as stated in NIST Statistical Test Suite publication to be defined (Barker and Roginsky, 2011). The Non-Parameterized Test does not require any additional parameter in obtaining the p-values for each test. The tests are divided according to their categories as listed in Table 2.

Parameterized Test Selection	
Statistical Test	**No. of P-values**
Block Frequency Test	1
Overlapping Template Test	1
Non-Overlapping Templates Test	148
Serial Test	2
Approximate Entropy Test	1
Linear Complexity Test	1

Non-Parameterized Test Selection	
Statistical Test	**No. of P-values**
Cumulative Sums (Forward/Reverse) Test	2
Runs Test	1
Longest Runs of Ones Test	1
Binary Matrix Rank Test	1
Spectral (Discrete Fourier Transform) Test	1
Random Excursion Test	8
Random Excursion Variant Test	18
Frequency Test	1
Maurer’s Universal Test	1

Table 2: Nine Data Categories and number of p-value

6. DATA CATEGORIES

Inputs to Speck Algorithms are established by nine data categories (Soto, 1999; Abdullah *et al.*. 2011). These data categories have specific function in evaluating the randomness of the algorithm. Each of these data categories will produce 1000 input samples. Sequence length of each sample is depending of the key size or block size of tested algorithm.

Strict Key Avalanche

Data is to examine the algorithm in changing the input parameter (key). All-zero plaintext and random base- key is encrypted as initial ciphertext for the test. For each base-key, the all-zero plaintext is encrypted with one of the flipped-key where a flipped-key is the base-key with flipped bit at the i^{th} bit, for $1 \leq i \leq key\ size$. The ciphertext produce by flipped-key will then be XORed with initial ciphertext to produce a derived block. In order to produce at least 10^6–bit sequence for each sample, derived block will be concatenated by other selected random base-key. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Strict Plaintext Avalanche

Data is to examine the algorithm in changing the input parameter (Plaintext). This test is similar to Strict Key Avalanche but differs at using plaintext as changing the changed parameter. All-zero key and random base-plaintext is encrypted as initial ciphertext for the test. For each base-plaintext, the all-zero key is encrypted with one of the flipped- plaintext where a flipped- plaintext is the base- plaintext with flipped bit at the i^{th} bit, for $1 \leq i \leq plaintext\ size$. The ciphertext produced by the flipped- plaintext will then be XORed with initial ciphertext to produce a derived block. In order to produce at least 10^6–bit sequence for each sample, derived block will be concatenated by other selected random base- plaintext. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Plaintext/Ciphertext Correlation

Data is to examine the correlation between plaintext/ciphertext pairs. For each sample, one chosen random key and adequate block of random plaintext are choose to produce at least 10^6-bit sequence. To generate a derived block, each plaintext block will be encrypted using the one random key and then the ciphertext will be XORed with each plaintext block. These derived blocks are computed in ECB mode and are then concatenated. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Ciphertext Block Chaining Block

Given a random key, an initialization vector (IV) of all zeroes, and a plaintext of all-zero, a sequence of at least 10^6-bits was constructed in CBC mode. The first ciphertext block (CT_1) is defined by $CT_1 = E_k(IV \oplus PT_0)$. Subsequent ciphertext blocks are defined by $CT_i + 1 = EK(CT_i \oplus PT_i)$ for $1 \leq i \leq derived\ block$. All 1000 sequences were generated, each with a different random key. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Random Plaintext / Random Key

Data is to examine the randomness of ciphertext based on random plaintext and random key. For each sample, one chosen random key and adequate blocks of random plaintext are choosen to produce at least 10^6-bit sequence using ECB mode. All 1000 sequences were generated, each with a different random key. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Low Density Plaintext

This data type is formed based on low density plaintext blocks which is all zero plaintext block, plaintext blocks of all zero with a single bit of ‘1’ and plaintext blocks of zeroes with two bits of ‘1’ in each combination of two bit positions within the plaintext size of positions. These entire plaintext blocks are encrypted using ECB mode with one random key. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

High Density Plaintext

This data type is formed based on high density plaintext blocks which is all ‘1’ plaintext block, plaintext blocks of all ‘1’ with a single bit of zero and plaintext blocks of all ‘1’ with two bits of zero in each combination of two bit positions within the plaintext size of positions. These entire plaintext blocks are encrypted using ECB mode with one random key. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

Low Density Keys

This data type is formed based on low density keys blocks which is all zero keys block, keys blocks of all zero with a single bit of ‘1’ and keys blocks of zeroes with two bits of ‘1’ in each combination of two bit positions within the keys size of positions. These entire plaintext blocks are encrypted using ECB mode with one random plaintext. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

High Density Keys

This data type is formed based on high density keys blocks which is all ‘1’ keys block, keys blocks of all ‘1’ with a single bit of zero and keys blocks of all ‘1’ with two bits of zero in each combination of two bit positions within the keys size of positions. These entire plaintext blocks are encrypted using ECB mode with one random plaintext. The number of derived blocks and derived sequence length for each sample of Speck family are listed in Table 3.

	Speck128/128	**Speck128/192**	**Speck128/256**
	Strict Key Avalanche		
Number of Base-Key block	62	41	31
Derived blocks	7936	7872	7936
Sequence length	1015808	1007616	1015808
	Strict Plaintext Avalanche		
Number of Base-Plaintext block	62	62	62
Derived blocks	7936	7936	7936
Sequence length	1015808	1015808	1015808
	Low Density Plaintext and High Density Plaintext		
Derived blocks	8257	8257	8257
Sequence length	1056896	1056896	1056896
	Low Density Key and High Density Key		
Derived blocks	8257	18527	32897
Sequence length	1056896	2371712	4210816

Table 3: Number of Derived Blocks and Sequence Length for Three Speck Families Algorithm.

7. NIST TESTING EXPERIMENTAL SETUP

NIST test was performed using the following approach:

(a) Input parameters for 15 tests such as the sequence length, sample size, and significance level were fixed for each sample. The sample size is corresponding to the choice of the significance level. The significance level was set to 0.001 and the sample size is 1000 sequence (Barker and

Roginsky, 2011). For each binary sequence and each statistical test, a p-value was reported. Parameters of Parameterized Test for Speck128/128, Speck128/192 and Speck128/256 are shown in Table 4.

(b) Inputs for all Speck algorithms are generated using nine different data categories. Each of these nine data categories will produce a sequence with 1000 input samples. Sequence length of each data categories is depending on block size and key size of the algorithm to be tested. The sequence length for each data categories and algorithm are as shown in table 3.

(c) The success or failure assessment on each p-value is based on whether or not it exceeded or fell below the selected significance level which is 0.001. For each statistical test and each sample, a sample was considered to have passed a statistical test if p-value for this sample is equal or greater than 0.001. If the p-value fell below 0.001, then the sample was flagged as failure.

(d) The maximum number of sequence that was expected to be rejected must be computed by using the following formula (Soto, J. 1999, Zawawi *et al.*. 2013). Since the experiment used 1000 samples, therefore, the calculation of the maximum rejection will be based on 1000 samples.

$$s\left(\alpha + 3\sqrt{\frac{\alpha(1-\alpha)}{s}}\right)$$

Where s represents the sample size used and α represents the significance level used.

In order to explain the parameters that were used in the tests, the following abbreviation is used: block length (M), sequence length (n), non-overlapping blocks ($N = n/M$), template length (m), theoretical probabilities (π_i) and number of block in the initialization sequence (Q).

The requirements for Parameterized Test are as per describe as below:

- Block Frequency test: M is selected such that $n \geq MN$,$M \geq 20$,$M \geq 0.01n$ and $N < 100$.
- Non-Overlapping Template test: $N = 8$ has been specified, m is recommended that $m = 9$ or $m = 10$, $N \leq 100$ and $M > 0.01n$.
- Overlapping Template test: m is recommended that $m = 9$ or $m = 10$, $n \geq MN$, $N(\min \pi_i)$, $\lambda = (Mm + 1)/2m \approx 2$, $m \approx \log_2 M$ and $K \approx 2\lambda$.
- Linear Complexity test: the value of M must be in the range of $500 \leq M \leq 5000$ and $N \geq 200$.
- Serial and Approximate Entropy tests: m and n chosen such that $m < [\log_2 n] - 2$.

Based on the requirement stated, the NIST parameter input for Speck128/128, Speck128/192 and Speck128/256 for all nine data categories are as shown in Table 4. Speck128/128, Speck128/192 and Speck128/256 use the same input for parameterized test except for some data categories in block frequency test. Parameter input for Block Frequency Test of Speck128/128 is 20000 and this parameter value applied for all nine data categories for Speck 128/128. NIST parameter input of Block Frequency Test for Speck128/192 for Low Density Keys and High Density Keys is 30000, and parameter input for other data categories is 20000. NIST parameter input of Block Frequency Test for Speck128/256 for Low Density Keys and High Density Keys is 45000, and the remaining data categories use 20000 as parameter input for block frequency test.

<table>
<tr><th colspan="4">Input for Parameterized Test</th></tr>
<tr><td rowspan="2">Block Frequency Test</td><td>1</td><td>2</td><td>3</td></tr>
<tr><td>20000</td><td>30000</td><td>45000</td></tr>
<tr><td>Overlapping Template Test</td><td colspan="3">10</td></tr>
<tr><td>Non-Overlapping Templates Test</td><td colspan="3">10</td></tr>
</table>

Serial Test	2
Approximate Entropy Test	2
Linear Complexity Test	2000

Table 4: Input for Parameterized Test

8. RESULTS AND ANALYSIS

The three chosen Speck algorithms namely Speck128/128, Speck128/192 and Speck 128/256 are tested under nine data categories with each having 1000 samples. For each experiment, the significance level was fixed at 0.001. The maximum rejection is calculated using formula that has been discussed earlier. The description of each test that exceeded the maximum number of rejection for each algorithm is shown in Table 5a, Table 5b and Table 5c. For example, data categories of low density plaintext for Speck 128/128 has exceed the maximum number of rejection for runs test, 5 out of 1000 samples are rejected.

Speck128/128			
Data Categories	**Statistical Test**	**Number of rejection**	**Maximum Number of Rejection**
Low Density Plaintext	Runs Test	5	3
Strict Key Avalanche	Overlapping Template Test	10	3
Random Plaintext/Random Keys	Serial (p-value1)	4	3
Low Density Plaintext	Serial (p-value2)	4	3

Table 5a: Result for each statistical test that exceeded the maximum number of rejection for Speck128/128.

Speck128/192			
Data Categories	**Statistical Test**	**Number of rejection**	**Maximum Number of Rejection**
Strict Key Avalanche	Random Excursion Variant Test	28	22
Strict Plaintext Avalanche	Random Excursion Variant Test	28	22
Random Plaintext/Random Keys	Random Excursion Variant Test	27	21
High Density Plaintext	Longest Runs of Ones Test	4	3
Cipher Block Chaining Mode	Linear Complexity Test	4	3
Low Density Keys	Overlapping Template Test	6	3
Low Density Plaintext	Approximate Entropy Test	4	3

Table 5b: Result for each statistical test that exceeded the maximum number of rejection for Speck128/192.

Speck128/256			
Data Categories	**Statistical Test**	**Number of rejection**	**Maximum Number of Rejection**
Strict Key Avalanche	Random Excursion Variant Test	22	21
Plaintext/Ciphertext Correlation	Random Excursion Test	13	11
High Density Keys	Spectral (Discrete Fourier Transform) Test	10	3
Strict Plaintext Avalanche	Maurer's Universal Test	4	3
Low Density Plaintext	Non-Overlapping Templates Test	214	184
Low Density Plaintext	Maurer's Universal Test	4	3
High Density Plaintext	Non-Overlapping Templates Test	204	184

Table 5c: Result for each statistical test that exceeded the maximum number of rejection for Speck128/256.

9. CONCLUSION

This paper has presented the statistical analysis on Speck Family algorithms which specifically focuses on Speck128/128, Speck128/192 and Speck128/256. The statistical analysis is using NIST Statistical Test Suite. During the analysis process, the significance level was set at 0.001 to determine whether the algorithm tested is random or non-random. At least one statistical test has exceeded the maximum number of rejection for each algorithm. Speck128/128, Speck128/192 and Speck128/256 have applied 135 tests for nine data categories. Speck 128/128 failed 4 tests, Speck128/256 failed 7 tests and Speck128/256 failed 7 tests.

REFERENCES

Beaulieu, R, Shors, D, Smith, J, Treatman-Clark, S, Weeks, B and Wingers, L. 2013. *The SIMON and SPECK Families of Lightweight Block Ciphers*. Cryptology ePrint Archive, Report 2013/404. Sourced from http://eprint.iacr.org/.

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. 2010. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. NIST SpecialPublication 800-22.

Alkhzaimi, H.A and Lauridsen, M.M. 2013. *Cryptanalysis of the SIMON Family of Block Ciphers*. Cryptology ePrint Archive: Report 2013/543. Sourced from https://eprint.iacr.org/2013/543

Abed, F, Eik List, Lucks, S and Wenzel, J. 2013. *Cryptanalysis of the Speck Family of Block Ciphers*. Cryptology ePrint Archive: Report 2013/568. Sourced from http://eprint.iacr.org/2013/568

Soto, J. 1999. *Randomness Testing of the AES Candidate Algorithms*. Sourced from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.39.21

Barker, E and Roginsky, A. 2011.Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths. NIST Special Publication 800-131A.

Abdullah, N.A.N, Zawawi, N.H L.A. and Rani, H.A. 2011. *Analysis on Lightweight Block Cipher, KTANTAN*. Information Assurance and Security (IAS), 2011 7th International Conference. page 46-51. IEEE.

Zawawi, N.H L.A, Seman,K and Zaizi, N.J.M. 2013. *Randomness analysis on grain - 128 stream cipher*. AIP Conference Proceedings 1557, 15 (2013).

Compression-Crypto Technique: Smaller Size of the Compressed Plaintext

[1]Arif Mandangan, [2]Loh Chai Mei, [3]Chang Ee Hung and
[4]Che Haziqah Che Hussin
[1,2,3]School of Science and Technology, Universiti Malaysia Sabah,
Jalan UMS, 88400 Kota Kinabalu Sabah Malaysia,
[4]Preparatory Centre for Science and Technology,
Universiti Malaysia Sabah,
Jalan UMS, 88400 Kota Kinabalu Sabah Malaysia.
Email: [1]arifman@ums.edu.my, [2]christineloh_90@hotmail.com, [3]fancy_2309@hotmail.com,
[4]haziqah@ ums.edu.my

ABSTRACT

Key distribution problem has become a major problem in symmetric cryptosystem. Fortunately, this problem has been solved by the emergence of public key cryptosystem. In order to provide good level of security, we need to use large numbers to implement the asymmetric cryptosystem. As a consequence, the efficiency of the asymmetric cryptosystem becomes a big problem in real life application especially to embed asymmetric cryptosystem in small gadgets such as smart phones, tablets and netbooks. RSA, El-Gamal and Elliptic Curve cryptosystems are the most established asymmetric cryptosystem. By using appropriate parameters and keys, these cryptosystem are able to provide good level of security. Hence, we would like to find a method to enhance the efficiency of these cryptosystem without altering the system itself in order to store their security ability. One of the ways to achieve this goal is by reducing the numbers of plaintext-ciphertext to be encrypted and decrypted. We introduced the Compression-Crypto Technique which is able to reduce from k plaintext, where $k \in \mathbb{Z}^+$ and $k > 2$, to only 2 plaintext. We observed that large numbers of plaintext will produce new plaintext with bigger size compared to the original plaintext. Thus, our main concern is that this problem will minimize the enhancement in encryption and decryption procedure. Therefore, in this paper we embed a new method into the Compression-Crypto technique in order to produce a new pair of plaintext with smaller sizes.

Keywords: RSA cryptosystem, compression-crypto technique, continued fraction.

1. INTRODUCTION

In these present days, the Internet is the main platform for communication across the globe and it has indirectly changed how we live every day. The ways we socialize, play, do our shopping and study have been changed by the emergence of the Intern*et al.*most everybody in this world is connected to the Internet via various mediums and devices such as smart phones, tablets, netbooks, notebooks, desktops and so on. As an open communication medium, the Internet is faced with some security problems such as confidentiality, integrity, repudiation and authentication (Farouzan, 2008). Therefore, the network security has become crucial and essential. Basically, network security is a set of protocols that is able to minimize security attacks in order to allow us to use the Internet comfortably. The most common tool to provide network security is cryptography.

Cryptography is a study about secret writing in order to provide confidentiality of which an important essence to a secured network. In cryptography, an original message is called plaintext. The transformed plaintext is called ciphertext. The transformation procedure is called encryption and a key is needed in this procedure. Decryption is the inverse of encryption and this procedure also needs a key. In symmetric cryptosystem, a common key will be shared by Alice (message sender) and Bob (message recipient) and used in encryption-decryption procedures. On the contrary, two different keys are used by Alice and Bob to communicate using asymmetric cryptosystem (Hoffstein *et al.*, 2008).

Due to practicality, asymmetric cryptosystem is currently the most preferred cryptosystem. To cope with today's needs, we are in dire need not only for a secured cryptosystem but also a system which is efficient enough to be embedded into small gadget. This explains why there are rigorous research in cryptography to enhance the efficiency of asymmetric cryptosystem. In (Chang and Mandangan, 2013), we introduced a technique which is able to reduce the number of plaintext from any numbers to only two plaintext. This technique, as we named it, is known as Compression-RSA since our first try on this technique was by embedding it into RSA cryptosystem (Rivest *et al.*, 1978).

After further research on this technique, it is found that this technique can be easily embedded into any asymmetric cryptosystem without altering its key generation, encryption and decryption algorithms. Consequently, we renamed it as the Compression-Crypto technique. Instead of encrypting large numbers of plaintext, an asymmetric cryptosystem only needs to encrypt two plaintext to produce two ciphertext by applying this technique. By decrypting these ciphertext and then applying the inverse of Compression-Crypto technique, we will get the actual and original plaintext without any alteration. In (Mandangan *et al.*, 2014), we observed that the number of original plaintext has a linear relationship with the sizes of each compressed plaintext. As the number of original plaintext increases, the sizes of the compressed plaintext M_1 and M_2 will also increase linearly.

In this paper, we did some modifications on the Compression-Crypto technique so that the compressed plaintext have smaller size compared to the compressed plaintext of those produced by the early designed Compression-Crypto technique. Before further discussion, we firstly introduced the Compression-Crypto technique. Then, we showed the modification done to the technique and finally we presented some examples to compare the size of plaintext produced by the old modified version of Compression-Crypto technique.

2. THE COMPRESSION-CRYPTO TECHNIQUE

Let the set of original plaintext as $\{m_1, m_2, m_3, \cdots, m_{k-1}, m_k\}$ where $k \in \mathbb{Z}^+$and $k > 2$. By using the Compression-Crypto technique, these k plaintext can be compressed to only 2 plaintext, denoted as $\{M_1, M_2\}$. No matter how big the value k is, the plaintext will be reduced to only 2 plaintext M_1 and M_2. The Compression-Crypto technique is basically designed by combining two methods namely Continued Fraction and Extended Euclidean Algorithm.

The algorithm of Compression-Crypto technique (Chang and Mandangan, 2013):

i. Compression procedure

Step 1: Let the set of original k plaintext as

$$\{m_1, m_2, m_3, \cdots, m_{k-1}, m_k\}$$

Step 2: By using Continued Fraction method, compute the new plaintext M_1 and M_2 as follows

$$m_1 + \cfrac{1}{m_2 + \cfrac{1}{m_3 + \cfrac{1}{\begin{matrix}\vdots \\ m_{k-1} + \cfrac{1}{m_k}\end{matrix}}}} = \frac{M_1}{M_2}$$

ii. Decompression procedure

By using Euclidean algorithm, compute the following

$$\begin{aligned} M_1 &= M_2(q_1) + r_1 \\ M_2 &= r_1(q_2) + r_2 \\ r_1 &= r_2(q_3) + r_3 \\ &\vdots \\ r_{k-3} &= r_{k-2}(q_{k-1}) + r_{k-1} \end{aligned}$$

$$r_{k-2} = r_{k-1}(q_k) + r_k$$

where M_1, M_2 are the compressed plaintext, q_i is quotient and r_i is remainder for $i = 1,2,\ldots,k$. From this step, we have

$$\{q_1, q_2, q_3, \cdots, q_{k-1}, q_k\} = \{m_1, m_2, m_3, \cdots, m_{k-1}, m_k\}$$

which is the set of original plaintext.

To show the implementation of the Compression-Crypto technique, we embed the technique into RSA cryptosystem. The implementation is shown in the Figure 1.

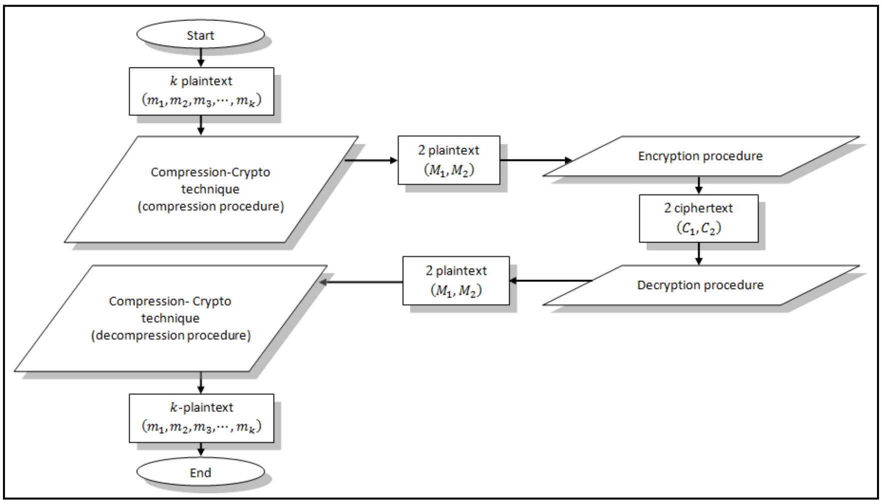

Start
k plaintext
$(m_1, m_2, m_3, \cdots, m_k)$
Compression-Crypto technique (compression procedure)
2 plaintext
(M_1, M_2)
Encryption procedure
2 ciphertext
(C_1, C_2)
Decryption procedure
2 plaintext
(M_1, M_2)
Compression- Crypto technique (decompression procedure)
k-plaintext
$(m_1, m_2, m_3, \cdots, m_k)$
End

Figure 1: This flow chart shows the implementation of Compression-RSA technique in RSA cryptosystem

Example 1: Let $\{1,4,8,6,3,7,5,9,2\}$ be the plaintext, public key set is $\{n = 3079471, e = 443\}$ and the decryption key is $d = 62483$.
Set the plaintext$\{1,4,8,6,3,7,5,9,2\} = \{m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9\}$. Compress the 9-plaintext $\{1,4,8,6,3,7,5,9,2\}$ to produce only 2 new plaintext M_1 and M_2 as follows

$$1 + \cfrac{1}{4 + \cfrac{1}{8 + \cfrac{1}{6 + \cfrac{1}{3 + \cfrac{1}{7 + \cfrac{1}{5 + \cfrac{1}{9 + \cfrac{1}{2}}}}}}}} = \frac{578559}{465616} = \frac{M_1}{M_2}$$

Encrypt the plaintext $M_1 = 578559$ and $M_2 = 465616$ as follows

$$C_1 = 578559^{443} (\text{mod } 3079471) = 2662637$$
$$C_2 = 465616^{443} (\text{mod } 3079471) = 1682050$$

To recover the original plaintext, firstly we need to decrypt the ciphertext $C_1 = 2662637$ and $C_2 = 1682050$ as follows

$$M_1 = 2662637^{62483} (\text{mod } 3079471) = 578559$$
$$M_2 = 1682050^{62483} (\text{mod } 3079471) = 465616$$

Next is to reveal the original plaintext by reversing the Compression-Crypto technique as follows

$$\begin{aligned}
&578559 = 465616(1) + 112943, && q_1 = 1\\
&465616 = 112943(4) + 13844, && q_2 = 4\\
&112943 = 13844(8) + 2191, && q_3 = 8\\
&13844 = 2191(6) + 698, && q_4 = 6\\
&2191 = 698(3) + 97, && q_5 = 3\\
&698 = 97(7) + 19, && q_6 = 7\\
&97 = 19(5) + 2, && q_7 = 5\\
&19 = 2(9) + 1, && q_8 = 9\\
&2 = 1(2) + 0, && q_9 = 2
\end{aligned}$$

Now we have set of quotients $\{q_1, q_2, q_3, \cdots, q_9\} = \{1,4,8,6,3,7,5,9,2\}$ is exactly same with the set of the original plaintext $\{m_1, m_2, m_3, \cdots, m_9\} = \{1,4,8,6,3,7,5,9,2\}$.

3. MODIFIED COMPRESSION-CRYPTO TECHNIQUE

Let the original plaintext as $\{m_1, m_2, m_3, \cdots, m_{k-1}, m_k\}$. We set $d_1 = m_1$. Then, for all $i = 2,3,\dots,k$ we compute

$$d_i = m_i - m_{i-1} \tag{1}$$

Now we have a new set of plaintext $\{d_1, d_2, d_3, \cdots, d_{k-1}, d_k\}$ where each of this plaintext is smaller than its corresponding original plaintext except the first plaintext where $m_1 = d_1$. That is, we have

$$d_i < m_i$$

for all $i = 2,3,\dots,k$. This reduction leads to the production of smaller size of the compressed plaintext M_1 and M_2. To recover the original plaintext, compute

$$m_i = d_i + m_{i-1} \tag{2}$$

for all $i = 2,3,\dots,k$.

Example 2: Suppose we have 9 plaintext $m = \{1, 4, 8, 6, 3, 7, 5, 9, 2\}$. Then, by applying equation (1), we have

$$\begin{array}{lll}
d_1 = m_1 = 1 & d_4 = m_4 - m_3 = 6 - 8 = -2 & d_7 = m_7 - m_6 = 5 - 7 = -2\\
d_2 = m_2 - m_1 = 4 - 1 = 3 & d_5 = m_5 - m_4 = 3 - 6 = -3 & d_8 = m_8 - m_7 = 9 - 5 = 4\\
d_3 = m_3 - m_2 = 8 - 4 = 4 & d_6 = m_6 - m_5 = 7 - 3 = 4 & d_9 = m_9 - m_8 = 2 - 9 = -7
\end{array}$$

Now, we have a new set of plaintext $d = \{1, 3, 4, -2, -3, 4, -2, 4, -7\}$. To recover the original plaintext m, we apply equation (2) as follows

$$\begin{array}{lll}
m_1 = d_1 = 1 & m_4 = d_4 + m_3 = -2 + 8 = 6 & m_7 = d_7 + m_6 = -2 + 7 = 5\\
m_2 = d_2 + m_1 = 3 + 1 = 4 & m_5 = d_5 + m_4 = -3 + 6 = 3 & m_8 = d_8 + m_7 = 4 + 5 = 9\\
m_3 = d_3 + m_2 = 4 + 4 = 8 & m_6 = d_6 + m_5 = 4 + 3 = 7 & m_9 = d_9 + m_8 = -7 + 9 = 2
\end{array}$$

These calculations only involve simple addition and subtraction operations which can be done in short time. For further discussion, we denote the original 9 plaintext as $m = \{1, 4, 8, 6, 3, 7, 5, 9, 2\}$ and the new 9 plaintext as $d = \{1, 3, 4, -2, -3, 4, -2, 4, -7\}$. By applying the Compression-Crypto Technique, we compress both sets of plaintext as follows:

a) For $m = \{1, 4, 8, 6, 3, 7, 5, 9, 2\}$,

$$1+\cfrac{1}{4+\cfrac{1}{8+\cfrac{1}{6+\cfrac{1}{3+\cfrac{1}{7+\cfrac{1}{5\cfrac{1}{9+\cfrac{1}{2}}}}}}}}=\frac{578559}{465616}=\frac{M_1}{M_2}$$

b) For $d=\{1,3,4,-2,-3,4,-2,4,-7\}$

$$1+\cfrac{1}{3+\cfrac{1}{4+\cfrac{1}{-2+\cfrac{1}{-3+\cfrac{1}{4+\cfrac{1}{-2\cfrac{1}{4+\cfrac{1}{-7}}}}}}}}=\frac{15817}{12121}=\frac{D_1}{D_2}$$

The size of M_1 is 20 bits and M_2 is 19 bits. On the other hand, the size of both D_1 and D_2 is 14 bits. In the given example, there is about 25% size reduction of the compressed plaintext by applying this method. Of course encrypting D_1 and D_2 will be faster than encrypting M_1 and M_2 due to their smaller size.

4. CONCLUSION AND DISCUSSION

Some modifications on the Compression-Crypto technique were done in this paper. We reduced the size of the compressed plaintext M_1 and M_2 to smaller size in order to enhance the performance of encryption and decryption procedures. Some experiments could be done in further research to find the actual percentage of the size reduction of the compressed plaintext especially when we deal with large numbers of original plaintext. Also, we did several attempts to embed the modified Compression-Crypto technique into other asymmetric cryptosystem such as El-Gamal and Elliptic Curve Cryptography.

5. ACKNOWLEDGMENTS

We would like to thank Universiti Malaysia Sabah for supporting our participation in CRYPTOLOGY 2014. This research is supported by *Research Acculturation Grant Scheme* (*RAGS*) RAG0001-SG-2012 MOHE (Malaysia).

REFERENCES

Chang, E. H. and Mandangan, A. 2013. Compression-RSA: New approach of encryption and decryption method. *AIP Conference Proceeding 1522, American Institute of Physics*:50-54.

Farouzan, B. A. 2008. *Introduction to Cryptography and Network Security*, New York: McGraw-Hill Companies: 20-28.

Hoffstein, J. , Pipher, J. and Silverman, J. H. 2008. *An Introduction to Mathematical Cryptography,* New York*:* Springer Science +Bussiness Media, 37-39.

Mandangan, A., Loh, C. M., Chang, E. H. and Hussin, C. H. C. 2014. Compression-RSA Technique: A More Efficient Encryption- Decryption Procedure. *To be published in The 3rd International Conference of Mathematical Sciences Conference 2014 Proceeding*.

Rivest R. L., Shamir, A. and Adleman, L. 1978. A Method for Obtaining Digital Signature and Public Key Cryptosystem, Commun. ACM21:120-126

A Construction of Secret Sharing Scheme

[1]Goh Y. L. and [2]Denis C. K. Wong
[1, 2]Department of Mathematical and Actuarial Sciences, University of Tunku Abdul Rahman
Jalan Genting Kelang, 53300 Setapak, Kuala Lumpur, Malaysia
Email: [1]gohyl@utar.edu.my, [2]deniswong@utar.edu.my

ABSTRACT

In this paper, the use of binary linear codes to construct the access structures of a secret sharing scheme is illustrated. The relationship between the minimal codewords of a linear code and the minimal access structure are shown. Finally, we generalize the construction of such scheme to various algebraic group codes.

Keywords: Secret sharing scheme, linear codes, group algebra

1. INTRODUCTION

To keep the secret efficiently and safely, Shamir (1979) developed the concept of secret sharing scheme which is a rapidly developed field in cryptography. One of the famous secret sharing scheme is constructed by Shamir, the (n, k) −theshold secret sharing scheme over $\boldsymbol{F}_q$, which is defined as follows: A secret $s \in \boldsymbol{F}_q$ is split into n shares $s_i \in \boldsymbol{F}_q$ for $i = 1,2,\cdots,n$ in such a way that any k shares uniquely determine the secret but any $k - 1$ or fewer shares provide no information about the secret. McEliece and Sarwate(1981) improved the (n, k) −theshold secret sharing scheme by introducing the following secret sharing scheme based on linear code:

> First, choose a $[n, k, n - k + 1] -$ linear $\boldsymbol{MDS}$ code C over $\boldsymbol{F}_q$. The secret is chosen as the first digit of a codeword $v \in C$. The next $k - 1$ digits are chosen uniformly at random over $\boldsymbol{F}_q$ and the codeword v then computed. The $n - 1$ shares are all the digits in v after the first. The threshold is k because the digits in any k positions of a codeword in an MDS code uniquely determine the full codeword, that is, any k positions are an information set.

In a more general setting, a secret sharing scheme involved a dealer, denoted by D, who is responsible for selecting a secret k, and then computing the shares s_i from the secret using some systematical algorithm. Other participants form a set P, who will share the secret. Furthermore, let $\tau \subseteq P$ where τ can determine the secret. τ is called the access structure and any subset of τ are called access sets.

Recently, many researchers have constructed secret sharing scheme by using linear codes as the theory of algebraic coding theory have been systematically developed (Ashikhmin and Barg 1998; Ding, Kohel and Ling 2000; Li, Xue and Lai 2010; Massey 1993; Yuan and Ding 2006). Algebraic coding theory is important in modern digital communication; however noises might occur during the transmission of digital data across a communication channel. This may cause the received data to differ from the transmitted data. Therefore, error correcting and detecting codes are used in modern digital communication system. The study of group codes as an ideal in a group algebra FG has been developed long time ago (Berman 1967; Berman 1989). In 1993, Massey has shown a nice relationship between the access structure and the minimal codewords of the dual code of the underlying code Massey (1993).

In this paper, a method to construct secret sharing scheme is proposed by using group algebra codes defined over various groups. The paper is organized into four sections. Section 2 introduces the group algebra codes. In section 3, the implementation of secret sharing scheme via group algebra codes are discussed. Finally, some remarks are given in the conclusion section.

2. SECRET SHARING SCHEME BASED ON LINEAR CODES

Before we start with the construction of secret sharing scheme, we first recall some well-known definitions of error correcting code. A q–ary**[*n,k,d*] – *linear code*** C is a subspace of $\boldsymbol{F}_q^n$ and a ***generator matrix*** of C is a $k \times n$ matrix where all rows of G form a basis for C. Any element of C is called a codeword of C. Furthermore, the $n \times (n-k)$ matrix where all columns of H form a basis for $C^{\perp}$ is called the ***parity check matrix*** of C. The ***support*** of $v \in F_q^n$ is defined as the set $\{0 \leq i \leq n-1, v_i \neq 0\}$, and say that $v_1 \in F_q^n$ cover $v_2 \in F_q^n$ provided $supp(v_2) \subseteq supp(v_1)$. An element $v \neq 0$ is ***minimal*** if it covers its scalar multiples. Furthermore, a codeword whose first component is 1 and only covers its scalar multiples is called a ***minimal codeword***. Clearly, every minimal codeword is a minimal vector. In this section, we show a construction of secret sharing scheme by using the matrices G and H. Massey (1993) points out a nice relationship between minimal codewords and minimum access structure of a secret sharing scheme. We will illustrate this relationship and hence refine the result obtained by Massey.

Construction 1: Secret sharing scheme based on a binary [7,4,3] – linear code.

Let C be a [7,4,3] – binary linear code with the following generator matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Any codeword v in C can be written uniquely in the form $v = (v_1, v_2, v_3, v_4, v_1 + v_3 + v_4, v_1 + v_2 + v_4, v_2 + v_3 + v_4)$, where $(v_1, v_2, v_3, v_4) \in \boldsymbol{F}_2^4$ is the corresponding message word. The parity check matrix of C is

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

It is well known that $v' \in C$ if and only if $v'H = (0,0,0) \in \mathcal{F}_2^3$. Suppose we let $v' = (a, b, c, d, e, f, g)$. Then, the condition $v'H = (0,0,0)$ produces the following system of equations:

$$a + c + d + e = 0 \qquad a + b + d + f = 0 \qquad b + c + d + g = 0 \qquad (1)\ (2)\ (3)$$

From equation (1), we have

$$a = c + d + e \tag{4.1}$$

From equation (2), we have

$$a = b + d + f. \tag{4.2}$$

By adding equations (1) and (3), we obtain

$$a = b + e + g \tag{4.3}$$

Finally, by adding equations (2) and (3), we obtain

$$a = c + f + g \tag{4.4}$$

Next, we setup the correspondence between each digit in a codeword, and the secret with distributions to participants as shown in Table 1. From equations (4.1) to (4.4), we see that the access structure for the secret sharing scheme with the above correspondence based on $C$are $\{P_2, P_3, P_4\}$, $\{P_1, P_3, P_5\}$, $\{P_1, P_4, P_6\}$ and $\{P_2, P_5, P_6\}$. Now, we consider the [7, 3, 4] – binary dual code $C^{\perp}$ of C which has the following generator matrix

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

All codewords of $C^{\perp}$ are listed in Table 2.

Digit in codeword	Secret and participants
v_1	Secret
v_2	P_1
v_3	P_2
v_4	P_3
v_5	P_4
v_6	P_5
v_7	P_6

Table 1: Correspondence between digits in codeword, secret and participants

$\mathcal{F}_2^3$	$C^{\perp}$
000	0000000
100	1011100
010	1101010
001	0111001
110	0110110
101	1100101
011	1010011
111	0001111

Table 2: Codewords in the [7, 3, 4] – binary dual code

By inspecting through each codeword in $C^{\perp}$ and compare the access structures obtained above, we see that all possible access structures are corresponding to the minimal codeword in $C^{\perp}$ with a "1" in the first position as shown in Table 3.

Minimal codewords in $C^{\perp}$	⟶	Access structure based on C
1011100		$\{P_2, P_3, P_4\}$
1101010		$\{P_1, P_3, P_5\}$
1100101		$\{P_1, P_4, P_6\}$
1010011		$\{P_2, P_5, P_6\}$

Table 3: Correspondence between minimal codewords and access structure

Construction 2: Secret sharing scheme based on a binary [9,5,3] – linear code.

Let consider the [9,5,3] – binary code C with the following generator matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

The corresponding parity check matrix for C is

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

All codewords in $C^{\perp}$ are listed in the Table 4. For all $v = (v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9) \in C$, if we setup the correspondence as shown in Table 5, it follows that the access structure of the secret sharing scheme based on C are $A_1 = \{P_1, P_4, P_5\}$, $A_2 = \{P_1, P_3, P_6\}$, $A_3 = \{P_2, P_3, P_7\}$, $A_4 = \{P_2, P_4, P_8\}$, $A_5 = \{P_2, P_4, P_{5,} P_6, P_7\}$,$A_6 = \{P_2, P_3, P_{5,} P_6, P_8\}$,$A_7 = \{P_1, P_3, P_{5,} P_7, P_8\}$,and$A_8 = \{P_1, P_4, P_{6,} P_7, P_8\}$. Therefore, the minimal access structures are A_1, A_2, A_3 and A_4.

$\boldsymbol{F}_2^4$	$C^{\perp}$
0000	000000000
1000	110011000
0100	110100100
0010	101100010
0001	101010001
1100	000111100
1010	011111010
1001	011001001
0110	011010110
0101	011110101
0011	000110011
1110	101011110
1101	101101101
1011	110101011
0111	110010111
1111	000001111

Table 4: All codewords in [9, 4, 4] – binary dual code of C

Digit in codeword	Secret and participants
v_1	Secret
v_2	P_1
v_3	P_2
v_4	P_3
v_5	P_4
v_6	P_5
v_7	P_6
v_8	P_7
v_9	P_8

Table 5: Correspondence between digits in codeword, secret and participants

In general, let $A = \{P_{i_1}, P_{i_2}, \cdots, P_{i_m}\}$ be a minimal access set of the secret sharing scheme based on a [*n*,*k*,*d*] – linear code C.

Suppose the columns $g_{i_1}, \cdots, g_{i_m}$ of the generator matrix *G* of C are linear dependent. Then, we have $\sum_{k=1}^{m} \lambda_k g_{i_k} = 0$, where not all λ_j are 0. Without loss of generality, we may assume $\lambda_1 \neq 0$. Thus, we have $g_{i_1} = \sum_{k=2}^{m} \frac{\lambda_k}{\lambda_1} g_{i_k}$. Therefore, the participants $\{P_{i_2}, \cdots, P_{i_m}\}$ can learn the share of P_{i_1} by combining their shares and hence they can recover the secret which is a contradiction. Hence, we known that the columns $g_{i_1}, \cdots, g_{i_m}$of G are linear independent. Then, there exist a codeword $a = (1,0, \cdots,0, a_{i_1}, 0, \cdots,0, a_{i_m}, 0, \cdots,0) \epsilon C^{\perp}$. This must be truth if not then will contradict the fact that the rows of the parity check matrix *H* of C are also linearly independent. If $a_{i_j} = 0$ for some $j \in \{1, \cdots, m\}$, it follows that $\left\{P_{i_1}, \cdots, P_{i_{j-1}}, P_{i_{j+1}}, \cdots, P_{i_m}\right\}$ can recover the secret which contradict the minimality of the access structure A.

Conversely, if $c = (1, \cdots,0, c_{i_1}, 0, \cdots,0, c_{i_m}, 0, \cdots,0)$ is a minimal codeword, it follows that all rows $g_0, g_{i_1}, \cdots, g_{i_m}$ of *G* are linear dependent. Thus, the set of participants $\{P_{i_1}, \cdots, P_{i_m}\}$ can recover the secret. If any proper subset of this can recover the secret, then there exists a nonzero codeword which c properly covers. This contradicts the minimality of c. Therefore, $\{P_{i_1}, \cdots, P_{i_m}\}$ is a minimal acccess set. Thus, we have the following proposition.

Proposition 1. The minimal access structures of a secret sharing scheme based on a [*n*, *k*, *d*] – linear code C is the set of all minimal codewords *v* of the dual code $C^{\perp}$ of C, where *v* has a “1” in the first coordinate.

3. SECRET SHARING SCHEME BASED ON GROUP ALGEBRA CODES

Motivated by the construction described in the previous section, we next proposed a secret sharing scheme based on group algebra code.

Let F_q denote a finite field with q elements such that q is a prime. Given a finite group G of order n, the group algebra F_qG is a vector space over F_q, with basis G and so, is isomorphic to F_q^n as a vector space. The **group algebra** F_qG of G with coefficients in F is the set of all formal sums $\sum_{g\in G} a_g g$, where $a_g \in \boldsymbol{F}$. Addition and multiplication in F_qG are defined as $\sum_{g\in G} a_g g + \sum_{g\in G} b_g g = \sum_{g\in G}(a_g + b_g)g$ and $(\sum_{g\in G} a_g g)(\sum_{h\in G} b_h h) = \sum_{h\in G}\sum_{g\in G}(a_g b_h)gh$, respectively. A ***group algebra code*** is defined as an ideal of the group algebra F_qG. In particular, if G is a cyclic group then the ideal in F_qG is a cyclic code, and if G is an abelian group then the ideal in F_qG is an abelian code.

It is well-known that if $q \nmid n$, it follows from ***Maschke's Theorem*** (Theorem 1.9 in Berman (1967)) that the group algebra F_qG is semisimple and hence F_qG is a direct sum of minimal ideals, $F_qG = I_1 \oplus I_2 \oplus \cdots \oplus I_s$, where $I_j = F_qGe_j$ is the principal ideal of F_qG generated by e_j where e_j is an idempotent in F_qG for $j = 1,2,\cdots,s$. Let $M = \{e_j\}_{j=1}^{s}$be the set of all pairwise orthogonal idempotents. Every ideal I of F_qG is a direct sum $I = I_{i_1} \oplus I_{i_2} \oplus \cdots \oplus I_{i_t}$, where $t \leq s$. Now, write $F_qG = I \oplus J$, where $J = I_{j_1} \oplus I_{j_2} \oplus \cdots \oplus I_{j_{s-t}}$ is the direct sum of minimal ideals such that $I_{i_l} \neq I_{j_m}$ for all $1 \leq l \leq t$ and $1 \leq m \leq s-t$. Using these observations, we see that

$$\begin{aligned} I &= I_{i_1} \oplus I_{i_2} \oplus \cdots \oplus I_{i_t} \\ &= \langle e_{i_k} | k = 1,2,\cdots,t \rangle \\ &= \{u \in F_qG | ue_{i_h} = 0, \text{for all } h = 1,2,\cdots,s-t\} \end{aligned}$$

There are some nice properties for the idempotent e_{i_u} for $u = 1,2,\cdots,s$. In general, an element $e \in F_qG$ is an ***idempotent*** if $e^2 = e$. Furthermore, two idempotents e_1 and e_2 are orthogonal provided $e_1e_2 = e_2e_1 = 0$. A direct computation can show that if e is an idempotent, it follows that $1 - e$ is an idempotent orthogonal to e. Furthermore, if e_1 and e_2 are orthogonal, it follows that $e_1 + e_2$ is an idempotent.

For any $v = \sum_{g \in G} a_g g \in F_qG$, the support of v is defined as

$$supp(v) = \{g \in G | a_g \neq 0\},$$

and we said that $v_1 \in F_qG$ cover $v_2 \in F_qG$ provided $supp(v_2) \subseteq supp(v_1)$. To construct a secret sharing scheme via group algebra codes, we proposed the following algorithms. First, to construct the secret k and the corresponding shares, we proceed as follows:

Algorithm 1:
Choose a finite group G and a finite field F_q satisfying the condition that $gcd(|G|,q) = 1$. Construct all idempotents of F_qG. Hence, choose a set of idempotents to construct the following group algebra code

$$I = \{u \in F_q[G] | ue_{i_h} = 0, \text{for all } h = 1,2,\cdots,s-t\}.$$

The dealer D chooses a codeword $u \in I$ and write u in terms of group algebra element $u = \sum_{g \in G} a_g g$. Take the secret k as $a = a_e$, where e is the identity element of G and the remaining $|G| - 1$ coefficients a_g, for all $g \neq e$, in $u = \sum_{g \in G} a_g g$ are uniformly distributed to the set of participants $P = \{p_1, p_2, \cdots, p_{|G|-1}\}$.

Next, to recover the secret from a subset of participants and hence obtain the access structure, we used the following algorithm.

Algorithm 2:
Let $\tau \subseteq P$ and $\tau = \{P_{i_1}, P_{i_2}, \cdots, P_{i_k}\}$ such that $1 \leq k \leq |G| - 1$. Form the group algebra element $w = \sum_{j=1}^{k} a_{g_{i_j}} g_{i_j}$. Next, $w \in I$ if and only if $we_{i_h} = 0$, for all $h = 1,2,\cdots,s-t$. Form a homogeneous system of $s-t$ equations with k unknowns in which the access structure can be determined from these equations. Upon solving, we can recover the coefficients $a_{g_{i_j}}$ for all $j = 1,2,\cdots,k$ and hence the secret a.

To illustrate the above algorithms, we first choose a finite group said the dihedral group of order 6, $D_6 = \langle r, s \mid r^3 = s^2 = 1,\ rs = sr^2 \rangle$. To ensure the semisimplicity of $F[D_6]$, we choose $F = \mathbb{R}$. By constructing the character table of D_6, we can obtain all the three idempotents of $F[D_6]$ which are listed as follows:

$$e_1 = \frac{1}{6}(\langle r\rangle + \langle r\rangle s), e_2 = \frac{1}{6}(\langle r\rangle - \langle r\rangle s), e_3 = \frac{1}{3}(1 - \langle r\rangle).$$

Next, we construct the following group algebra code:

$$I = \{u \in \mathbb{R}[D_6] | ue_1 = ue_2 = 0\}.$$

Any $u \in I$ can be written in the form $u = \lambda_1 + \lambda_2 r + \lambda_3 r^2 + \lambda_4 s + \lambda_5 rs + \lambda_6 r^2 s$. Hence,

$$u = \lambda_1 + \lambda_2 r + \lambda_3 r^2 + \lambda_4 s + \lambda_5 rs + \lambda_6 r^2 s \ \in I$$

if and only if

$$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \lambda_6 = 0$$

and

$$\lambda_1 + \lambda_2 + \lambda_3 - \lambda_4 - \lambda_5 - \lambda_6 = 0.$$

The dealer D chooses $u = 2 - \frac{1}{3}r - \frac{5}{3}r^2 \in I$. Take the secret $k = 2$ and distribute $-\frac{1}{3}$ to P_1 (corresponds to the term r) and $-\frac{5}{3}$ to P_2 (corresponds to the term r^2). To recover the secret k, the participants P_1 and P_2 together will form the group algebra element $w = k - \frac{1}{3}r - \frac{5}{3}r^2$. Hence, $w \in I$ if and only if $k - \frac{1}{3} - \frac{5}{3} = 0$, that is, $k = 2$.

4. CONCLUDING REMARKS

The two algorithms proposed here are depended heavily on the group algebra codes, in which the idempotents used to generate the codes play an equally important role in determining the minimal access structure of a constructed secret sharing scheme. Our future investigation is to use the group algebra codes obtained in Denis C.K. and Ang (2013a) and Denis C.K. and Ang (2013b) together with algorithms 1 and 2 proposed here to obtain a subtle relationship between minimal codewords and minimal access structures.

REFERENCES

Ashikhmin, A. and Barg, A. 1998. Minimal vectors in linear codes. *IEEE T. Inform. Theory*. 44(5): 2010-2017.

Berman, S. D. 1967. Semisimple cyclic and abelian codes, II. *Kibernetika*. 3: 21-30.

Berman, S. D. 1989. Parameter of abelian codes in the group algebra KG of G = <a>×<b>, $a^p = b^p = 1$, p is prime, over a finite field K with a primitive p^{th} root of unity and related MDS-Codes. *Contempary Math*. 93: 77-83.

Denis C.K., Wong and Ang, M. H. 2013a. Group algebra codes defined over extra special *p*-group of order p^{2r+1}. *JP Journal of algebra, number theory and appl*. 30(1): 47- 60.

Denis C.K., Wong and Ang, M. H. 2013b. Group codes define over dihedral groups of small orders. *Malaysian Journal of Math. Sci*. 7(S): 101-116.

Ding, C., Kohel, D. and Ling, S. 2000. Secret sharing with a class of ternary codes. *Theor. Comput. Sci*. 246: 285-298.

Li, Z. H., Xue, T. and Lai, H. 2010. Secret sharing schemes from binary linear codes. *Information sci*. 180: 4412-4419.

Massey, J. L. 1993. Minimal codewords and secret sharing, In: *Proceedings of the 6th Joint Swedish-Russian International Workshop on Information Theory*. 276-279.

McElliece, R. J. and Sarwate, D. V. 1981. On sharing secrets and Reed Solomon codes. *Communications of the ACM*. 24: 583-584.

Shamir, A. 1979. How to share a secret. *Communications of the ACM*. 22(11): 612-613.

Yuan, J. and Ding, C. 2006. Secret sharing schemes from three classes of linear codes. *IEEE T. Inform. Theory*. 52(1): 206-212.

The Development of Deniable Authentication Protocol Based on The Bivariate Function Hard Problem

[1]Normahirah Nek Abd Rahman and [1,2]Muhammad Rezal Kamel Ariffin

[1]Al-Kindi Cryptography Research Laboratory,
Institute for Mathematical Research, Universiti Putra Malaysia,43400 UPM Serdang,
Selangor, Malaysia

[2]Department of Mathematics, Faculty of Science,
Universiti Putra Malaysia,
43400 UPM Serdang, Selangor, Malaysia

Email: mahirah_mayrah@yahoo.com, rezal@putra.upm.edu.my

ABSTRACT

A deniable authentication protocol enables a receiver to identify the true source of a given message but not to prove the identity of the sender to the third party. Non-interactive protocol are more efficient than interactive protocol in terms of communication overhead, and thus several non-interactive deniable authentication protocol have been proposed. So, it is very necessary to design a deniable authentication protocol which is non-interactive, secure and efficient. This paper propose a deniable authentication protocol based on the bivariate function hard problem (BFHP) cryptographic primitive. An improvement based on the BFHP is suggested since the problem of the BFHP provides the needed security elements plus its fast execution time. At the same time, the proposed protocol has properties of completeness, deniability, security of forgery attack, security of impersonation attack, security of compromising session secret attack and security man-in-the-middle attack also has been proved.

Keywords: Bivariate function hard problem, deniable authentication protocol.

1. INTRODUCTION

Deniability is a privacy property that ensures protocol participants can later deny taking part in a particular protocol run while authentication is used to ensure that users are who they say they are. So, a deniable authentication protocol is a protocol that enables a receiver to identify the true source of a given message, but not to prove the identity of the sender to a third party. There are many interactive and non-interactive deniable authentication protocols have been proposed. However, the interactive manner makes deniable protocols inefficient.

Deniable authentication has two characteristics that differ from traditional authentication. The first one is only the intended receiver can identify the true source of a given message (i.e. able to identify the signature of the sender) and the second one is the receiver cannot prove the source of the message to a third party (i.e. unable to prove the signature of the sender to a third party that the signature belongs to the sender). In other words, once the receiver has obtained and authenticated this message from the sender, the receiver cannot impersonate as the sender to a third party. Because of these two characteristics, the deniable authentication protocol is very useful for providing secure negotiation over internet.

For example, suppose that a customer wants to order an item from a merchant, then the customer should make an offer to the merchant and create an authenticator for the offer because the merchant must be sure that this offer really comes from the customer. However, the merchant wants to be able to prevent the customer from showing this offer to another party in order to elicit a better deal. Therefore, we need a protocol that enables a receiver to identify the source of a given message, but prevents a third party from learning the sender’s identity.

In 1998, Dwork *et al.* proposed an interactive deniable authentication protocol based on concurrent zero knowledge proof while Aumann and Rabin also proposed an interactive deniable authentication protocol based on the integer factorization problem (IFP). Later, Deng *et al.* (2001)

introduced two interactive deniable authentication protocols, respectively based on the discrete logarithm problem (DLP) and IFP. In 2002, Fan *et al.* introduced another simple interactive deniable authentication protocol based on Diffie-Hellman Key Distribution Protocol. However, there is a common weakness in the four previous protocols which the sender does not know to whom he proves the source of a given message. That is, a third party can impersonate the intended receiver to identify the source of a given message. Meanwhile, these four protocols are interactive and less efficient.

This scenario has led many cryptographers to come up with non-interactive deniable authentication protocol in order to enhance the efficiency. Shao, (2004) proposed a non-interactive deniable authentication protocol based on generalized ElGamal signature scheme. Lu and Cao (2005) proposed two deniable authentication protocols based on bilinear pairing and IFP respectively but their protocol is still unable to achieve the second characteristic of being a deniable authentication protocol.

In 2008, Hwang and Ma proposed deniable authentication protocol with anonymous sender protection. The sender's anonymity is also used to protect the sender's privacy. Though the sent message is forgeable by the receiver, but the sender can provide evidence to prove the message was really sent by him. Hence, to reduce the computational cost of proposed protocols with anonymous sender protection, Hwang and Chao (2010) proposed a new deniable authentication protocol with anonymous sender protection in an efficient way based on Schnorr signature scheme.

Then, Y. Zhang *et al.* (2011) proposed a new non-interactive deniable authentication protocol based on generalized ElGamal signature scheme, which is more efficient than the previous two protocols (Shao, 2004; Lee *et al.*, 2007) both in computation and communication. To authenticate the source of a message, although the proposed protocol needs one more modular exponentiation than Shao's protocol, but as to the length of the communicated messages, just $2|h|$ are required to be transmitted compared to $3|h|$ in Shao's protocol. Lee *et al.*'s protocol needs five exponentiation computation altogether compared to proposed protocol which needs only four. The transmitted bit for the proposed protocol are reduced to 320 bits compared to Lee *et al.*'s protocol which is $1184 \sim 2208$ bits.

In this paper, we propose a new non-interactive deniable authentication protocol based on the Bivariate Function Hard Problem (BFHP) (Ariffin *et al.*,2013). We prove our protocol is secure against forgery attack, impersonation attack, compromising session secret attack and man-in-the-middle attack and prove the properties of completeness and deniability of this protocol. With its guaranteed security, we also show that the performance of the protocol requires reasonable numbers of operation in both sign and verify phases.

The layout of the paper is as follows. In section 2, we will first review the definition of the BFHP. Proof will be given on the uniqueness and intractability of the BFHP. We will also review in this section, deniable authentication protocol in the standard model. In section 3, we propose the standard model of the deniable authentication protocol followed by the security analysis in which proof are given. In section 4, we provide efficiency analysis and comparison of the protocol. In section 5, the conclusion about our deniable authentication protocol is made.

2. PRELIMINARIES

2.1 Bivariate Function Hard Problem (BFHP)

The following proposition gives a proper analytical description of the Bivariate Function Hard Problem (BFHP).

Definition 1.

We define $\mathbb{Z}^+_{(2^{m-1},2^m-1)}$ as a set of positive integers in the interval $(2^{m-1}, 2^m - 1)$. In other words, if $x \in \mathbb{Z}^+_{(2^{m-1},2^m-1)}$, x is a *m*-bit positive integer.

Proposition 1. (Ariffin *et al.* (2013))

Let $F(x_1, x_2, \dots, x_n)$ be a multiplicative one-way function that maps $F: \mathbb{Z}^n \to \mathbb{Z}^+_{(2^{m-1},2^m-1)}$. Let F_1 and F_2 be such function (either identical or non-identical) such that $A_1 = F(x_1, x_2, \dots, x_n), A_2 = F(y_1, y_2, \dots, y_n)$ and $\gcd(A_1, A_2) = 1$. Let $u, v \in \mathbb{Z}^+_{(2^{n-1},2^n-1)}$. Let (A_1, A_2) be public parameters and (u, v) be private parameters. Let

$$G(u,v) = A_1u + A_2v \tag{1}$$

with the domain of the function G is $\mathbb{Z}^2_{(2^{n-1},2^n-1)}$ since the pair of positive integers $(u, v) \in \mathbb{Z}^2_{(2^{n-1},2^n-1)}$ and $\mathbb{Z}^+_{(2^{m+n-1},2^{m+n}-1)}$ is the codomain of G since $A_1u + A_2v \in \mathbb{Z}^+_{(2^{m+n-1},2^{m+n}-1)}$.

If at minimum $n - m - 1 = k$, where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, it is infeasible to determine (u, v) over $\mathbb{Z}$ from $G(u, v)$. Furthermore, (u, v) is unique for $G(u, v)$ with high probability.

Remark 1. We remark that the preferred pair (u, v) in $\mathbb{Z}$, is the *prf*-solution for (1). The preferred pair (u, v) is one of the possible solutions for (1) given by

$$u = u_0 + A_2t \tag{2}$$

and

$$v = v_0 - A_1t \tag{3}$$

for any $t \in \mathbb{Z}$.

Remark 2. Before we proceed with the proof, we remark here that the diophantine equation given by $G(u, v)$ is solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found. That is the BFHP is *prf*-solved when the preferred parameters (u, v) over $\mathbb{Z}$ are found.

Proof. We begin by proving that (u, v) is unique for each $G(u, v)$ with high probability. Let $u_1 \neq u_2$ and $v_1 \neq v_2$ such that

$$A_1u_1 + A_2v_1 = A_1u_2 + A_2v_2 \tag{4}$$

We will then have

$$Y = v_2 - v_1 = \frac{A_1(u_1 - u_2)}{A_2}$$

Since $\gcd(A_1, A_2) = 1$ and $A_2 \approx 2^n$, then the probability that Y is an integer is 2^{-n}. Then the probability that $v_1 - v_2$ is an integer solution not equal to zero is 2^{-n}. Thus $v_1 = v_2$ with probability $1 - \frac{1}{2^n}$.

Next we proceed to prove that to prf-solved the Diophantine equation given by (1) is infeasible to be solved. The general solution for $G(u, v)$ is given by (2) and (3) for some integer t.

To find u within the stipulated interval $u \in (2^{n-1}, 2^n - 1)$ we have to find the integer t such that the inequality $2^{n-1} < u < 2^n - 1$ holds. This gives

$$\frac{2^{n-1} - u_0}{A_2} < t < \frac{2^n - 1 - u_0}{A_2}$$

Then, the difference between the upper and the lower bound is

$$\frac{2^{\,n} - 1 - 2^{n-1}}{A_2} = \frac{2^{n-1} - 1}{A_2} \approx \frac{2^{n-2}}{2^m} = 2^{n-m-2}$$

Since $n - m - 1 = k$ where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, we conclude that the difference is very large and finding the correct t is infeasible. This is also the same scenario for v.■

Example 1. Let $A_1 = 191$ and $A_2 = 229$. Let $u = 41234$ and $v = 52167$. Then $G = 19821937$. Here we take $m = 16$ and $n = 8$. We now construct the parametric solution for this BFHP. The initial points are $u_0 = 118931622$ and $v_0 = -99109685$. The parametric general solution are: $u = u_0 + A_2 t$ and $v = v_0 - A_1 t$. There are approximately $286 \approx 2^9$ (i.e. $\frac{2^{16}}{229}$) values of t to try (i.e. difference between upper and lower bound), while at minimum the value is $t \approx 2^{16}$. In fact, the correct value is $t = 519172 \approx 2^{19}$.

2.2 Deniable Authentication Protocol in Standard Model

A deniable authentication protocol in standard model consists of four phases (Setup, Key Generation, Sign, Verify) which are defined as follows:

1. **Setup:** The authority determines the parameters that can be used by sender and the receiver to generate their private and public key.
2. **Key Generation:** An algorithm that generates private and public key. The private key which is randomly chosen and remain secret, to be used to generate the public key that will be published in public.
3. **Signing:** An algorithm that generates message authentication code (MAC) from the original message which involves hash function.
4. **Verifying:** An algorithm that involves verification of the new MAC generated with the MAC that has been sent by the sender. If both holds, the original message is authentic and has not be altered.

3. THE STANDARD MODEL OF DENIABLE AUTHENTICATION PROTOCOL BASED ON THE BFHP

3.1 Proposed Deniable Authentication Protocol

Setup: The authority randomly chooses the following public parameters:

1. p is a large prime number of n-bit size.
2. g is a primitive root in $\mathbb{Z}_p$.
3. $H(\cdot)$ is a collision free hash function with an output is n bits.

Key Generation:

When a user wishes to join the system, he chooses a random number $t \in \mathbb{Z}_p$ as his private key and compute $v = g^t (\bmod p)$ as his public key. The public key of each user is certificated by certification authority.

The sender, S chooses his secret key $t_s \in \mathbb{Z}^+_{(2^{2n-1},2^{2n}-1)}$ and computes $v_s = g^{t_s} (\bmod p)$ as his public key. The reason why t_s is chosen out of $\mathbb{Z}_p$ can be observe in step 2(i) of signing phase in order for BFHP to hold.

The receiver, R chooses his secret key $t_R \in \mathbb{Z}_p$ and computes $v_R = g^{t_R} (\bmod p)$ as his public key.

Signing:

When S wants to deniably authenticate a message M to the intended receiver R, he computes the following protocol:

1. Chooses randomly value of $\alpha \in \mathbb{Z}^+_{(2^{2n-1},2^{2n}-1)}$.
2. Computes
   i) $\sigma = H_1(M)t_s + H_2(M)\alpha$
   ii) $k_1 = (v_R)^{-H_1(M){t_s}^2} (\bmod p)$
   iii)$k_0 = (v_R)^{\alpha H_2(M)t_s} (\bmod p)$
   iv) $MAC = H(k_0 \parallel M)$

Then, S sends (k_1, σ, MAC) together with message M to R.

Verifying:

After receiving (k_1, σ, MAC) together with message M from S, receiver, R computes

$$k_1{}^* = (v_S)^{\sigma t_R}$$
$$k_0{}' = k_1 \cdot k_1{}^*$$
$$MAC = H(k_0{}' \parallel M).$$

R verifies whether $MAC = H(k_0{}' \parallel M)$

If two equations hold, R accepts the received information. Otherwise, R rejects it. Note that $\parallel$ is the concatenate operator of strings.

Proposition 2. (Completeness) If the sender and the receiver follow the protocol, the receiver is able to calculate $k_0{}'$ and then identify the source of the message.

Proof. From the proposed protocol, we have

$$\begin{aligned} k_0{}' &= k_1 \cdot k_1{}^* \\ &= ((v_R)^{-H_1(M){t_s}^2}) \cdot (v_S)^{\sigma t_R} (\bmod p) \\ &= g^{-t_R H_1(M){t_s}^2} \cdot g^{{t_s}^2 H_1(M) t_R} \cdot g^{H_2(M) t_s \alpha t_R} (\bmod p) \end{aligned}$$

$$= g^{H_2(M)t_s\alpha t_R} \pmod p$$

$$= (v_R)^{H_2(M)t_s\alpha} \pmod p$$

$$= k_0$$

So, $H(k_0{}' \parallel M) = H(k_0 \parallel M)$.■

3.1 Security Analysis of Deniable Authentication Protocol

Proposition 3. The proposed protocol is deniable.

Proof.

If the receiver can simulate all the transmitted information between him and the sender, he cannot prove to any third party where the message is from because the third party cannot identify whether the message is from the sender or is forged by receiver himself.

So, if the receiver tells a third party that the data is from the sender, the sender can deny it and claims that the receiver himself forge the data. Hence the third party cannot identify who tells the truth.

After receiving (k_1, σ, MAC), the receiver can identify the source of the (k_1, σ, MAC) with his own private key, t_R. However, he cannot prove the source of the message to any party because the receiver can calculate k_0, so he can select any other message M' and construct $MAC' = H(k_0{}' \parallel M')$ and tells the third party (k_1, σ, MAC') is the information he gets from S.

Without the randomly selected $\alpha \in \mathbb{Z}^+_{(2^{2n-1}, 2^{2n}-1)}$, the secret key t_s of S and secret key t_R of R, the third party cannot derive k_0 and $k_0{}'$. So he cannot prove whether the receiver is telling the truth.■

Proposition 4. If the attacker cannot personate as the sender to communicate with the intended receiver, then the proposed protocol can withstand forgery attack.

Proof.

The session secret key $k_0 = (v_R)^{H_2(M)\alpha t_s} \pmod p$ is protected by BFHP. That is, the pair (α, t_s) is protected by BFHP on σ. If the BFHP surrounding σ is *prf*-solved, both (α, t_s) are found. Hence, no third party can forge a valid k_0 to cheat the receiver.■

Remark 3.

On the other hand, if the DLP is solved, $t_s \in \mathbb{Z}_p$ would be found. However, the corresponding preferredα would not be obtained. In fact, both the preferred integers (α, t_s) is still not obtained.

Observed from $v_s = g^{t_s} \pmod p$. Solving the DLP, we will get $t_{s_0} \in \mathbb{Z}_p$. If $t_s \equiv t_{s_0} \pmod p$, the attacker may initiate search for t_s since $t_s = t_{s_0} + pj$ for some $j \in \mathbb{Z}$. Observe that since $t_{s_0}, p \sim 2^n$ and $t_s \sim 2^{2n}$, we have $j \sim 2^n$. Hence the probability to obtain the correct j is $\frac{1}{2^n}$.

If $t_s \not\equiv t_{s_0} \pmod p$, the attacker may not initiate search for t_s since he cannot find $j \in \mathbb{Z}$ as j is the number of time t_{s_0} is reduced by p until t_s is obtained.

Proposition 5. If an attacker wants to impersonate as the intended receiver in order to identify the source of a given message, then the proposed protocol can withstand such an impersonation attack.

Proof.

In our protocol, any third party want to impersonate as the intended receiver cannot identify the source of the message even if he obtains (k_1, σ, MAC). If he can verify the message authenticator, he must find k_0 and $k_0{}'$. As we prove above, he cannot forge k_0 and $k_0{}'$ as

$$\begin{aligned}
k_0{}' &= k_1 \cdot k_1{}^* \\
&= ((v_R)^{-H_1(M)t_s{}^2}) \cdot (v_S)^{\sigma t_R} \pmod p \\
&= g^{-t_R H_1(M)t_s{}^2} \cdot g^{t_s{}^2 H_1(M)t_R} \cdot g^{H_2(M)t_s\alpha t_R} \pmod p \\
&= g^{H_2(M)t_s\alpha t_R} \pmod p \\
&= (v_R)^{H_2(M)t_s\alpha} \pmod p
\end{aligned}$$

It is shown that t_R is required in each step to calculate $k_0{}'$. Without the receiver's private key, t_R, it is impossible for the attacker to forge $k_0{}'$. ■

Proposition 6. The proposed protocol is secure against man-in-the-middle attack if man-in-the-middle cannot establish any session key with either the sender or the receiver.

Proof.

Objective of the man-in-the-middle attack is to pretend to be the sender and cheat the receiver. In order to pretend as a sender, he needs to compute σ' for a corresponding M'. But this is infeasible because the pair (α, t_s) is protected by BFHP within the initial σ. On the other hand, the man-in-the-middle cannot pretend to be the receiver to cheat the sender because he needs to obtain the receiver's private key, t_R to compute $k_1{}^* = (v_S)^{\sigma t_R}$. This is also infeasible because t_R is protected by the DLP within v_R. Therefore, the attacker is unable to pretend to be the sender or the receiver.■

Proposition 7. A compromised session secret does not affect the security of the proposed deniable authentication protocol.

Proof.

The session secret can be derived from

$$k_0{}' = ((v_R)^{-H_1(M)t_s{}^2}) \cdot (v_S)^{\sigma t_R} \bmod p = g^{H_2(M)t_s\alpha t_R} \pmod p$$

where random α is chosen independently for each session. If an attacker wants to forge the deniable information with the forged message M' by using the compromised session secret key k_0, the receiver will derive a different session secret from the forged information. This is because the message and its corresponding session secret are interdependent. In other words, the session secret and its corresponding message M for each round are independent. This has been realized in our proposed protocol as shown in equation $\sigma = H_1(M)t_s + H_2(M)\alpha$. Hence, a compromised session secret does not affect the security of other session.■

4. COMPARISON

To study the performance of the proposed protocol, we compare it with some previous proposed deniable authentication protocols. We make comparison against the most known efficient interactive protocol (Fan *et al.* 2002) and non-interactive protocol (Y. Zhang *et al.*, 2011). The comparison is summarized as in Table 1.

	Fan *et al.* protocol		Y. Zhang *et al.* protocol		The proposed protocol	
	S	*R*	*S*	*R*	*S*	*R*
Exponentiation	2+1	2+2	2	3	2	1
Hashing Computation	1+1	1+1	2	2	3	1
Data Transmission Overhead	$2\|n\| + 2\|h\|$		$2\|h\|$		$3\|n\| + \|r\|$	
Interactive	Yes		No		No	

Table 1: The comparison among deniable authentication protocol

To authenticate the source of a message in Fan *et al.*'s interactive protocol, two modular exponentiation computation and one hashing computation are required by both sender and receiver. In addition, the sender needs to compute a signature with a message recovery which requires one modular and one hash function computation. The receiver needs to verify the signature which requires two modular exponentiation computation and one hash function computation. The data transmission overhead for Fan *et al.*'s protocol is $2|n| + 2|h|$ bits where $|n|$ is modular size and $|h|$ is output size of hash function.

Our proposed protocol is non-interactive so that the communication process is shorter than in any interactive protocol. In signing phase, the sender needs two modular exponentiation computation and three hash function computation. The receiver needs one modular exponentiation computation and one hash function computation in verifying phase. Data transmission overhead for our proposed protocol is $3|n| + |r|$ bits, $|r|$ denotes the size of α and t_s while Y. Zhang *et al.*'s protocol is $2|h|$ bits.

5. CONCLUSION

A new deniable authentication protocol based on the bivariate function hard problem has been developed. One can observe from the Table 1 that the number of exponentiation computation needed is less that known efficient deniable authentication schemes. This suggest that the proposed method has better computational complexity on both the sender and the receiver's end.

Some possible attacks have also been considered and we showed that our proposed protocol is secure against forgery attack, impersonation attack, compromising session secret attack and man-in-the-middle attack. Hence, our proposed deniable authentication protocol is more desirable than existing schemes.

REFERENCES

Ariffin, M.R.K., Asbullah, M. A., Abu, N. A. and Mahad, Z. 2013. A new efficient asymmetric cryptosystem based on the integer factorization problem of $\mathrm{N} = \mathrm{p}^2\mathrm{q}$. Malaysian Journal of Mathematical Sciences 7(S): 19-37.

Aumann, Y. and Rabin, M.O. 1998. Authentication, enhanced security and error correcting codes. *Advances in Cryptology – CRYPTO '98.* Vol. 1462, pp. 299 – 303. Springer-Verlag: Lecture Notes in Computer Science (LNCS).

Deng, X., Lee, C.H., and Zhu, H. 2001. Deniable authentication protocol. *IEE Proc. Comput. Digital Techniques* 148: 101-108.

Dwork, C., Noar, M. and Sahai, A. 1998. Concurrent zero-knowledge. *Proceedings of 30th Annual ACM Sympossium of Theory of Computing 1998*, pg. 409-418.

ElGamal, T. 1985. A public key cryptosystem and signature scheme based on discrete logarithms. *IEEE Transaction on Information Theory* 31(4): 469-472.

Fan, L., Xu, C.X. and Li, J.H. 2002. Deniable authentication protocol based on Diffie Hellman algorithm. Electronic Letters 38(4): 705-706.

Hwang, S. J. and Chao, C. H. 2010. An efficient non-interactive deniable authentication protocol with anonymous sender protection. *Journal of Discrete Mathematical Sciences and Cryptography*, 13:3, 219-231.

Hwang, S. J. and Ma, J. C. 2008. Deniable authentication protocol with anonymous sender protection. *International Computer Symposium, Tamsui, Taiwan, 2008*, pp. 412-419.

Lee, W.B., Wu, C.C. and Tsaur, W.J. 2007. A novel deniable authentication protocol using generalized ElGamal signature scheme. *Information Sciences* 177: 1376-1381.

Lu, R. and Cao, Z. 2005. A new deniable authentication protocol from bilinear pairings. *Applied Mathematics and Computation* 168:954-961.

Lu, R. and Cao, Z. 2005. Non-interactive deniable authentication protocol based on factoring. *Computer Standard & Interfaces* 27: 401-405.

Shao, Z. 2004. Efficient deniable authentication protocol based on generalized ElGamal signature scheme. *Comput. Standards Interfaces* 26: 449-454.

Zhang, Y., Xu, Q. and Liu, Z. 2011. A new non-interactive deniable authentication protocol based on generalized ElGamal signature scheme. *Information Technology and Artificial Intelligence Conference (ITAIC), 2011 6th IEE Joint International*, pg. 193-197.

Digital watermarking using a fusion of Digital Holographic interferometry and Singular Value Decomposition

[1]Hao-Qiang Tan,[2]Thian-Khok Yong, [3]Bok-Min Goi,[4]Tong-Yuen Chai
Faculty of Engineering Science, Universiti Tunku Abdul Rahman
Kuala Lumpur, Malaysia
Email: [1]tanhq1@mail2.utar.edu.my, [2]yongtk@utar.edu.my,[3]goibm@utar.edu.my,[4]chaity@utar.edu.my

ABSTRACT

One of the unique characteristic of holographic techniques allows precise measurement of the object displacement without any physical contact. In double exposure digital holographic interferometry (DHI), two different states of the object are made to interfere and form the interference fringe. In this paper a new method of watermarking using digital holographic interferometry (DHI) is presented here. First a 3D object undisturbed state is digitally captured as a hologram. Then a load is added onto the 3D object to create the disturbed state. Both holograms are saved as digital image. For watermarking purpose, the undisturbed digital hologram is embedded as watermark into the host image using Singular value decomposition (SVD) algorithmic. While the disturbed digital hologram serves as a key. To recover the watermark information, the undisturbed digital hologram can be numerically reconstructed using the correct geometrical parameters and key to produce the interference fringe. The use of key offers an extra level of security in overall.Experiment results show that the proposed method is more robust than conventional holographic watermarking against most image processing attacks.

Keywords : digital holography , digital Holographic interferometry, singular value decomposition

1. INTRODUCTION

Today's technology is rapidly evolving. Digital contents such as image, audio and video are easily downloadable and duplicated. As a result, it is difficult to differentiate a pirated copy from its original. These cause a concern over ownership and copyright protection of digital content. To prevent such information theft, digital watermarking techniques (Cox, Miller and Bloom, 2000) are used to protect the owner copyright by embedding hidden information into the digital contents. Meanwhile, optical watermarking techniques by digital holography have been extensively studied due to its unique characteristic (Schnars and Jüptner, 2002) it processed. As a result of this characteristic, every part of a hologram contains the information (amplitude and phase) about the entire object data. In holographic watermarking (Yong *et al.*, 2012; Takai and Mifune, 2002; De, Nah and Kim, 2009), a hologram is embedded into host image as watermark. The method reported to have strong robustness especially against geometrical distortion. However, the security of the watermark is not safe as the geometrical parameters including the wavelength and diffraction distance can be deducted through research. To further enhance the holographic security, data encryption method (Kishk and Javidi, 2003; Kishk and Javidi, 2002; Meng *et al.*, 2007; Giuseppe and Michele, 2011; Nishchal, Pitkä̈aho and Naughton, 2010; Okman and Akar, 2007) have been proposed for holographic watermarking. Each method mentioned above to have reported a high security watermarking as they require a number of correct keys to recover the watermark image. However the data quantity of the keys is very large. Furthermore the sizes of the resultant encrypted data increase proportionally since the holograms are complex signal. Moreover, the encrypted data requires a set of transformation to recover the watermark information.

Meanwhile, the other unique characteristic of holographic techniques allows precise measurement of the object displacement without any physical contact. In digital holographic interferometry, two different states of the object are made to interfere to form the interference fringes. The fringe pattern shows the phase different between the interfering waves. In this experiment, a cantilever beam is used as object. Using the double exposure digital holographic interferometry, the object undisturbed state is firstly captured as a hologram before a second hologram is capture for the now disturbed object. Both holograms are then numerical simulated to reconstruct the fringes pattern.

In this paper, a new approach of watermarking based on digital holographic interferometry (DHI) is proposed here. There are a few advantages of this proposed scheme. Firstly either one of the hologram (undisturbed or disturbed) can be used as a watermark or as a key. Secondly, watermark information (fringes pattern) can only be retrieved by using the correct key and geometrical parameters. Lastly, the proposed scheme is less complex as data does not require a set transformation to recover the watermark information. To test the watermark robustness, we embed the undisturbed hologram as watermark into the host image using Singular Value Decomposition (SVD) algorithmic (Tan, Yong and Goi, 2012). The watermark recovery process requires knowledge of the key and geometrical parameters. For comparison with holographic watermarking, the cantilever hologram from the undisturbed object is embedded into the host image via SVD. The robustness of the proposed method and holographic watermarking are compared and tested against image processing attacks. The proposed watermarking scheme is shown in Figure 1.

2. DIGITAL HOLOGRAPHY INTERFOMETRY

The experimental set-up for the implementation of DHI based on off-axis configuration is shown in Figure 2. The object is a cantilever beam, which is firmly secured at one end and loaded at the free end. The load is applied with a micrometer screw toward the direction of the CCD camera. Two holograms are captured. First the undisturbed object is recorded. The complex amplitude of the object wave in undisturbed state is given as:

$$O_1(x,y) = o(x,y)\exp[\mathrm{i}\varphi(\mathrm{x},\mathrm{y})] \tag{1}$$

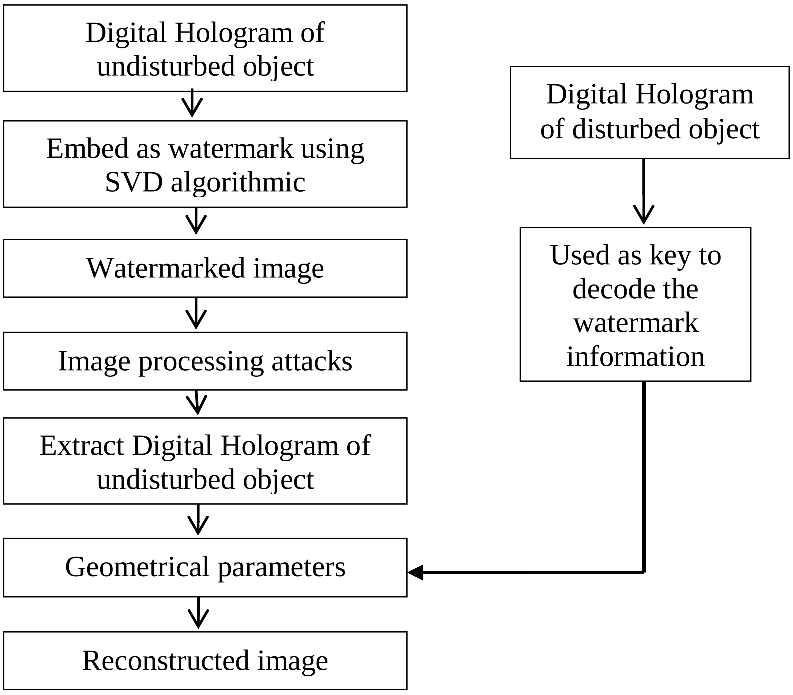

Digital Hologram of undisturbed object
Embed as watermark using SVD algorithmic
Watermarked image
Image processing attacks
Extract Digital Hologram of undisturbed object
Geometrical parameters
Reconstructed image
Digital Hologram of disturbed object
Used as key to decode the watermark information

Figure 1: Proposed watermarking scheme using fusion of DHI and Singular Values Decomposition

where $o(x,y)$ and $\varphi(x,y)$ are the real amplitude and phase of the object wave, respectively.A second hologram is recorded after the cantilever is bent a few microns to produce the disturbed state. The complex amplitude of the disturbed object wave is given as:

$$O_2(x,y) = o(x,y)\exp\left[i\left(\varphi + \Delta\varphi(x,y)\right)\right] \tag{2}$$

where $\Delta\varphi(x,y)$ is the interference phase between the undisturbed and disturbed state of the object.

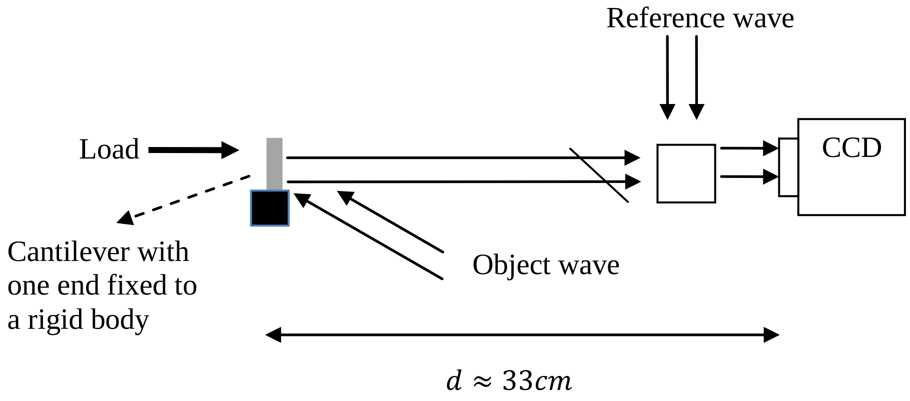


Figure 2: Experiment setup for digital holographic interferometry based on off-axis configuration

3. PROPOSED WATERMARKING SCHEME

3.1 Embedding of watermark

Figure 3 shows the procedure to embed and extract the watermark using the SVD algorithmic. A host image, I is SVD transformed into three (U, S, V) components matrices by

$$I = \quad USV^T \tag{3}$$

where U and V are the orthogonal matrices and S is a diagonal matrix. The hologram H is embedded into the S matrix by

$$Sw = \mathrm{S} + \alpha H \tag{4}$$

where S and H are host image and watermark respectively, α is the weighting factor and its value is set to 0.2 in this experiment. An inverse SVD transform is applied to obtain the final watermarked image.

$$G = \mathrm{USw}V^T \tag{5}$$

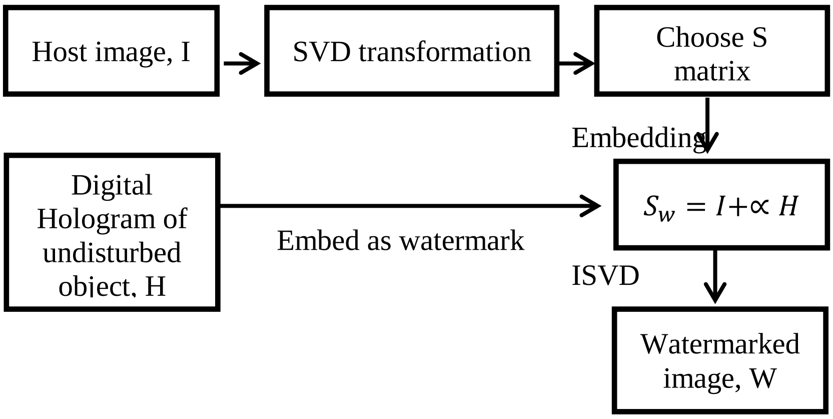


Figure 3: Flowchart of the procedure to embed and extract the watermark using the SVD algorithmic

3.2. Watermark image extraction and reconstruction

Figure 4 shows the procedure to extract and reconstruct the watermark. The watermark image is recovered by using a reverse watermark embedding process. To reconstruct the watermark information, firstly the complex amplitude of two holograms is calculated. Secondly the phase of each hologram is calculated by using the 2-dimensional Discrete Fourier Transform (Schnars and Jüptner, 2002). Lastly, the interference phase the hologram is determined and displays in grayscale image.

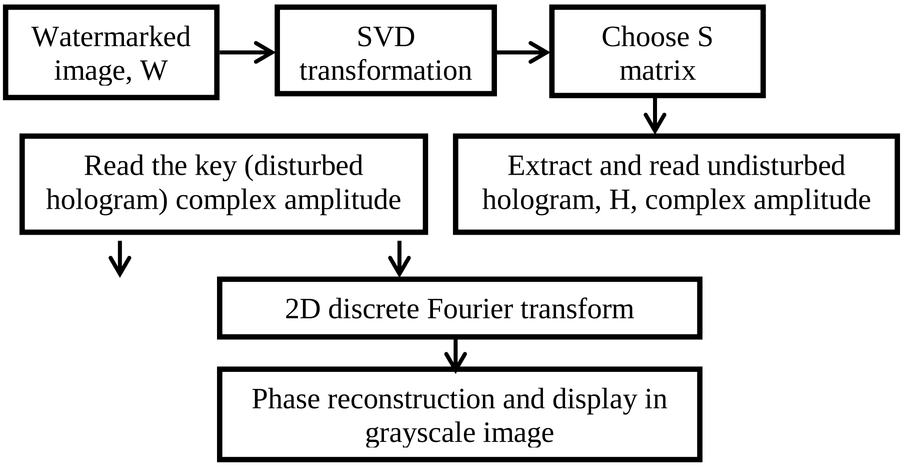

Watermarked image, W
SVD transformation
Choose S matrix
Read the key (disturbed hologram) complex amplitude
Extract and read undisturbed hologram, H, complex amplitude
2D discrete Fourier transform
Phase reconstruction and display in grayscale image

Figure 4: Flowchart of the extraction and reconstruction of the watermark

4. RESULTS AND DISCUSSION

The light source used in this experiment is a 30-mW DPPS with wavelength of 532.8 nm and recording distance of 33cm. Figure 5(a)-(b) shows the undisturbed hologram and disturbed hologram. Figure 5 (c) shows the object deformation phase contrast image. Figure 5(d) shows the reconstructed phase contrast image using wrong key.

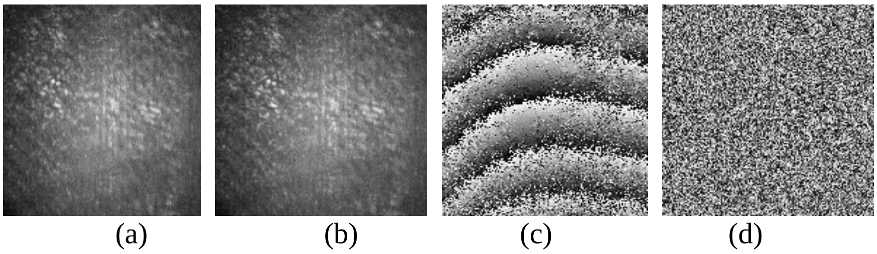
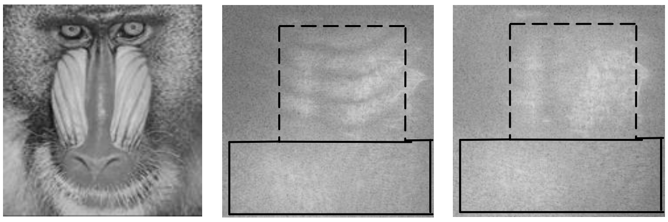


Figure 5: Digital Hologram (a) before disturbed (b) after disturbed. The results of numerical reconstruction c) Phase contrast image (d) Reconstructed image using wrong key

The host image is a 1024x1024 gray image baboon image as shown in Figure 6(a). Figure 6(b) shows the reconstructed cantilever contrast image reconstructed from undisturbed hologram. In order to test the robustness, the watermarked image will undergo some common image processing attacks including salt and pepper noise, JPEG compression, Gaussian filter and image cropping. The quality of the retrieved is tested by 2-D correlation coefficient (CC) (Okman and Akar, 2007).



(a) (b) (c)

Figure 6: (a) Baboon Image. The results of numerical reconstruction (b) Amplitude contrast image (c) Cantilever hologram

Table 1 shows the 2-D correlation coefficient (CC) after each image processing attacks. The proposed method shows better robustness especially against JPEG compression and Gaussian filter. However, the proposed method is slightly vulnerable to noise attacks. Both the watermark of proposed method and the holographic watermarking are detected after 75% of the total watermarked image is chopped.

	Normalized Correlation Coefficient (NCC)		
	Proposed method	**Holographic watermarking (Cantilever hologram)**	**Percentage difference (%)**
No Attack	0.996	0.940	-5.66
Salt and pepper Noise			
100%	0.610	0.642	-4.98
150%	0.600	0.632	-5.06
200%	0.590	0.618	-4.53
250%	0.587	0.613	-4.24
JPEG Compression			
Q = 25	0.784	0.715	9.65
Q = 50	0.737	0.703	4.84
Q = 75	0.684	0.681	0.44
Q = 95	0.600	0.600	0.00
3x3 Gaussian filter			
σ = 0.2	0.987	0.735	34.29
σ = 0.3	0.980	0.730	34.25
σ = 0.4	0.947	0.719	31.71
σ = 0.5	0.849	0.704	20.60
Image cropping			
20%	0.785	0.658	19.30
50%	0.710	0.636	11.64
60%	0.673	0.628	7.17
75%	0.555	0.540	2.78

Table 1: 2-D correlation coefficient values after each attack

5. CONCLUSION

In this paper, we proposed a watermarking scheme using holographic interferometry. Experiment results from different image processing attacks demonstrated that the proposed method is more robust compared to holographic watermarking. Another advantage of this method as it requires a key (hologram of disturbed object) besides the geometrical parameters to recover the watermark image. Therefore the proposed method offers higher level of higher security in overall. For future work, we are planning to compare results of the proposed system with other recent methods.

REFERENCES

Cox. I. J. , Miller, M. L. and Bloom, J. A. 2000 "Watermarking applications and their properties," *Information Technology: Coding and Computing, 2000. Proceedings.* 6 – 10.

Schnars, U. and Jüptner, W. 2002. Digital recording and numerical reconstruction of holograms. *Meas. Sci. Technol.* 13: 85-101.

Yong, X., Shan, W. Y., Cao, X. L., and Feng. Q. Q. 2012. Analysis and comparison of holographic and traditional digital image watermarking in DWT domain. *Computer Science & Education (ICCSE).* 790-793.

Takai, N. and Mifune, Y. 2002. Digital watermarking by a holographic technique. *Appl. Opt.* 41, 865-873.

De, L., Nah, J., and Kim, J. 2009, December. A Forensic Marking Algorithm based on DWT-SVD using Hologram. In *Multimedia, 2009. ISM'09. 11th IEEE International Symposium on* (pp. 589-594). IEEE.

Kishk, S. and Javidi. B. 2003. 3D Object Watermarking by a 3D Hidden Object. *Optics Express*. 11: 874-888.

Kishk, S. and Javidi. B. 2002. Information hiding technique using double phase encoding. *Appl. Opt.* 41: 5470-5482.

Meng, X. F., Cai, L. Z., He, M. Z., Dong, G.Y., and Shen, X. X. 2007. Cross-talk free image encryption and watermarking by digital holography and random composition. *Optics Communications*. 269(1): 47-52.

Giuseppe, S. S. and Michele, D. S. 2011. Holographic watermarking for authentication of cut images, *Optics and Lasers in Engineering.* 49(12): 1447-1455.

Nishchal, N. K., Pitkäaho, T. and Naughton, T. J. 2010. Digital Fresnel hologram watermarking," *9th Euro-American Workshop on Information Optics*. 1-3.

Okman, O. E. and Akar, G. B. 2007. Quantization index modulation-based image watermarking using digital holography. *J. Opt. Soc. Am.* 24(1): 243–252.

Tan, H. Q., Yong, T. K., and Goi, B. M. 2012. Holographic watermarking based on off-axis hologram and DWT. *IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (STUDENT 2012).* 222-226.

Tan, H. Q., Yong, T. K., and Goi, B. M. 2012. Holographic Watermarking Based On Off-Axis Hologram And Dwt-Svd. *International Journal of Cryptology Research.* 4(1): 17 – 31.

Analysis of Steganography Substitution System Methods Using New Testing Techniques Proposed for Strict Avalanche Criterion

[1]Abdul Alif Zakaria, [2]Nor Azeala Mohd Yusof,
[3]Wan Zariman Omar, [4]Nik Azura Nik Abdullah and
[5]Hazlin Abdul Rani
Cyber Technology Research Department
CyberSecurity Malaysia
Kuala Lumpur, Malaysia
[1]alif@cybersecurity.my, [2]azeala@cybersecurity.my,
[3]wanzariman@cybersecurity.my, [4]azura@cybersecurity.my, [5]hazlin@cybersecurity.my

ABSTRACT

In this research, we present the analysis on steganography substitution system methods; Least Significant Bit Substitution, Random Interval, Pseudorandom Permutation, Image Downgrading & Covert Channels, and Cover-regions & Parity Bits. New testing techniques proposed by Phyu Phyu Mar and Khin Maung Latt for strict avalanche criterion is implemented to analyze secret message bit distribution in all methods. One million bits of secret messages have been used for sampling in this testing. This analysis compares each methods results on three tests; total number of bit changes in each output, output values in each output, and total number of bit changes in each bit position. From our observation, Random Interval and Cover-regions & Parity Bits produced the best test results compared to the other three steganography substitution methods.

Keywords: Steganography, Substitution System, Strict Avalanche Criterion, Hamming Weight

1. INTRODUCTION

Steganography is the art and science of hiding messages. The word Steganography comes from the Greek words Steganós meaning covered and Graptos meaning writing. Steganography and cryptography have similarities where both are used to protect important information. However, steganography is different from cryptography because it involves hiding information without noticing any alteration made to the cover object (Zaidan *et al.*, 2009). Cover object or carrier is the file such as text, picture, image, audio or video in which secret message is hidden. The secret message can also be the same form as cover object. File containing secret message that has been hidden in cover object is called stego-object (Bandyopadhyay and Banik, 2012).

Steganalysis is the art and science of detecting a secret communication in steganography. Detectable traces in the cover medium may exist if message is hidden. Changes in statistical properties of the cover may lead to steganalyst attempting to detect the existence of the secret communication. This attempting process to detect statistical traces is called statistical steganalysis (Leivaditis, 2010).

Many developed embedding techniques in the last few years have been successfully attacked. Statistical properties of secure stego-systems should be protected and controlled because each time a new secure embedding algorithm is developed, steganalyst will find a new statistic to ensure the success of their attack (Leivaditis, 2010). It is important to inject a secret message into a carrier document that no detectable changes are introduced for secure secret communication. The main objective is to avoid introducing statistically detectable modifications into the carrier document and to not raise suspicion to the attacker. Size of the secret message, format and content of the carrier image may directly influence the ability of not being detected and lead to higher probability that the modification can be statistically detected (Fridrich and Du, 2000).

This paper illustrates the results for steganography substitution system methods using Hamming weight tests. A short description of Least Significant Bit Substitution, Random Interval, Pseudorandom Permutation, Image Downgrading & Covert Channels, and Cover-regions & Parity Bits methods are described in **Section 2**. **Section 3** explains the purposes for all Hamming weight tests. In **Section 4** and **Section 5**, the experimental setup and analysis are discussed. Summary and conclusion of the research are lastly demonstrated in **Section 6**.

2. STEGANOGRAPHY SUBSTITUTION SYSTEM METHODS

Figure 1 shows an existed classification of steganography (Al-Ani *et al.*., 2010). Substitution system is one of six steganography classification method that is applied in steganography. This system substitutes redundant or least significant parts of a cover with a secret message. Receiver can extract the information if the position where secret information has been embedded is known. Below are the methods under substitution system that are publicly known and implemented in steganography tools (Johnson and Katzenbeisser, 2000).

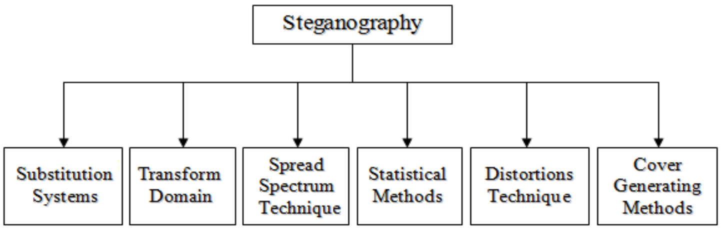


Figure 1: Steganography Classification

A. *Least Significant Bit Substitution*

This method stores one message bit in the least significant bit (LSB) of cover-element. To reconstruct the secret message, the LSB of the selected cover-elements are extracted and lined up.

B. *Random Interval*

A pseudorandom number generator is used in this method to spread the secret message over the cover-elements in a rather random manner. Both sender and receiver share a stego-key as a seed for a random number generator. A random sequence is created in which the distance between two embedded bits is determined pseudorandomly. The secret message bits are stored according to the distance between two embedded bits.

C. *Pseudorandom Permutation*

Distribution of the secret message is done in random manner over the whole cover-elements. The main goal is to increases the complexity for an attacker because there is no guarantee that subsequent message bits are embedded in the same order. A sequence is generated using a pseudorandom number generator. The secret message bits are stored according to bit position of cover-elements which is determined by the generated sequence.

D. *Image Downgrading and Covert Channels*

Images can be exchanged covertly using this method. Both secret messages and covers are in form of images. Given a cover-image and secret image of equal dimensions, the sender exchange the four least significant bits of the cover's color values with the four most significant bits of the secret image. Access to the most significant bits of the secret image is gained when the receiver extracts the four least significant bits out of the stego-image.

E. *Cover-regions and Parity Bits*

A stego-key is used as the seed to generate pseudorandom sequence of disjoint cover-regions. Only one bit of the secret message is stored in a whole cover-region rather than in a single element. In the embedding process, disjoint cover-regions are selected, each encoding one secret bit in the parity bit. One LSB of a random chosen cover-element is flipped if the parity bit of the cover-region does not match with the secret bit to encode. The parity bits of all the selected cover-regions are calculated and lined up to reconstruct the message at the receiver (Bandyopadhyay and Banik, 2012).

3. NEW TESTING TECHNIQUES

These testing techniques are based on new techniques for Strict Avalanche Criterion (SAC) that was proposed by Phyu Phyu Mar and Khin Maung Latt (Mar and Latt, 2008). The proposed techniques highlighted three main criteria which are avalanche effect, completeness, and strong function (Mar and Latt, 2008; Li and Cusick, 2007). Definition of each criteria are described as follows:-

- Avalanche effect; a function exhibits the avalanche effect if and only if an average of one half of the output bits change whenever a single input bit is complemented.
- Completeness; a function is complete if and only if each output bit depends on all of the input bits. Thus, if it is possible to find the simplest Boolean expression for each output bit in terms of the input bits, each of these expressions would have to contain all of the input bits if the function is complete.
- Strong function; a function is a strong function if and only if each of its output bits should change with a probability of one half whenever a single input bit is complemented.

This new testing techniques are simpler and easier because the existing techniques use mathematical equations and requires test to be repeated. By using only simple calculation, the result can easily be evaluated from representation of bar graphs as it indicates whether the testing substitution system methods are good or poor.

4. EXPERIMENTAL SETUP

This research paper considers five substitution system algorithms (methods) to be tested namely Least Significant Bit Substitution, Random Interval, Pseudorandom Permutation, Image Downgrading & Covert Channels, and Cover-regions & Parity Bits. Each algorithm is tested by using 100 samples containing 10,000 bit message per sample. Outputs from each algorithm are analyzed using the new testing techniques proposed for Strict Avalanche Criterion (Mar *et al.*, 2008).

A. Frequency Analysis Of Various Hamming Weight (Avalanche Effect)

Output values of the algorithm which correspond to two inputs were chosen. Apply XOR function to compute the differential value of these two outputs and find the hamming weight in the differential value. For necessary count of testing, repeat above steps. Analyze the frequency of various differential values by counting the number of '1's in each output. This method is to observe total number of bit changes in each output.

B. Frequency Analysis Of Various Differential Value (Completeness)

First, two inputs with their corresponding output values of the algorithm were chosen. Next, the differential value of these two outputs was computed by applying XOR function. Then, the hamming weight in the differential value of the outputs was determined. Above steps were repeated for necessary count of testing. The frequencies of various differential values were analyzed by counting the total number of occurrence for each output. The objective of this method is to observe the value of each output.

C. Analysis Of Hamming Weight According To The Bit Position (Strong Function)

Choose two inputs and find their corresponding output value of the algorithm. Compute the differential value of these two outputs by applying XOR function. Repeat above steps for necessary count of testing. Analyze the hamming weight according to the bit position of resulting differential values by counting total number of '1's in each bit position. This method is to observe total number of bit changes in each bit position.

5. RESULTS AND ANALYSIS

A. *Frequency Analysis Of Various Hamming Weight (Avalanche Effect)*

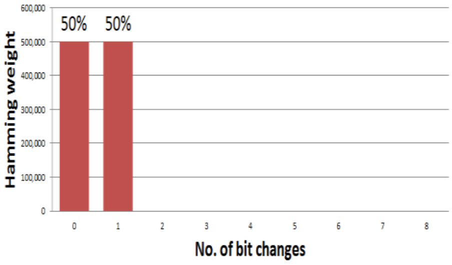


Figure 2: Least Significant Bit Substitution

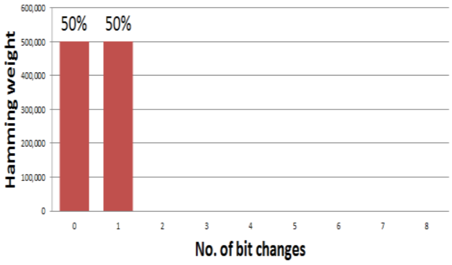


Figure 3: Random Interval

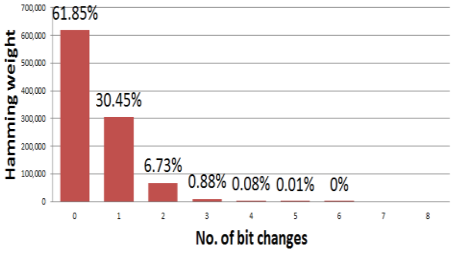


Figure 4: Pseudorandom Permutation

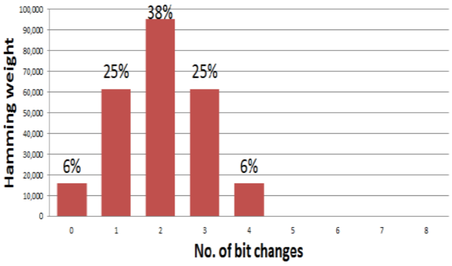


Figure 5: Image Downgrading & Covert Channels

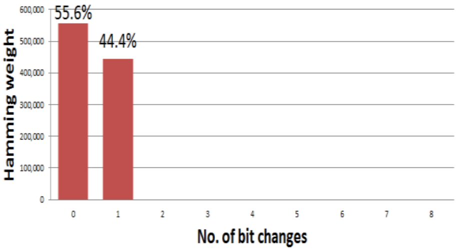


Figure 6: Cover-regions & Parity Bits

Test results for avalanche effect indicate that Least Significant Bit Substitution, Random Interval, and Cover-regions & Parity Bits methods changed up to one bit output, whereas Image Downgrading & Covert Channels method changed up to 4 bits output. Pseudorandom Permutation method changed up to 6 bits output. It is important to note that more bit changes will increase the suspicious level of the existence of secret messages. These results are referred from Figure 2,3,4,5 and 6.

B. Frequency Analysis Of Various Differential Value (Completeness)

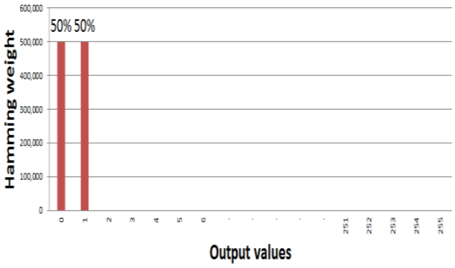


Figure 7: Least Significant Bit Substitution

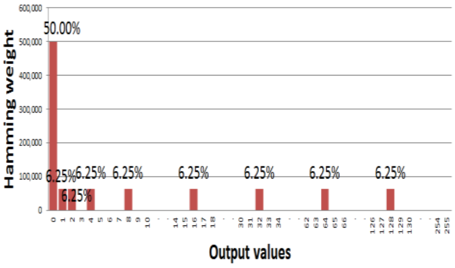


Figure 8: Random Interval

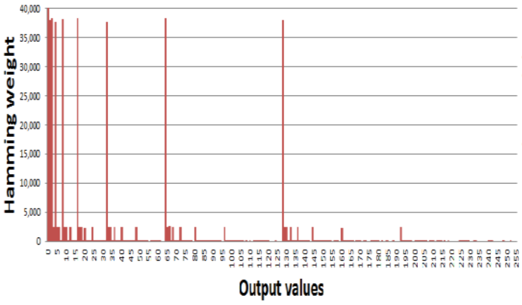


Figure 9: Pseudorandom Permutation

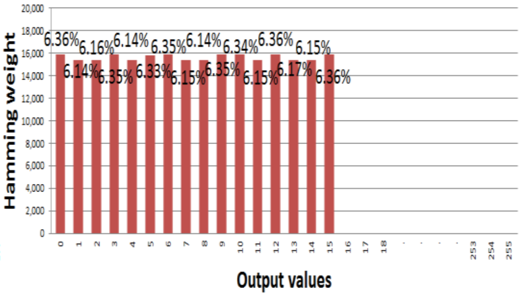


Figure 10: Image Downgrading & Covert Channels

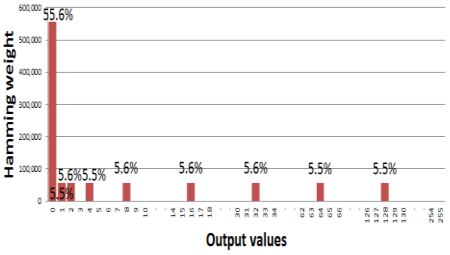


Figure 11: Cover-regions & Parity Bits

Completeness test results are showed in Figure 7,8,9,10 and 11. Least Significant Bit Substitution method produced 2 output values. Random Interval and Cover-regions & Parity Bits methods produced 9 output values. Image Downgrading & Covert Channels method produced 16 output values. Pseudorandom Permutation method produces various output values. Lesser output values will increase the ability to guess the secret message bit position.

C. Analysis Of Hamming Weight According To The Bit Position (Strong Function)

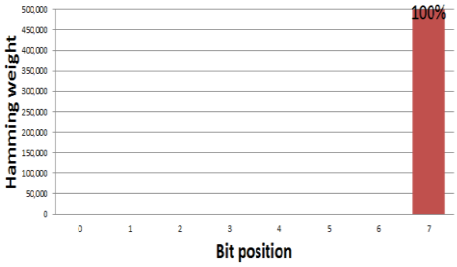


Figure 12: Least Significant Bit Substitution

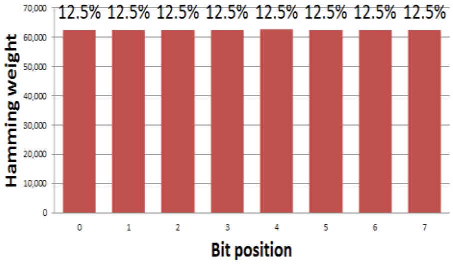


Figure 13: Random Interval

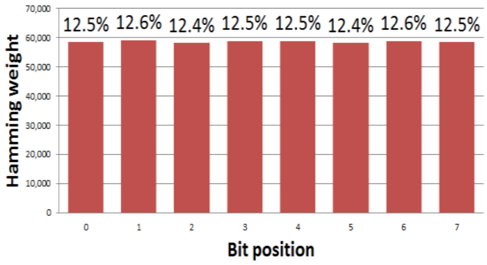


Figure 14: Pseudorandom Permutation

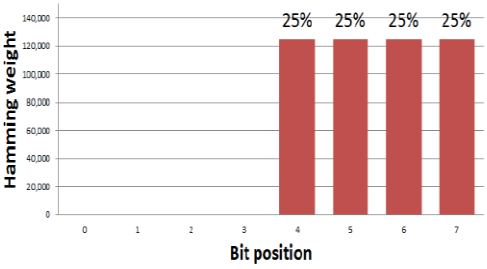


Figure 15: Image Downgrading & Covert Channels

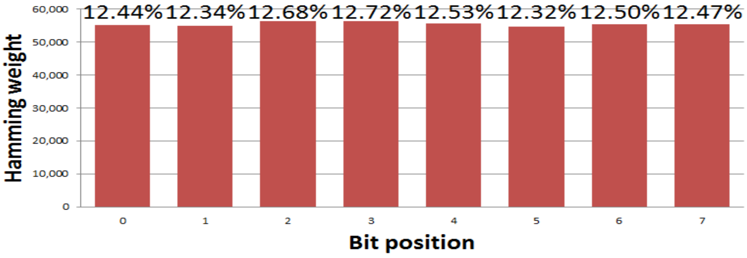
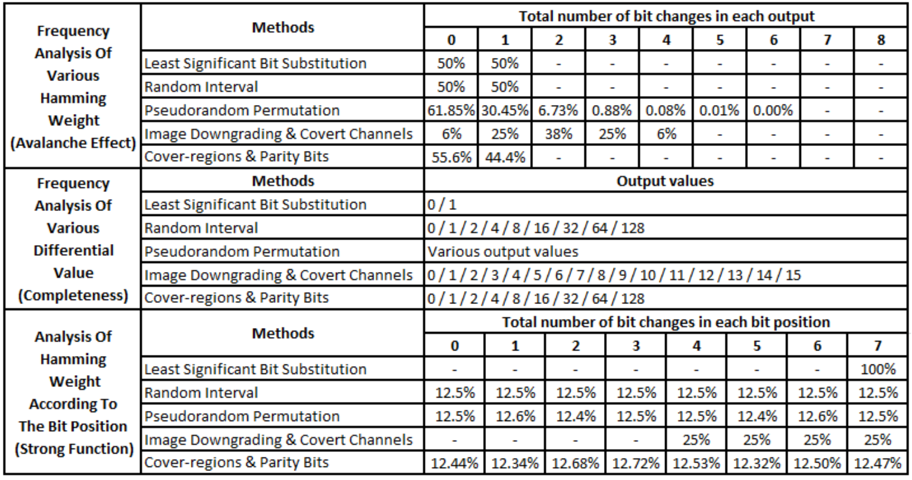


Figure 16: Cover-regions & Parity Bits

In the strong function test, results show that Least Significant Bit Substitution method embeds secret messages in 7th position. Image Downgrading & Covert Channels method embeds secret messages in 4th, 5th, 6th and 7th position. Random Interval, Pseudorandom Permutation, and Cover-regions & Parity Bits methods embed secret messages in all positions. Lesser bit position embeds will increase the ability to guess the secret message bit position. The results are showed in Figure 12,13,14,15 and 16.

6. CONCLUSIONS AND FUTURE WORKS

In this research paper presented the analysis on steganography substitution methods using new testing techniques proposed for strict avalanche criterion. Overall results in Table 1 showed that different methods have different characteristics. All methods are analyzed in detail to determine the best method to be implemented. Random Interval and Cover-regions & Parity Bits methods seem to be the two best methods because both produced good results in all three tests. Results of this analysis can be used to change or improve current embedding methods. Therefore an important consideration in steganography is to have a good randomization method which could improve the security. Motivated by the results derived, we will continue on this research to propose a better substitution system method.



Frequency Analysis Of Various Hamming Weight (Avalanche Effect)	Methods	Total number of bit changes in each output								
		0	1	2	3	4	5	6	7	8
	Least Significant Bit Substitution	50%	50%	-	-	-	-	-	-	-
	Random Interval	50%	50%	-	-	-	-	-	-	-
	Pseudorandom Permutation	61.85%	30.45%	6.73%	0.88%	0.08%	0.01%	0.00%	-	-
	Image Downgrading & Covert Channels	6%	25%	38%	25%	6%	-	-	-	-
	Cover-regions & Parity Bits	55.6%	44.4%	-	-	-	-	-	-	-
Frequency Analysis Of Various Differential Value (Completeness)	Methods	Output values								
	Least Significant Bit Substitution	0 / 1								
	Random Interval	0 / 1 / 2 / 4 / 8 / 16 / 32 / 64 / 128								
	Pseudorandom Permutation	Various output values								
	Image Downgrading & Covert Channels	0 / 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9 / 10 / 11 / 12 / 13 / 14 / 15								
	Cover-regions & Parity Bits	0 / 1 / 2 / 4 / 8 / 16 / 32 / 64 / 128								
Analysis Of Hamming Weight According To The Bit Position (Strong Function)	Methods	Total number of bit changes in each bit position								
		0	1	2	3	4	5	6	7	
	Least Significant Bit Substitution	-	-	-	-	-	-	-	100%	
	Random Interval	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	
	Pseudorandom Permutation	12.5%	12.6%	12.4%	12.5%	12.5%	12.4%	12.6%	12.5%	
	Image Downgrading & Covert Channels	-	-	-	-	25%	25%	25%	25%	
	Cover-regions & Parity Bits	12.44%	12.34%	12.68%	12.72%	12.53%	12.32%	12.50%	12.47%	

Table 1: Overall results

7. ACKNOWLEDGMENT

The main author would like to acknowledge the effort of team members in producing this paper. Not forgetting Cyber Technology Research Department of CyberSecurity Malaysia for support and encouragement. A special thanks is also due to Dr Tuan Sabri b. Tuan Mat for introducing cryptography to the main author and guidance throughout involvement in this field.

REFERENCES

Al-Ani Z. K., Zaidan A. A., Zaidan B. B. and Alanazi H. O. 2010. Overview: Main fundamentals for Steganography. *Journal of Computer.* Vol. 2, No. 3: 158-165.

Bandyopadhyay S.K. and Banik B.G. 2012. Multi-Level Steganographic Algorithm for Audio Steganography using LSB Modification and Parity Encoding Technique. *International Journal of Emerging Trend & Technology in Computer Science (IJETTCS).* Vol. 1, Issue 2: 71-74.

Fridrich J. and Du R. 2000. Secure Steganographic Methods for Palette Images. *Pfitzmann A. (ed.): 2nd International Workshop on Information Hiding. Lecture Notes in Computer Science.* Vol. 1768: 47–60.

Johnson and Katzenbeisser. 2000. A survey of Steganographic techniques. *Information hiding Techniques for Steganography and Digital Watermarking.* 43-78.

Leivaditis M. 2010. Statistical Steganalysis. *MSc thesis, University of Surrey.*

Li Y. and Cusick T.W. 2007. Strict Avalanche Criterion over Finite Fields. *J. Math. Crypt.* Vol. 1: 65-78.

Mar P.P. and Latt K.M. 2008. New analysis methods on strict avalanche criterion of Sboxes. *World Academy of Science, Engineering and Technology.* 48: 150-154.

Rodrigues J.M., Rios J.R. and Puech W. April 2004. SSB-4 System of Steganography using Bit. 5th *International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2004).*

Zaidan B.B., Zaidan A.A., Taqa A. and Othman F. December 2009. Stego-Image Vs Stego-Analysis System. *International Journal of Computer and Electrical Engineering(IJCEE).* Vol. 1, No. 5: 572-578.

A Balanced Secure Cryptographic Algorithm against Timing and Side Channel Attacks

[1]Nur Azman Abu and [2]Amir Hamzah Abd Ghafar
*[1]Faculty of Information and Communication Technology,
Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya,
76100 Durian Tunggal,
Melaka, Malaysia.
[2]Al-Kindi Cryptography Research Laboratory,
Institute for Mathematical Research,
Universiti Putra Malaysia,
43400 Serdang, Selangor, Malaysia
Email: [1]nura@utem.edu.my, [2]amirghafar87@gmail.com*

ABSTRACT

Historically, a computing resource is scarce and expensive. In the last few decades, considerable efforts have been made to design efficient codes in terms of the storage space and running time. Due to the progress on computing resources and cost of memory, an efficient algorithm has ironically become a threat to cryptographic operations. An efficient unbalanced code opens another room for timing and side channel attacks on the private key of public key infrastructure (PKI). This paper shall highlight and propose balanced secure algorithms for cryptographic operations to avoid feasible timing and side channel attacks in the immediate future.

Keywords: Side channel attacks, secure programming

1. INTRODUCTION

Electronic computers have evolved from exiguous experimental enterprises (Denning, 1982) in the 1950s to prolific data processing systems in 2000s. There are two computing resources which computer scientists have been paying attention in the last 50 years. They are space and time. The constraint on the space has been discounted from the last century due to electronic progress and advancement from silicon chip industry. The running time of computational operations is still flexibly upgraded by a more efficient code. Even though, computer science textbooks has clearly outlined the efficiency of running time in terms of asymptotic big-oh notation, the real applications still quest for more efficient average running time with regards to its constant factors.

This quest for efficient running time has become meaningless in the realm of microseconds per transaction. Still there is an argument for batch processing such as check clearance which runs few millions per hour. The same quest has been welcomed in cryptography. An efficient algorithm has always been accepted as an economic progress. Theoretically, a more efficient algorithm which saves even 50% of running time such as the ones proposed by (Othman, 2008) and (Koval, 2010) or by a constant factor is not a progress at all especially in cryptography. Unless it is asymptotically better such as a reduction from $O(n^3)$ down to $O(n^2 \log n)$, the saving by a constant factor such as from $O(5n^3)$ down to $O(2n^3)$ is not a considerable progress at all. The savings by a constant factor may easily come from the necessary redundancy to keep the code running with balanced operations. It is crucial to keep the balanced computational operation in cryptography between bit 0 and bit 1 during a decryption process while using the private secret key.

This vulnerable phenomenon is due to an attack called side-channel attack by Paul (Kocher, 1996) which uses the timing difference observed from modular exponentiation process to determine the secret exponent used. Modular exponentiation algorithm is commonly used in various cryptosystems including Diffie-Hellman, DSS and RSA. Usage of physical indicator, in this case computational time; makes this attack quickly expanding. Next, Paul (Kocher *et al.*, 1999) orchestrated an attack concentrated on computational power. It used the same weakness in unbalanced modular exponentiation algorithm which will be discussed in the next few sections. Other than computational time and power, other external factors such as transmission methodology by cryptographic hardware have also been used in side channel attack such as cache response (Osvik *et al.*, 2006) and electromagnetic radiation (Gandolfi *et al.*, 2001).

Daniel (Genkin *et al.*, 2013) focuses on a source of noise such as the vibration of electronic components in the computer. During a decryption process, the noise sometimes heard as a faint high-pitched tone or hiss often generated by capacitors and correlated with system activity since CPUs drastically change their power draw according to different type of operations they perform. The key extraction attack finds the secret key bits sequentially one by one. It is sufficient to differentiate between the pitch during the decryption process when the bit is 0 and 1. Since the CPU is running different operation during the short period of bit 0 from bit 1, naturally, the computer will emit different sound signals. Theoretically, this attack is feasible due to an imbalance algorithm deep in the internal details of GnuP's implementation of RSA (Rivest *et al.*, 1978).

2. FREQUENCY SAMPLING

According to Nyquist Theorem, 2 samples per cycle of the input signals should properly define the target signals. During the decryption process when the bit is 1, the computing device produce higher pitch sounds. If this pitch sound can be accurately measured and identified at least at the level of twice the frequency signals, the attacker will be able to differentiate the pitch during the decryption process when the bit is 0 or 1. The report Daniel (Genkin *et al.*, 2013) gives a sample set of a strong component at 35.2 kHz and 38.1 kHz when the secret bits are 0 and 1 respectively. A minimum of sampling frequency at 96 kHz is required to observe such signal components. It should be noted here that a sampling frequency comes in a multiple of 44,100 Hz from Audio CD or 48,000 Hz the standard audio sampling rate used by most professional digital video equipments. The next available high sampling rate is 192 kHz. It is sufficient to analyse the residue details of computing devices during the decryption or digital signing process. The sampling frequency goes up to 5,644,800 Hz for the Super Audio CD (Reefman and Nuijten, 2001).

3. AN EFFICIENT UNBALANCED ALGORITHM

An efficient algorithm which produces an accurate output does not imply the algorithm is secure. In order to be secure the algorithm must be balanced on the different critical input especially on the secret element the algorithm supposed to protect at all computational cost such as the private key. An efficient classic algorithm which runs right-to-left on the exponent of power modulo is pseudo coded in Algorithm − ***A*** below. Algorithm – ***A*** is widely used in implementation of RSA due to its efficiency.

```
Algorithm − $\boldsymbol{A}$ : PowerMod ($a$, $b$, $N$)
Let $b$ = $b_0 b_1 b_2 \cdots b_{n-1} = \sum_{i=0}^{n-1} b_i \cdot 2^i$  written in big-endian
$L$= $a^0$ = 1, $R$ = $a^1$= $a$.
for $i$ = 0 to $n$−1 do
        if $b_i$ = 1 then
                $L = L \cdot R$ mod $N$
        end( if-then )
        $R = R^2$ mod $N$
end( for )
return($L$).
```

This is an efficient textbook algorithm. It is necessary to have square mod operation on each bit of the exponent. However, it is only sufficient to have multiply mod operation whenever the exponent bit is one. Algorithm − ***A*** will be complemented with Montgomery reduction (Montgomery, 1985) in most of RSA implementations. Montgomery happens every time there is a multiplication of $L{\cdot}R$ mod N and $R = R^2$ mod N. This complementary algorithm is shown below.

Algorithm – ***A1*** : Montgomery (a, b, N)

```
Let $K=2^n$ for $n \in \mathbb{Z}$
$a' = a{\cdot}K$ mod $N$, $b' = b{\cdot}K$ mod
$K{\cdot}K' - N{\cdot}N' = 1$
$z = a'{\cdot} b'$
$r = (z$ mod $R)(N'$ mod $R)$
$s = (z+r{\cdot}N) = R$
if $s \geq N$ then
        $s = s - N$ //extra reduction process
end ( if-then )
return (s).
```

Usage of Montgomery reduction will result in multiplication of two numbers that are about the same number of bits. This triggers implementers to use Karatsuba (Karatsuba and Ofman, 1962) multiplication as this type of multiplication is effective on these numbers. Unfortunately, the difference in the processing power required to have or not to have multiply mod operation can certainly be acoustically detected.

4. A BALANCED SECURE ALGORITHM

In this section we will discuss the following algorithm:

Algorithm – *B* : Target Sum(b)

```
Let $b = b_0 b_1 b_2 \cdots b_{n-1} = \sum_{i=0}^{n-1} b_i \cdot 2^i$ written in big-endian
$L = 0$, $R = 1$.
for $i = n-1$ downto 0 do
        if $b_i = 0$ then
                $R = L+R$
                $L = 2L$.
        end( if-then )
        if $b_i = 1$ then
                $L = L+R$
                $R = 2R$.
        end( if-then )
end( for )
return($L$).
```

The security of the cryptographic system should not depend on the secrecy of the algorithms nor apparatus being used. They shall be made public and open to all users (Abu and Sahib, 2010). The algorithms operating on secret private keys must be securely balanced in order to operate in open environment. It should be noted that the Algorithm − ***A*** is a popular cryptographic codings in practice. It is not balanced hence rendering them insecure and vulnerable to side channel attack in an open environment. Alternatively, an open balanced algorithm is called for here. The Algorithm – ***B***, however, is the basic left-to-right algorithm (Levitin, 2012) which has been neglected and discounted on its importance. The algorithm presented above carry the same operation regardless of the binary of the secret key whether it is zero or one.

4.1 Algorithm – *B* and Elliptic Curve Cryptosystem

It is very natural to adjust Algorithm – ***B*** to compute point projection in an elliptic curve cryptosystem. Neal Koblitz and Victor S. Miller have independently first proposed the use of elliptic curves for cryptography at about the same time in 1986. A sample on point projection is written in Algorithm – ***C*** below. The initial value zero has been replaced by an identity point zero at infinity. The addition *L*+*R* and doubling have been replaced by point addition and point doubling respectively.

```
Algorithm – C: Point Projection ($\lambda$, $P$)

Let $\lambda = b_0b_1b_2\cdots b_{n-1} = \sum_{i=0}^{n-1} b_i \cdot 2^i$ written in big-endian
$L = \mathbf{0}$, $R = P$,   where $\mathbf{0}$ is an identity point zero at infinity
for $i = n-1$ down to 0 do
        if $b_i = 0$ then
                $R$ = AddPoint($L$, $R$)
                $L$ = DoublePoint($L$)
        end( if-then )
        if $b_i = 1$ then
                $L$ = AddPoint($L$, $R$)
                $R$ = DoublePoint($R$)
        end( if-then )
end( for )
return($L$).
```

4.2 Algorithm – *B* and RSA

Next, Algorithm **–** ***B*** shall be used to do power mod operation in RSA encryption or decryption (Rivest *et al.*., 1978). In this case, however the exponent shall be the binary sequence $b = b_0b_1b_2\cdots b_{n-1}$. As tailored in Algorithm **–** ***D***, the initial value zero has been replaced by one. The addition operation *L*+*R* and doubling have been replaced by multiplication modulo *N* and square modulo *N* respectively.

```
Algorithm – D: PowerMod ($a$, $b$, $N$)

Let $b = b_0b_1b_2\cdots b_{n-1} = \sum_{i=0}^{n-1} b_i \cdot 2^i$ written in big-endian
$L = a^0 = 1$, $R = a^1 = a$.
for $i = n-1$ downto 0 do
        if $b_i = 0$ then
                $R = L\cdot R \bmod N$
                $L = L^2 \bmod N$
        end( if-then )
        if $b_i = 1$ then
                $L = L\cdot R \bmod N$
                $R = R^2 \bmod N$
        end( if-then )
end( for )
return($L$).
```

4.3 Algorithm – *B* and the General Lucas Sequences

The LUC cryptosystem (Smith and Lennon, 1993) has been designed based on the general Lucas functions. Therefore, the security of this cryptosystem is analogous to the RSA cryptosystem. It relies on the difficulty of factoring *N* back into its prime factors *P* and *Q*. Similarly, an LUC cryptosystem is also vulnerable to side channel and timing attacks. The encryption and decryption processes are similar to power modulo operation with minor adjustment.

Let (p, q) be nonzero integers, and let α and β be the two complex roots of the quadratic polynomial $x^2 - px + q$. Then the general Lucas sequences (U_n, V_n) satisfy the recurrence relations

$U_0 = 0$, $U_1 = 1$ and $U_{n+1} = p\,U_n - q\,U_{n-1}$,
$V_0 = 2$, $U_1 = p$ and $V_{n+1} = p\,V_n - q\,V_{n-1}$.

The Lucas sequences may be written in closed forms as

$$U_n = \frac{\alpha^n - \beta^n}{\alpha - \beta} \text{ and } V_n = \alpha^n + \beta^n$$

Typically, *q* is set to be one. There is only one parameter *p* in computing the Lucas sequences. Even though the sequence *V* is more popular for LUC cryptosystem, here in Algorithm – ***E*** below, both sequences *U* and *V* are being written for better references. While the product of public and private exponents; $ed \equiv 1 \bmod (P-1)(Q-1)$ in RSA cryptosystem, it is $ed \equiv 1 \bmod (P^2-1)(Q^2-1)$ in the LUC cryptosystem. The encryption process of this system is the computations of *V*(*e*), while the decryption process is done by the computations of *V*(*d*). The *V*(*e*) and *V*(*d*) are both Lucas functions (Ali *et al.*, 2008).

Algorithm – ***E***: General Lucas Sequences UV(*b*, *p*, *N*)

```
Let b = $b_0 b_1 b_2 \cdots b_{n-1} = \sum_{i=0}^{n-1} b_i \cdot 2^i$ written in big-endian
$UL = 0$, $UR = 1$, $VL = 2$, $VR = p$,

for $i = n-1$ down to 0 do
    if $b_i = 0$ then
        $UR = UR \cdot VL - 1 \bmod N$
        $UL = UL \cdot VL \quad \bmod N$
        $VR = VL \cdot VR - p \bmod N$
        $VL = VL^2 - 2 \quad \bmod N$

    end( if-then )
    if $b_i = 1$ then
        $UL = UR \cdot VL - 1 \bmod N$
        $UR = UR \cdot VR \quad \bmod N$
        $VL = VL \cdot VR - p \bmod N$
        $VR = VR^2 - 2 \quad \bmod N$

    end( if-then )
end( for )
return($UL$, $VL$).
```

The idea is as follows: Let $b = (b_{n-1} \ldots b_1 b_0)_2$ in binary and begin with the ordered pairs $(U_0, U_1) = (0, 1)$ and $(V_0, V_1) = (2, p)$. Moving from left to right in the binary representation of *b*, having the pair

(U_m, U_{m+1}) we compute either (U_{2m}, U_{2m+1}) and (V_{2m}, V_{2m+1}) (if the bit b_i is a 0) or (U_{2m+1}, U_{2m+2}) and (V_{2m+1}, V_{2m+2}) (if the bit b_i is a 1). This process will terminate with the ordered pair (U_b, U_{b+1}) and (V_b, V_{b+1}).

5. DISCUSSION

An individual CPU operation clocked at several Gigahertzes is too fast for a high fidelity microphone to sample and digest. However, long operations such as RSA modular exponentiation do create distinctive acoustic spectral characters over few milliseconds. In this paper, we have highlighted algorithm for common public key cryptography such as RSA, ECC and LUC that carry the same operation regardless of the binary of the secret key whether it is zero or one. Popular cryptographic codings in practice, however, are not balanced hence rendering them unsecure and vulnerable to side channel attack in an open environment. It is just a matter of time it will happen since the available frequency sampling is already there to harness and harvest the private key.

6. CONCLUSION

An efficient unbalanced code opens another room for timing and side channel attacks on the private key of public key infrastructure. An inexpensive cheap high fidelity digital audio has been well developed about the same time of PKI. Sufficiently high frequency sampling has become a new threat to the unbalanced cryptographic codes running on small devices. A few hundred kHz frequencies sampling, using ultrasound microphones in several orders of magnitude relative to the GHz-scale clock rates of the attacked computers, is sufficient to capture reminiscent of decryption process. It is also just a matter of time a more advanced audio processing operating on higher frequency becomes cheaply available poses serious threat to these unbalanced cryptographic codes running on regular computers. This paper has proposed balanced secure algorithms for cryptographic operations to avoid feasible timing and side channel practical attacks in the immediate future.

REFERENCES

Abu, N. A. and Shahrin, S. 2010. Random Ambience Key Generation Live on Demand, *Proceedings 2nd International Conference on Signal Processing Systems (ICSPS 2010)*. 1: 110-114.

Ali, Z. M., Othman, M., Said, M. R. M., and Sulaiman, M. N. 2008. An Efficient Computation Technique for Cryptosystems Based on Lucas Functions, *Proceedings of the International Conference on Computer and Communication Engineering*. 187-190.

Denning, D. E. 1982. *Cryptography and Data Security*. Addison-Wesley, preface page v.

Gandolfi, K., Mourtel, C. and Olivier, F. 2001. Electromagnetic analysis: Concrete results. *Procedings of CHES ’01*: 251–261.

Genkin, D., Shamir, A. and Tromer, E. 2013. RSA Key Extraction via Low-Bandwidth Acoustic Cryptanalysis. Accessed 27th March 2014. Sourced from http://eprint.iacr.org/2013/857.pdf

Karatsuba, A. A. and Ofman, Y.P. 1962. Multiplication of Many-Digital Numbers by Automatic Computers. *Proceedings of the USSR Academy of Sciences*: 293–294.

Koblitz, N. 1987. Elliptic Curve Cryptosystems. *Mathematics of Computation*. 44(177): 203–209.

Kocher, P. C. 1996. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. *Proceedings of CRYPTO '96*: 104-113.

Kocher, P. C., Jaffe, J. and Jun, B. 1999. Differential Power Analysis, *Proceedings of CRYPTO '99*: 388-397.

Koval, A. 2010. On Lucas Sequences Computation. *International Journal of Communications, Network and System Sciences*. 3: 943-944.

Levitin, A. 2012. *Introduction to the Design and Analysis of Algorithms.* Pearson.

Miller, V. S. 1985. Use of Elliptic Curves in Cryptography. *Proceedings Advances in Cryptology (CRYPTO ’85) Lecture Notes in Computer Science*. 218: 417-426.

Montgomery, P. L. 1985. Modular Multiplication without Trial Division. *Mathematics of Computation.* 44(170): 519-521.

Osvik, D. A., Shamir, A. and Tromer, E. 2006. Cache attacks and countermeasures: the case of AES. *Proceedings of CT-RSA 2006*: 01–20.

Othman, M., Abulhirat, E. M., Ali, Z. M., Said, M. R. M. and Johari, R. 2008. A New Computation Algorithm for a Cryptosystem Based on Lucas Functions. Journal of Computational. 4: 1056-1060.

Reefman, D. and Nuijten, P. 2001. Why Direct Stream Digital is the Best Choice as A Digital Audio Format. *110th Audio Engineering Society Convention*.

Rivest, R. L., Shamir, A. and Adelman, L. 1978. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Communications of ACM*. 21(2): 122-126, 1978.

Smith, P. J., Lennon, G. J. J. 1993. LUC: a new public key cryptosystem, *9th IFIP Symposium on Computer Science Security*: 103-117.

Investigation of Medical Identity Theft in Healthcare Sector: Its Statistics, Effects and Way Out

[1] Rashidah. F. Olanrewaju, [1]Muktar Yahuza, [2]Nor'ashikin Bte Ali and [1]Othman Khalifa
[1]*Department of Electrical & Computer engineering, Faculty of Engineering, International Islamic university Malaysia*
Kuala Lumpur Malaysia,
[2]*College of Information Technology, Universiti Tenaga Nasional Malaysia*
Email: [1]frashida@iium.edum.my

ABSTRACT

Information Technology is reaching all angles of the healthcare sector in the form of electronic diagnosis, surgery consultation and medical records. With the current trends in information and communication technology in healthcare system, such as the cloud computing and ubiquitous computing, where small computers are being embedded in almost every day object around us using both mobile devices as well as wireless connections for recording, storage and delivery of medical data, the possibly of attack on these records becomes more likely. As a result, intruders has the potential to alter, steal or destroy individual medical or health insurance records, alter computer-based prescriptions at pharmacies to life-threatening doses, or make private medical condition public. Consequently, leading to negative social and psychological effects on the affected individuals. The medical identity theft may lead to wrong diagnosis and wrong decision taken on the patient, some of which may be a life threatening cases. In addition, it cost some hospitals billions of law suits. Therefore, medical identity theft (MIDT) in healthcare needs to be investigated. In this paper, MIDT statistics is explored, it's effects and the possible ways of solving the MIDT problem.

Keywords: Medical Identity theft, medical fraud, healthcare security, Authentication, Telemedicine, Telehealth

1. INTRODUCTION

Medical identity theft occurs when someone uses your personal health information without your knowledge or consent to obtain, or receive payment for medical treatment, services or goods. It can take place whether the victim is alive or deceased (AHIMA 2014). The victims of medical identity theft are the patients, and other health personnel; mostly include the physicians and nurses. The hackers may steal the medical records of a patient and sell them to others in order to; obtain healthcare services of the victims, obtain pharmaceuticals or other medical equipment, obtain governmental benefits, bill the healthcare plan, insurance company or government program, or they may alter the record in other to claim the patient's future medical benefits. Sometimes, MIDT causes the victims not only to lose their financial assistances, but also to suffer from wrong diagnosis which may lead to their death.

Medical identity theft is seen to be increasing every year. The most recent survey conducted by Identity theft resource center (ITRC) on February, 2014 reported a dramatic increase in medical identity theft from the year 2012 to the year 2013, which was an increase of 8.9% over the medical breaches that occurred in 2012 (Updated ITRC, 2013). A typical example of MIDT is electronic health records of the United States which was stolen in 2004 and was found on a computer server in Malaysia. It is controlled by cyber criminals. The stolen files included names of health care providers, social security number, birthdates and addresses of the patients. Criminals may create false billing, which can bring in millions of dollars from stolen health records. The discovery of the stolen records has revealed the vulnerability of electronic medical records, and can cause more damage than the loss of money to false billing. When cyber criminals alter a patient's medical records, the results could be potentially deadly (Berwin, 2008)

Besides, Ponemon Institute revealed that 1.52 million Americans were affected by MIDT in the year 2012, which rises to 1.84 million in the year 2013. This is almost 32% increase in just one year (Ponemon Institute 2013). Furthermore, it also reported that the growing of medical identity theft is estimated to have affected as many as 18, 36312 people in the year 2013. These victims had their information used to receive medical care, benefits or insurance. Wrong information in health file can lead to negative consequences to affected victims. False entries on medical files can trench an individual's medical coverage and, in some instances, make them uninsurable (having a disease that is not yours). Sometimes, it can lead to victim to be unemployable, e.g. if it contain psychiatric history. This may not be discovered until incorrect medical treatment or outstanding bills appeared in the victim's file. Unlike credit report, patients do not have the same rights to correct errors in their medical histories; nor do they have the right to receive a free copy of their medical file (as one would do with credit card report). In an event reported, a teenager was denied the opportunity to donate blood because the Red Cross marked her social security number as belonging to a person who had tested positive for HIV (Rick, 2012).

Eva Velasquez, President and CEO of the Identity Theft Resource Centre (ITRC) has reported that "Medical identity theft has vast potential to dramatically increase due to a combination of the Affordable Care Act adding millions of newly insured individuals, HIPAA/HITECH policies encouraging the use of electronic health records that are valuable targets to data thieves, and the ever-increasing sophistication of data hackers stealing medical information to sell on the black market" (ITRC, 2013). Hence, it is important to assess the public's perception of medical identity theft and the risks associated with said crime.

The rest of the paper is organize as follows: statistics of Medical Identity theft is explored in section 1, Effect of Medical Identity theft is given in section 2, curbing the MDT problems is extensively discussed section 3, and section 4 concludes the papers.

2. STATISTICS OF MEDICAL IDENTITY THEFT

Acquiring of information about Medical Identity theft comes in different ways, according to a survey conducted by the Canadian Healthcare Anti-fraud Association in 2004 which involved 109 senior healthcare insurance professionals and claims processors. It was pertinent to note that about 95% were healthcare claim fraud victims, and half had over 30 incidents of fraud. 77% of the fraud were detected through a claims review process. 69% detected fraud were from external tips, 87% indicated that Health care providers were responsible for the fraud, while 9% indicated that individual were responsible for the fraud (Canadian Healthcare Anti-fraud Ass. 2004).

In a similar survey conducted by Fair Warning Inc. in October 2011, among the 1,002 Canadian correspondents, 3.7% have been data breach victims of personal medical Identity information. Among the victims, 57% were negatively impacted. 11% of the victims had inaccurate medical records (Fair Warning 2011). Equally, a survey conducted by PWC Health Research Institute in 2011, in which 600 executives from U.S.A hospitals, physicians, health insurers, and pharmaceutical companies were involved. The survey revealed that 36% of provider organizations had experienced medical identity theft (PWC Health Research Institute 2011). Likewise, a survey by Nationwide Insurance in 2012 out of the 2,001 adult respondents, only 15% of insured adults say they are familiar with medical identity theft. Out of this 15%, only few could correctly define Medical identity theft (Nationwide Insurance, 2012).

Further evidence of the significance of the medical fraud problem is the allocation of $1.7 billion for fraud detection in the 2011 by U.S. Health and Human Services Department budget (HHS budget, 2010)

The 6th annual security report of the Healthcare Information and Management System Society (HIMSS) released on 19th February 2014, revealed that about 80% of the responded contacted stated that almost all of the threats to patient's data is compromised by the staffs of an organization (HIMSS, 2014).The response given by the responded was recorded on a scale of 1 to 7 (one to seven), where 1 is when there is no threat and 7 are when the threat is very high. It revealed that, human related factors leading to medical threats is the greatest among the other factors stated. It was estimated to be at a scale of

5.64, while the least factor to healthcare threat is that due to the loss of integrity of information which was estimated to be at a scale of 4.32. The table 1 below summarizes the finding

FACTOR	SCALE
Loss of integrity of information	4.32
Functionality of devices	4.36
Lack of planning, policies and procedures	4.51
Infiltration attacks	4.56
Virus or malware software	4.66
Human related factors	5.64

Table 1: Factors posting threats in healthcare data

Moreover, in the HIMSS survey, respondents were also asked to determine what are the motivators and influencers that provoke medical data threats. 80% of the respondents identified that the key motivators are the work force members snooping information of others. 51% stated that the key motivators to the threat are medical identity theft, while 2/3 of the respondents identify that the key motivators to the threat is financial identity theft.

Table 2 below shows the statistics of the most common motivators to medical data threats:

MOTIVATOR	PERCENTAGE (%)
Workforce member snooping	80
Financial identity theft	67
Medical identity theft	51
Outsourced personnel snooping	23
Cyber terrorism	16
Black market activities	11
Intellectual property theft	8
Business espionage	4
Others	2

Table 2: The most common medical data threat motivators

Furthermore, according to a survey conducted by Ponemon institute dated September, 2013 estimated the average cost of medical identity theft per victim to be $18,660 in the USA (Ponemon Institute, 2013).

2.1 Awareness about MIDT

In a survey conducted by Ponemon in comparison of how victim learn about MIDT was compared for 2013 and 2012. It was reported that victims initially learn about the medical identity theft upon encountering of uncovered mistakes in their health records (29% in 2013 and 32% in 2012) as shown in Figure 1. Others are by receiving of letter (24% in 2013 and 39% in 2012) or errors posted to medical invoices by their healthcare provider (25% in 2013 and 26% in 2012). Some victims are provided with information by healthcare provider during visit or medical treatment which as 13% in 2013 with no any record in 2012. Contrary entry on their credit score (11% in 2013 and 15% in 2012), some victims have errors forwarded to their medical invoices of 25% in 2013 while the value was 26% in 2012. It was also noted that 6% of the victims received an alert from a healthcare provider in 2013 while 10% was recorded in 2012. Meanwhile 3% received data breach notification 2013 in comparison to 4% in 2012. The Figure 1 summarizes the result of means of how victims learn about MIDT.

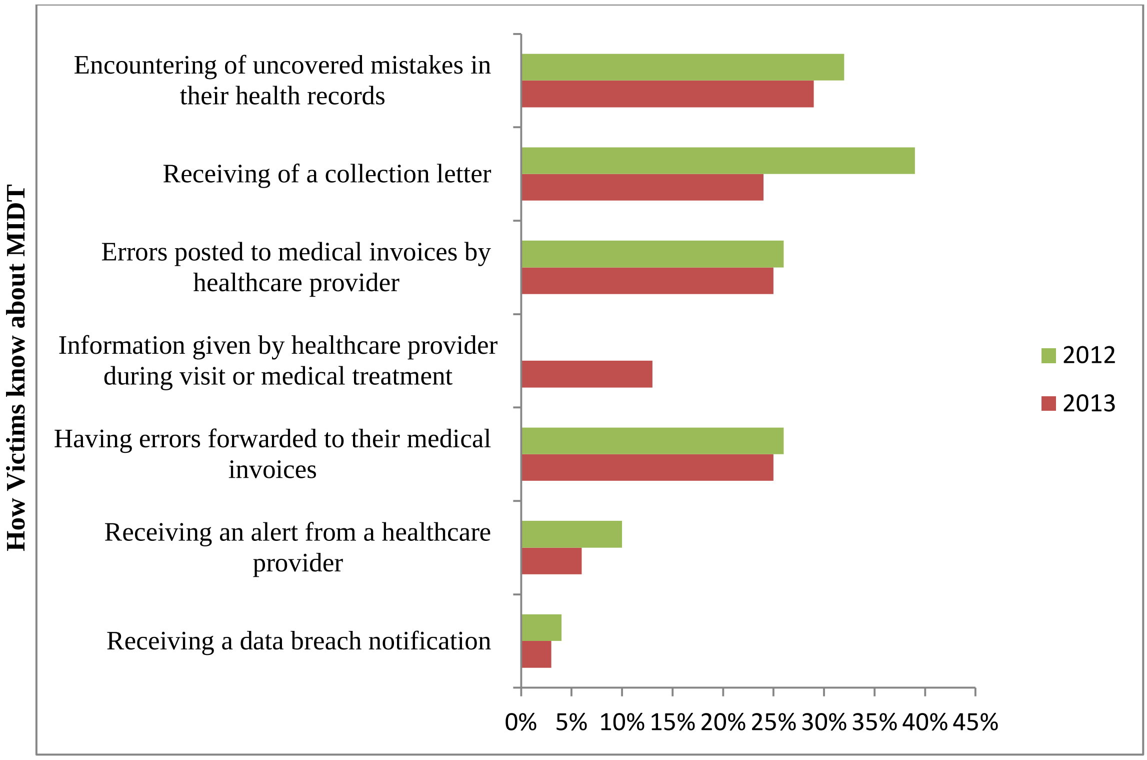


Figure 1: How patient learn about MIDT

This survey resulted in some hazardous consequences on the affected victims which are; out-of-pocket payments (43% in 2013 and 47% in 2012), loss of health insurance as a result of MIDT (39% in 2013 and 41% in 2012), increased health insurance payments as a result of inaccuracies in health records (7% in 2013 and 8% in 2012), lost trust and confidence in my healthcare provider (56% in 2013 and 51% in 2012), misdiagnoses of illness because of inaccuracies in health records (15% in 2013 and 12% in 2012), wrong pharmaceuticals prescribed (11% in 2013 while there is no record in 2012), termination by plan or provider (15% in 2013 and 12% in 2012), diminished credit score (20% in 2013 and 21% in 2012), legal fees (17% in 2013 and 15% in 2012), or loss of time and productivity trying to fix inaccuracies in credit report (23% in 2013 and 19% in 2012),

Figure 2 summarizes the effect of MIDT on the victims from 2012 to 2013:

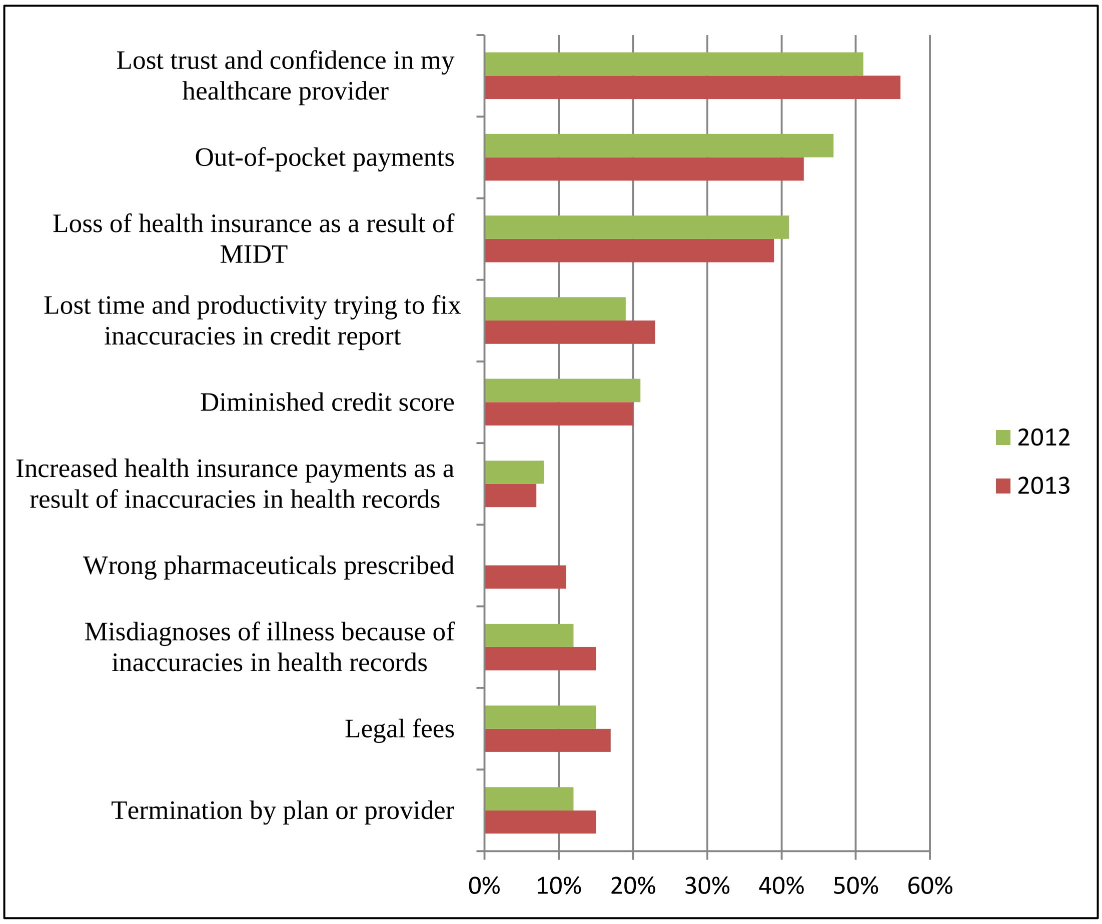


Figure 2: Effects of MIDT on the affected Victims

In addition, the survey enumerated that the majority of respondents said the crime happened because they knowingly shared their personal identification or medical credentials with someone they knew (30%), 28% of the survey respondent claimed that a member of the family took their personal identification or medical credentials without their consent. Personal information was accidentally provided to a fake email or spoofed website amounted to 8% of the survey. Health care provider, insurer or other related organization had a data breach of 7%. An employee working in the healthcare provider’s office stole health information (5%), the identity thief intercepted a mailed statement or invoice (5%), Lost their wallet containing personal identification or medical credentials, and lastly, 14% said they do not know how it happened. The bar chart of Figure 3 summarizes the result obtained:

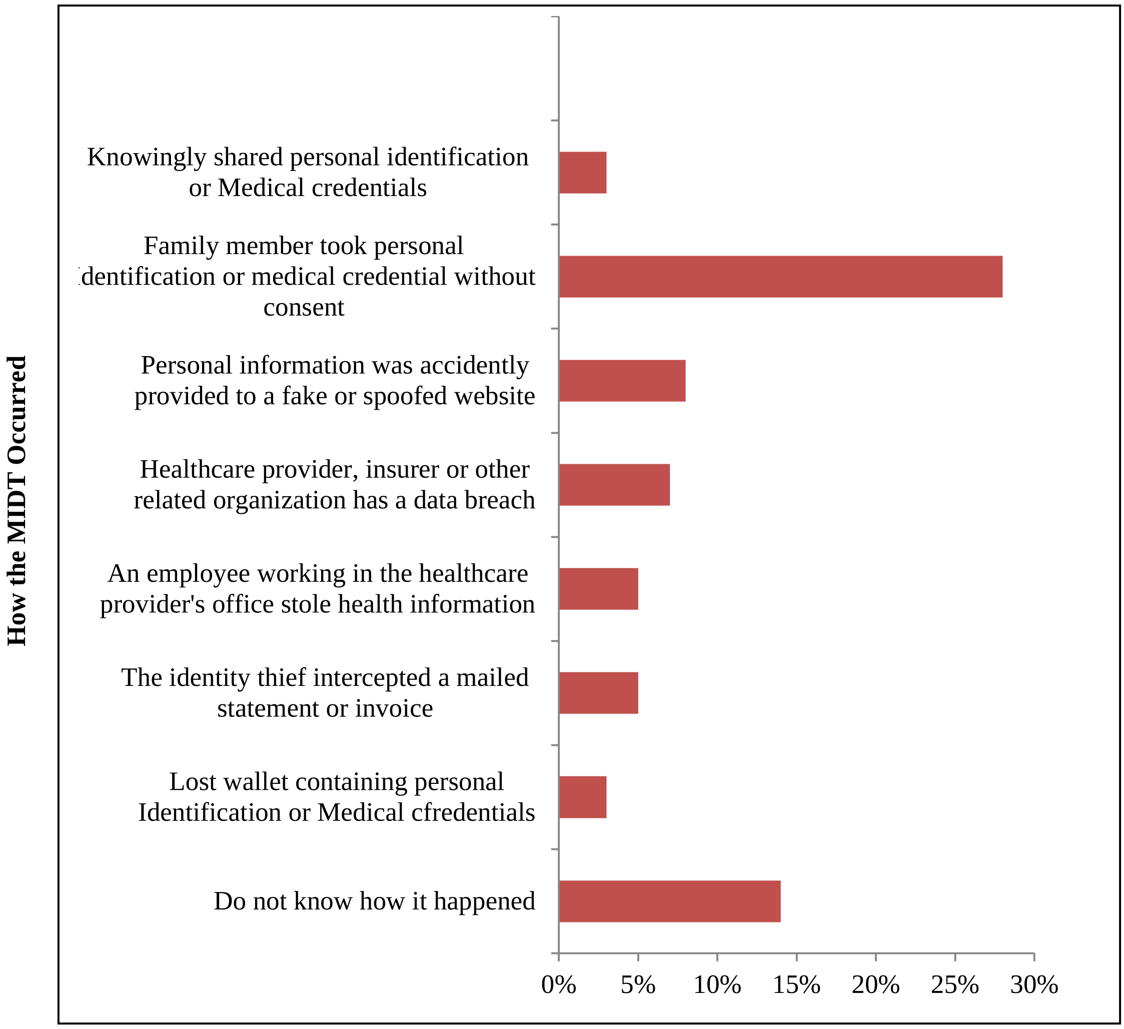


Figure 3: Causes of MDT

The objectives of the Medical identity theft criminals was to obtain healthcare services or treatments (23% in 2013 and 19% in 2012), to obtain prescription pharmaceuticals or medical equipment (23% in 2013 and 19% in 2012), to obtain government benefits including Medicare (23% in 2013 and 19% in 2012), for Healthcare records to be accessed or modified (23% in 2013 and 19% in 2012), for Credit report to be accessed or modified, or for the thief to obtain fraudulent credit accounts (23% in 2013 and 19% in 2012).

The bar chart of figure 4 summarizes the intentions of medical identity criminals:

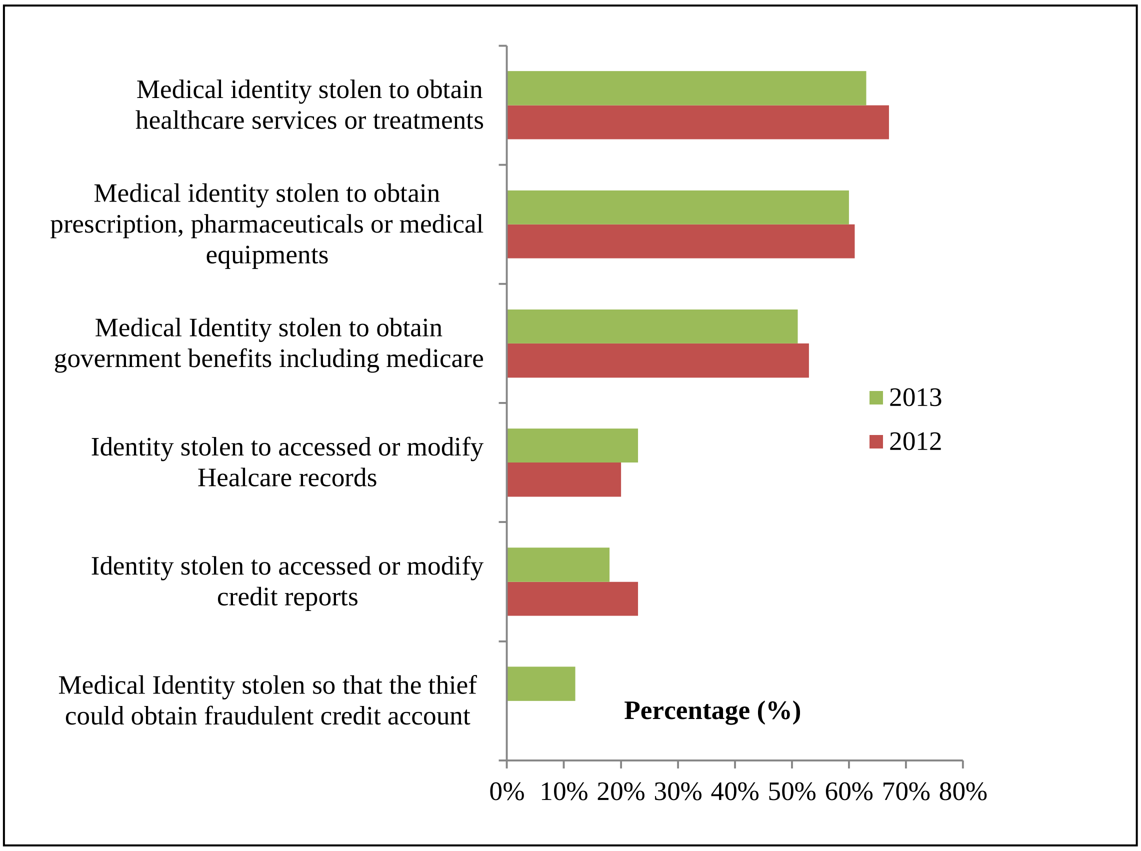


Figure 4: The Intention of Medical Identity thief

When asked the respondents during the survey, who were the subject of the crime? 45% said they were the victims, 22% believed it was their spouse, 20% indicated that their parents was the target, 10% said it was their dependent child, and 3% assumed it was another family member. The pie chart of figure 5 summarizes the results:

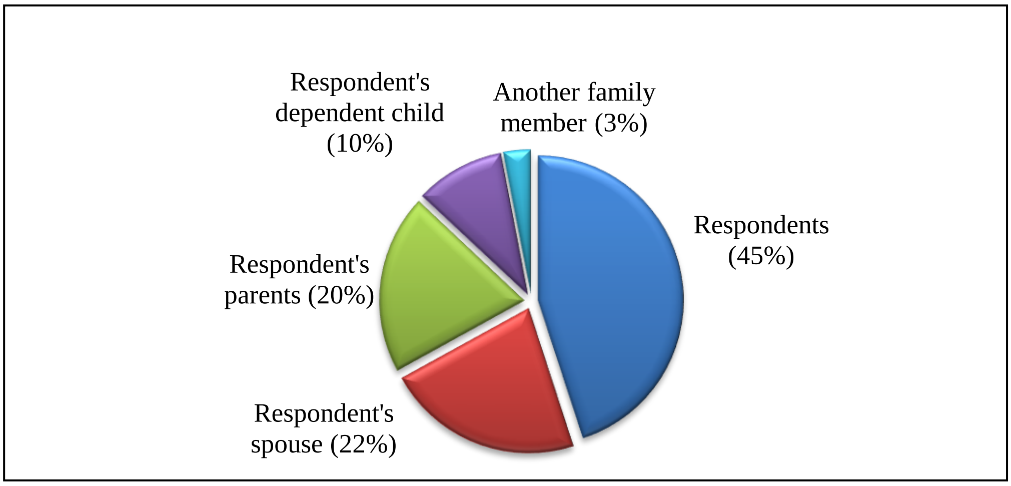


Figure 5: Medical Identity Victims

Furthermore, according to survey recently conducted by the ITRC which provide the top 600 breach list in 2013, data breach information involving five (5) industrial sectors was explored (Updated

ITRC 2013). The sectors include: Business, Education, Government, Healthcare, and Financial. The table 3 below summarizes the findings from the year 2009 to the year 2013:

Industrial Sector	No. of breaches in 2009	No. of breaches in 2010	No. of breaches in 2011	No. of breaches in 2012	No. of breaches in 2013	% of total breaches in 2012 (%)	% of total breaches in 2013 (%)
Education	78	65	60	65	55	13.7	9.0
Healthcare	65	160	87	165	269	34.9	43.8
Business	208	279	198	172	211	36.4	34.4
Financial	57	54	28	18	23	3.8	3.7
Government	90	104	48	53	56	11.2	9.1

Table 3: The breach statistics from 2009 to 2013

It can be seen from the above table that, healthcare sector is having the highest percentage of the in the year 2013 which accounted for 43.8% of the total breaches on the list. This is due to the fact that high number of breaches is reported to the department of healthcare. The number of the recorded breach in the other sector is having the lower percentage because there is no enough report on the number of breaches that occurred.

Additionally, the survey also stated the circumstance under which the breach occurred. The circumstances are: Hacking, Accidental exposure, Third party, Employee negligence, Employee theft, and Data on the move. Table 4 summarizes the circumstances that led to the breaches from the year 2009 to the year 2013:

Category	No. of breaches in 2009	No. of breaches in 2010	No. of breaches in 2011	No. of breaches in 2012	% of total breaches in 2012 (%)	No. of breaches in 2013	% of total breaches in 2013 (%)
Hacking	97	113	108	128	27.3	160	26.1
Accidental Exposure	59	71	44	41	8.7	46	7.5
Third party	37	58	32	53	11.2	88	14.3
Data on the move	78	110	76	57	12.1	80	13.0
Employee theft	85	102	56	40	8.5	72	11.7
Employee negligence				33	7.0	57	9.3

Table 4: The Circumstances that resulted to the breaches

It is pertinent to note that hacking emerged the first type of breach ever had. It constitutes more than ¼ of the total breaches occurred in the year 2013. The second type of breach on the list is that due to the third party which constituted 14.3%, followed by the breach due to the data on the move constituting 13%, breach due to the employer theft constituted about 11.7%, that due to employer negligence is at 9.3, while the last breach with the less record is that due to the accidental exposure which about 7.5% of the recorded breaches.

3. EFFECT OF MEDICAL IDENTITY THEFT

There is a great improvement in healthcare sector with the latest emergence of electronic health records (EHRs) and also healthcare portal where patients and providers can easily share and access their medical information. However, it also paves way to a lot of hackers (or cybercriminals) to gain access to the online medical data as well as healthcare personal information. This information include patient's date of birth, social security numbers, and also record of sensitive medical diagnoses, treatments and other information that is private to the patient alone. Significantly, the information includes financial data for medical payments and other account management services.

Based on the research conducted by the Identity theft resource center (ITRC) on June 2013, 45.2% of the breaches were in the medical sector. It was revealed that, over 2 million records were compromised. Business sector is the only one ahead of medical sector in terms of number of breaches identified (EMC Academic Alliance).

The effect of Medical Identity Theft to the victims can be unforgiving. Victims of MIDT suffer from emotional consequences which similarly result from any Identity theft. Emotional reaction to MIDT includes anger, fear, loss and anxiety (Betz, 2012). At times victims of Identity theft commit suicide (Sullivan B. 2004). Also, they often experience physical consequences as well which include heart shocks, hyperventilation, dizziness, sweating, high blood pressure and muscle aches, and sexual dysfunction (ITRC, 2009).

3.1 Cateories of MDT Victims

There are two types of victim for Medical Identity theft; they are primary and secondary victims. The primary victims are those that wrong health information ended in their file which lead to wrong diagnosis. As a result, it may lead to death, improper denial of insurance and billing for health services not received. The secondary victims are the Healthcare providers, Insurance Company, and organizations or agencies that the accuracy of Medical record relies on them (Katherine M. 2009).

Medical identity theft is a great concern not only because of its rapid growth rate, but rather because it is the most costly and time consuming to resolve of all types of identity theft. Additionally, medical identity theft is very difficult to detect which makes this form of scam extremely dangerous. More than 60% of MIDT victims did not discover that they were victims until long period of time after the occurrence of the event **(**Larry, Ponemon 2010**).** This resulted to misdiagnosis. If the victims are unaware that someone has been using their information to be treated for conditions that they do not have or to obtain medical prescriptions that they do not need, this can lead to severe medical problems that may lead a life threaten situation. Therefore, this high percentage of victims shows that there are significant numbers of people who are victims and are still not conscious of it. Unfortunately, by the time MIDT is discovered, the damages have been done.

43% percent of victims had to make out-of-pocket payments to resolve the crime. They paid an average of $18,660 for identity protection, credit reporting and legal counsel; medical services and medications because of delay in healthcare coverage; and refunds to healthcare providers to pay for services resulted from the fraud. Based on the survey, $12.3 billion was estimated to be the total out-of pocket cost incurred by MIDT victims in the United State of America.

An estimated 1.84 million Americans were victims of medical identity theft in 2013; this is an increase from 1.52 million individuals in last year study (Ponemon Institute 2013). Medical identity theft is expected to increase dramatically as new federal regulations defined in the Health Information

Technology for Economic and Clinical Health (HITECH) Act which provide motivations for healthcare providers to quickly transfer medical records electronically using internet **(**Larry Ponemon 2010**).**

Furthermore, the consequences of MIDT can lead the Victims to lose trust, confidence, and loyalty to their respective Healthcare providers following to the loss of their medical credentials. According to the Ponemon institute research report of 2013 on MIDT, 56% of the respondents lose trust to their respective Healthcare provider which is an increased to the 51% of the last year's study. Hence, MIDT is categories under a federal crime in UK and some countries**.** The Identity Theft and Assumption Deterrence Act of 1998 makes it a federal crime to use another person's identification in UK.

4. WAY OUT TO MEDICAL IDENTITY THEFT

Due to the devastating effects of Medical identity theft to its victims, various solutions that will curb or reduce the repercussion to minimum where developed. These include; Policies that involved Prevention, Detection and Mitigation policies; and also Technological measures.

4.1 Policies of Curbing MIDT

4.1.1 Prevention Policy

The essence of this policy is to stop medical identity theft from happening, with an emphasis on preventing its effect on patient medical records. Among the important prevention policies are: Exercising thorough caution in hiring medical personnel who have access to patient's record by the Healthcare providers, limited access should be imposed to electronic health record as well as paper health records, Patients should be educated about the right to review and request corrections to their own medical records. Clear instructions to patients on how they can get a copy of their records should also be made. Principal languages of patient population should be used in documentations available at registration counters and on respective website, and the crime of MIDT should be clearly enlightened to the patients, i.e. they should be well known that sharing their or using someone else's medical credentials is a major crime which have a lot of potential vulnerabilities (Rashidah *et al,* 2013; Kamala, 2013**).**

4.1.2 Detection policy

The Detection activities are those that assist in accurately identifying instances of medical identity theft once they have occurred and may also include determining how, where, and when the theft occurred. The essence of this policy is to provide the means of identifying the past, present and attempted MIDT. Here, both patients and health providers have a role to play. They include: determining what medical information was involved and how, when and where it was stolen and used, Denoting any problem that requires further investigation by using "Red flag" or other means of marking contradictions at different contact points with patients and medical records, training of Health employees on how to identify contradictions that need to be flagged. And also they should be trained to check for and follow up on, red flags at any contact to the patients, affected patient accounts should be placed on hold pending the outcome of the investigation, training the patients on how to thoroughly understand their explanation of benefits (EOB) documents. This will enable them to detect any error which may be a sign of MIDT, designing a process that will notify patients who have been identified as victims of medical identity theft (Booz Allen Hamilton, 2009; Kamala, 2013).

4.1.3 Mitigation policy

This policy is aimed at helping victims of MIDT in fixing the damages once the threat has been discovered. Victims can be the patients, Health providers or health personnel. Mitigation involves minimizing the risks and costs to all victims and doing everything possible to fixed back medical and financial records to the initial status before the hazard. They include: Establishing a clear written plans and procedures for handling records proved to have been corrupted by medical identity theft, if the complaints done by the patient about a billing error confirms MIDT, the first priority should be correcting the claims record to eliminate the possibility that the patient's benefits could be stopped or terminated, developing accurate abilities to receive and broadcast to healthcare participant any red-flag indicating that a medical health record has been `compromised by MIDT (Booz Allen Hamilton, 2009; Rashidah *et al.* 2013; Kamala D. Harris, 2013).

4.2 Technological Measures of Curbing MIDT

4.2.1 Smart card Technology

A Smart Card is nowadays used as a technological means of curbing the Medical identity theft. A temper-resistant chip with security software is embedded in this type of cards. The use of this card provides the safest and most secure strong Authentication of identity that is term as "two way Authentication mechanism" (Smart Card Alliance, 2010). This card enables the patients to clearly identify themselves to their individual healthcare providers whenever they want to access their records, and also, whenever the patients request healthcare services. It also provides a strong set of encryption capabilities including key generation, safe key storage, hashing and digital signature (Smart Card Alliance). Smart cards also add strong authentication capabilities that ensure only authorized healthcare personnel or providers are able to access the personal Health information. Since the card is portable, the patients can stay with it and not in the computer where third party can have access to it. Therefore, the patient can securely store their health information, thus enabling them to transfer their data securely to healthcare system.

4.2.2 Biometric technology

A biometric scanner is also use in curbing the growing problem of medical Identity theft. Biometric devices recognize people's unique physical traits—such as a fingerprint, iris, face, or voiceand use them as a means of authentication. Using these scanners, the Iris of patient's eye, Finger prints, and image of the palm veins are used to authenticate the patients in order to have right patient connected to right medical record (Kelly Santos, 2014).

4.2.3 The two-step verification

Two steps verifications, also referred to as two-factor authentication, is a step further in technological deterrent. The two-step verification adds an additional layer of protection. It requires users to have an extra credential, beyond just a password, to access an online account. Two-step verification generally requires a user to know something, such as a password, and have something, such as a specific mobile device (Joslin Woods, 2013).

Automated auditing and monitoring is also among the recent technological measures employed to curb Medical Identity theft. This is effective when the number of healthcare transaction is very large (Booz Allen Hamilton, 2009). This provides quite a lot of advantages to stakeholders in healthcare transaction who are been concerned with the detection of the threat regarding Medical Identity theft.

5. CONCLUSION

Information Technology is reaching all angles of the healthcare sector in the form of electronic health and medical records. However, information privacy and security issues continue to plague electronic healthcare system projects, especially due to the extensive use of new communication technologies like wireless network. Medical Identity Theft is the serious threat affecting the Healthcare sector nowadays. The crime is costly and on the rise according to literature survey and investigation. In addition, many victims risk their lives by having inaccuracies in their medical records as a result of someone using their medical credentials.

Individuals, Healthcare providers, Health personnel and government working together can reduce the risk of medical identity theft. Individuals need to be aware of the negative consequences of sharing their credentials. Healthcare organizations and government must improve their authentication procedures to insure intruders are not obtaining medical services and products. We have explored some of the technological and policies deterrent to MDT. It was found that applying both measures are promising to reduce the risk of MDT.

REFERENCES

AHIMA Foundation 2014. *Avoiding medical identity theft*. Accessed April 28, 2014. Source from: http://www.myphr.com/Privacy/medical_identity_theft.aspx

Betz, and A.E. 2012. The experiences of adult/child identity theft victims. *unpublished doctoral dissertation*. Iowa State University, Ames.

Booz Allen Hamilton. *Medical Identity Theft final report*. US Department of Health and Human Services. Accessed 3rd February 2014. Source from: http://www.healthit.gov/sites/default/files/medidtheftreport011509_0.pdf

Canadian Healthcare Anti-fraud Association 2004. *Canadian Health Care Fraud Survey*. Accessed 12th January 2014. Source from: http://cfevancouver.com/file/download/38.pdf

EMC Academic Alliance. *Cyber crime and the healthcare industry*. Accessed 28th April, 2014. Source from: http://www.emc.com/collateral/white-papers/h12105-cybercrime-healthcare-industry-rsa-wp.pdf

Fair Warning Inc. 2011. *Study of Canadian Health Data Breaches*. Accessed 9th February 2014. Source from: http://www.fairwarning.com/Canada/whitepapers/2011-12-WP-CANADA-PATIENT-SURVEY.pdf

HIMSS 2014. *6th Annual HIMSS security survey*. Experian Data Breach Resolution, February 19th, 2014.

ITRC 2013. *Medical Identity Theft Knowledge Survey*. Accessed 15th November 2013. Source from: http://automatedshredding.com/shredding-blog/itrc-medical-identity-theft-knowledge-survey/

ITRC 2009. *Overcoming the emotional impact*, ITRC fact sheet 108

Joslin Woods (2013). *Technology That Can Help Prevent Identity Theft*. Accessed May 2014. Source from: http://www.identityprotection.com/education/protect-yourself/technology-that-can-help-prevent-identity-theft

Katherine M. Sullivan 2009. But Doctor, I Still Have Both Feet! Remedial Problems Faced by Victims of Medical Identity Theft. *American Journal of Law & Medicine*. vol. 35: P 647-681 American Society of Law, Medicine & Ethics, Boston University School of Law.

Kamala D. Harris 2013. Recommendations for the Age of Electronic Medical Records. Accessed 12th January 2014. Source from : https://oag.ca.gov/sites/all/files/agweb/pdfs/privacy/medical_id_theft_recommend.pdf

Kelly Santos. *Biometric scanning growing to curb Identity theft*. Accessed 3rd February 2014. Source from: http://www.idt911blog.com/2014/02/biometric-scanning-growing-to-curb-identity-theft/

Larry Ponemon. 2010. *The Potential Damages and Consequences of Medical Identity Theft and Healthcare Data Breaches.* National Survey on Medical Identity Theft. Accessed 28th January 2014. Source from: http://www.experian.com/assets/data-breach/white-papers/consequences-medical-id-theft-healthcare.pdf

Nationwide Insurance 2012. *Few aware of risk of medical ID theft.* Accessed 10th February 2014. Source from: http://www.nationwide.com/about-us/061312-few-aware-of-medical-id-theft.jsp

Ponemon Institute, 2013. *Survey on Medical Identity Theft*. Accessed November 23, 2013. Source from: http://www.clearwatercompliance.com/wpcontent/uploads/2013/10/2013-Medical-Identity-Theft-Report-FINAL.pdf

PWC Health Research Institute 2011. *Medical Identity Theft Survey*. Accessed 9th February 2014. Source from http://www.amednews.com/article/20111017/business/310179959/4/

Rashidah. F. Olanrewaju, Nor'ashikin Bte. Ali, Othman Khalifa, Azizah AbdManaf. ICT in Telemedicine: Conquering Privacy and Security Issues in HealthCare Services. *Electronic Journal of Computer Science and Information Technology,* EJICST. 19-24, 4(1) 2013.

Rick Kam and Christine Arevalo 2012. *A glimpse inside the $234 billion world of Medical Fraud*. Accessed 12th September 2013. Source from: http://www.govvhealthit.com/news/glimpse-inside-234-billion-world-medical-id-theft

Sullivan B. 2004. *Behind the identity theft epidemic.* Hoboken N.J.: John Wiley & Sons.

Smart Card Allianc. *Medical Identity in Healthcare*. Accessed 3rd January 2014. Source from http://www.smartcardalliance.org/pages/publications-medical-identity-theft-in-healthcare

Updated 2013 ITRC. *Brech List Tops 600 in 2013*. Accessed April 24, 2014. Source from: http://www.idtheftcenter.org/ITRC-Surveys-Studies/2013-data-breaches.html

A Survey of Network Protocol based Steganographic Techniques

[1]Osamah Ibrahiem Abdullaziz, [2]Vik Tor Goh, [3]Huo-Chong Ling and [4]KokSheik Wong
[1,2,3]Faculty of Engineering, Multimedia University,
Cyberjaya, Malaysia,

[4]Faculty of Computer Science & Information Technology
University of Malaya, Kuala Lumpur, Malaysia
Email: [1]osamah.ibrahiem@gmail.com, [2]vtgoh@mmu.edu.my, [3]hcling@mmu.edu.my,
[4]koksheik@um.edu.my

ABSTRACT

Steganography is a sub-discipline of information hiding, which hides external information into an innocuous carrier to establish stealthy communication. Steganography in network protocols however, is an emerging research area, which exploits network protocols specifications, protocol mechanisms, network applications, and network services to realize covert channels between network end-systems. In this paper, we aim to clarify topics related to information hiding and its applications especially steganography. We also provide a brief comparison between steganography and cryptography. Eventually, we categorize and present some of the current work related steganography in networks.

Keywords: Information Hiding, Covert Channels, Network Steganography

1. INTRODUCTION

Information hiding in general is the art and science of embedding data into a digital carrier in such a way, that the distortion of the carrier is unnoticeable. The main objective of information hiding is to provide a venue, usually inessential with respect to the carrier, to accommodate external information. For the rest of the paper, the term *external information* refers to the secret data to be embedded into a carrier. Information hiding techniques can be categorized based on the nature of their applications, which includes Watermarking, Fingerprinting, Authentication and Steganography as depicted in Figure 1.

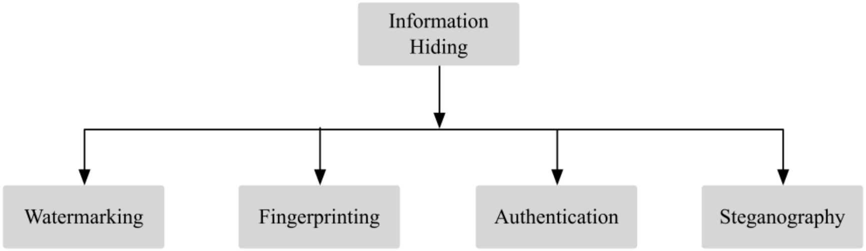


Figure 1: Information Hiding Applications

First, watermarking is the process of hiding information into a digital signal for copyright protection (Fridrich *et al.*, 2001). It is utilized to identify the ownership of the digital media. For example, companies' logos are embedded into images to protect the companies' ownership. Next, fingerprinting is the process of labeling digital media to allow the media owner to identify the customer and whether the media has been distributed to a third party. In other words, watermarking identifies the owner and fingerprinting identifies the customers. Furthermore, authentication is the process of embedding external information to verify content/user to prevent impersonation. Finally, steganography is the process of hiding information for the purpose of secret communication which is also known as covert channel (Anderson, 1996).

The main focus of this paper is on steganography, which aims to hide the very existence of the communication to avoid suspicion whereas cryptography protects communication from being decoded by unauthorized parties. Nonetheless, steganography fails if the secret communication is detected (i.e., steganography provides security through obscurity). This is unlike cryptography, which fails if the right encryption key is uncovered and utilized to decode the sensitive information. It is important to note that steganography is not intended to replace cryptography. Nevertheless, both can be combined to provide multiple layers of protection. Besides, steganography will be useful in some regions where encryption is prohibited or requires licensing.

Usually, steganography is realized on digital media such as text, image, audio and video. Another emerging area of interest in the field of information hiding is network steganography. The huge amount of data and the large number of different protocols in the Internet serves as a high bandwidth carrier for covert communication. The bandwidth of steganographic techniques in computer networks has greatly increased because of the new high-speed network technologies, and this trend is likely to continue. Even if only one bit per packet can be covertly transmitted, a large Internet site could lose 26GB of data annually due to unauthorized usage of network steganography techniques (Fisk *et al.*, 2003).

It is worth emphasizing that steganographic techniques are usually evaluated through three criteria (Johnson *et al.*, 2001) namely *bandwidth*, *robustness* and *undetectability*. The term bandwidth in the context of information hiding refers to the amount of the external information, which is transmitted with respect to the unit of time. In additional, robustness refers to the ability of resisting alteration of the external information due to noise or carrier manipulation. Finally, undetectability is the ability to resist anomaly detection. In fact, it should be noted that the three criteria are interrelated which reveals a trade-off among them. For example, the more bandwidth is utilized from the innocent carrier, the less robust and undetectable the external information technique will be. Figure 2 illustrates the criteria, which is considered to benchmark steganographic techniques.

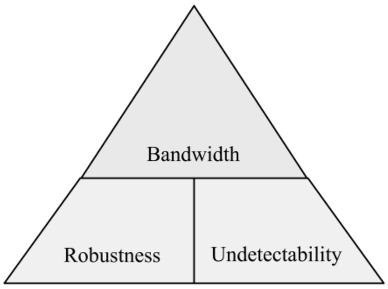


Figure 2: Steganography Criteria Triangle

2. COMMUNICATION SCENARIOS

Steganographic techniques assume the prisoners' problem (Simmons, 1983) as a communication environment. The scenario consists of two accomplices namely, Alice and Bob in a crime who have been arrested and locked up in separated cells. Their only mean of communication is by exchanging messages through an overt channel, which is monitored by a warden. The warden allows the exchange of the communication messages as long as they are presumably innocuous and do not raise suspicion. On the other hand, the prisoners will want to coordinate an escape plan through sending secret messages by utilizing a shared secret (i.e. steganographic algorithm), which establishes a covert channel. Figure 3 illustrates the prisoners' problem.

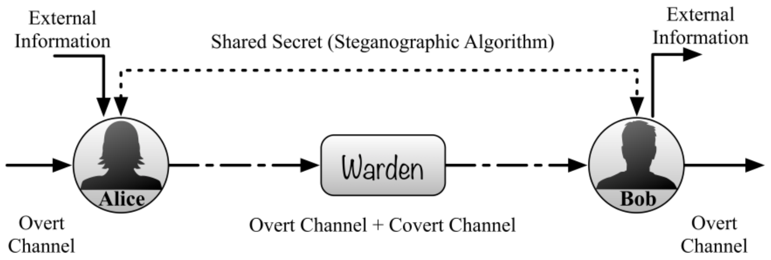


Figure 3: Prisoners' Problem

Nonetheless, there are four possible one-to-one secret communication scenarios, which are feasible in the context of steganographic techniques. In these scenarios, the packet sender may not be the generator of the overt communication traffic. On the other hand, the receiver may also not be the final destination of the traffic generated. The scenarios are detailed as follows:

1. Steganographic sender (Alice) is the overt sender and the steganographic receiver (Bob) is the overt receiver.
2. Alice is an intermediate node and Bob is the overt receiver. Overt sender (S) sends packets to Alice, which in turn, tampers the packet and forwards it to Bob.
3. Alice is the overt sender and Bob is an intermediate node. Alice generates traffic with the intended external information. The packets are then, forwarded by Bob to the overt receiver (R) with the same tampered information.
4. Both Alice and Bob are intermediate nodes. S sends packets to Alice. Alice manipulates the packets then transmits them to Bob.

Figure 4 depicts the explained scenarios in which the dotted line denotes the secret communication channel. It is not necessary that a steganographic channel can only be established in a one-to-one communication scenario. In networks that support one-to-many transmissions (i.e., multicast networks) steganographic techniques can be more deceptive since the traffic is not targeting on an individual recipient.

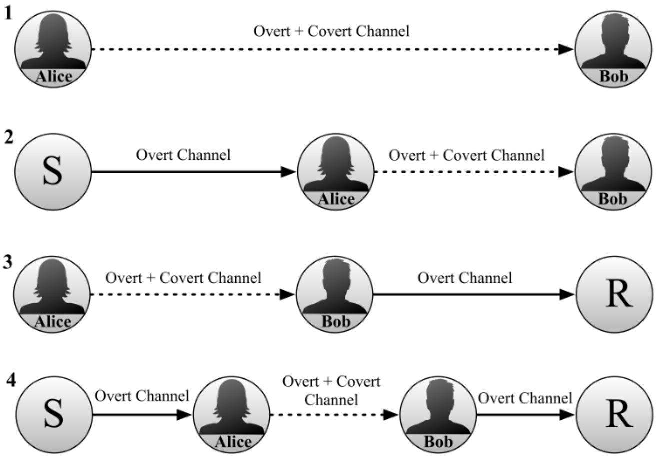


Figure 4: One-to-one Communication Scenarios

3. NETWORK STEGANOGRAPHIC TECHNIQUES

We classify network steganography techniques into three categories namely, (a) Packet Data Unit (PDU) based method, which involves encoding external information into packet header or user data field,

(b) Protocol Behavior (PB) based method, which involves encoding external information by utilizing the behavioral mechanism of network protocols, and c) Network Application and Service (NAS) based method, which involves encoding external information by utilizing the design of network applications.

4. PACKET DATA UNIT (PDU) BASED METHODS

In the case of PDU-based methods, the steganographic methods are generally less complex and of high channel bandwidth. For instance, the IP Identification (IPID) field in the IP header is utilized as the venue for external information embedding. The main purpose of IPID is to recover a complete packet from IP fragmentation process. That is, fragmentation breaks a packet into smaller fragments (i.e., with same header information), which collectively form the entire packet at the receiving end. IP fragmentation allows large packets to pass through links with smaller Maximum Transfer Unit (MTU) than the actual packet size.

In particular, (Rowland, 1997) pioneered the IPID based data hiding method where the IPID field is replaced with the numerical ASCII representation of the external information. This proposal is straightforward, easily implemented and adds no traffic overhead. However, Rowland’s method suffers from few flaws such as, no packet fragmentation handling and it is easily detected. That is, if fragmentation occurs, the external information will be duplicated. Therefore, the receiver will receive multiple copies of the same information. The method is easily detected since it results in abnormal distribution when compared to ordinary IPID distribution. In addition, the method does not provide protection against unauthorized viewing. For that, Ahsan *et al.* (2003) extended this work by improving the security of the encoded external information. In particular, their method utilizes 8 bits of the IPID field to transmit the external information and randomly generates the remaining 8 bits for identification purposes. Furthermore, the method attempted to improve the secrecy of the external information by scrambling it before the embedding process. However, it suggested probing the network for MTU prior to commencing a communication session as a solution to avoid fragmentation and thus avoiding the duplication of external information. It is important to note that IPID generation was not specified by RFC document (Postel, 1981), which makes it freely implemented by operating systems. This observation makes the abovementioned methods susceptible to detection if they fail to maintain the ordinary IPID distribution. On the other hand, Danezis (2011) designed a method based on global incremental IPID generation (i.e., commonly implemented in Windows OS). The external information is encoded by forcing an end-system to increase its IPID counter. The receiver decodes the external information by probing the amount of the counter increment. However, this method is of low channel bandwidth and did not consider packet fragmentation.

In addition, Zander *et al.* (2007) proposed a method to embed external information into the Time to Live (TTL) field. The TTL field is an 8-bit IP header field, which restricts the packet lifetime in the network to avoid routing loops. TTL value is reduced by 1 every time the packet passes through a router in the communication path. One of their proposed methods encodes ‘0’ by repeating the last TTL sent in the previous packet or ‘1’ otherwise. Besides, Shah (2011) developed a method to encode external information into a 32-bit Options field of the IP header. The main use of the option field is to record optional information such as the routing path and information for diagnostics purposes. Nonetheless, options field is not mandatory in IP header and thus not a reliable location to encode external information since network routers can filter it out. On another hand, a steganographic method is proposed in (Jankowski *et al.*, 2010) to encode external information into Ethernet frame padding. The method takes advantage of the Etherleak vulnerability (Arkin *et al.*, 2003) in which small Ethernet frames (i.e., smaller than 64 octets which is the minimum frame size) are not always padded with zeros. Some routers may fill frames with some random data from memory in LANs. Although this method provides high bandwidth for external information, it is limited to local area networks.

Furthermore, Ji *et al.* (2009) suggested the possibility to utilize the length of the packet to embed external information. First, normal packet length distribution is used as a reference to avoid detection when steganalysis of abnormal network traffic is exploited. Both communicating parties record packet lengths generated by normal communications and utilize the statistics as a reference for choosing a suitable length *L* for data embedding. Next, the external information is divided into segments and converted to decimal representation, then appended to the chosen length *L*. Finally, when the packet is received, the embedded information is extracted by subtracting *L*. Correspondingly, Nair *et al.* (2011) proposed another packet length based method, which relies on User Datagram Protocol (UDP). UDP was selected due to the random packet length pattern as compared to TCP (i.e., TCP often sends packets with maximum size), which makes it suitable to embed data by modifying packet lengths. First, the external information is divided into segments of 4 bits and converted to decimal representation. Next, a table (i.e., a matrix shared offline between communicating parties) is utilized to determine the length to be sent to convey the external information. Finally, the same table is also utilized to retrieve the external information at the receiving end.

5. PROTOCOL BEHAVIOR (PB) BASED METHODS

On the other hand, PB based method exploits the behavioral mechanisms of network protocols to encode external information. For instance, the retransmission mechanism of reliable network protocols is utilized as suggested by (Mazurczyk *et al.*, 2013). The method utilizes the retransmission mechanism of TCP Protocol (Postel, 1981) to embed external information. In this method, the sender marks a packet for retransmission so that the receiver is aware that the packet will be utilized for the external information transfer. Immediately after the receiver receives the marked packet, it does not acknowledge the successful reception of that packet. Consequently, the sender retransmit the same packet but with external information in the packet payload instead of normal user data. The limitation of this method would be the frequent utilization of retransmission, which may raise suspicion especially in networks with low error rate.

In addition, time relations of packets reception can be utilized to realize covert communication. For example, Cabuk *et al.* (2004) proposed a method in which the reception and the absence of a packet within a predetermined time interval to convey '1' and '0', respectively. Although the method is feasible, it is very difficult to predict the network conditions to keep the channel synchronized and it provides low channel bandwidth. Another class of PB based methods considers the Stream Control Transmission Protocol (SCTP) to encode external information (Fraczek *et al.*, 2012). In these methods, two features of SCTP are commonly considered, namely, multi-streaming and multi-homing. Multi-streaming is the ability to have more than one connection between sender and receiver simultaneously. Every stream in this method is assigned a bit combination depending on the number of streams created. The external information is encoded by alternating transmission through the streams depending on the external information to be embedded. Furthermore, multi-homing is the ability of an end-system to have more than one IP address depending on the number of Network Interface Cards that the end-system has. Therefore, sending a packet through an IP address encodes a secret bit (say 0) and sending through another IP address conveys another secret bit (say 1).

6. NETWORK APPLICATION AND SERVICE (NAS) BASED METHODS

Finally, steganographic methods based on NAS exploit network applications design specifications and services. In this case, applications type, implementation, protocols and network services (e.g., Voice over IP) utilized for communication are considered. For instance, Mazurczyk *et al.* (2013) described a steganographic method based on the well-known IP telephony application Skype. Skype's voice data is encrypted even when the speakers are silent. In other words, Skype does not suppress silent periods of the conversation and thus the background noise, if any, is also encrypted and transmitted through the network. Therefore, the proposed method embeds the external information into the silence signal packets. With this approach, the quality of the voice signal will not be significantly affected and the covert channel will be of high capacity since it is statistically shown that the silence signals contribute 35% to 70% of a typical VoIP call.

Furthermore, Kopiczko *et al.* (2013) put forward another steganographic method which utilizes the peer-to-peer file transfer service BitTorrent traffic to encode external information. The order in which the packets arrived at the destination is considered. In this method, the steganographic communicating end-systems operate with certain number of BitTorrent clients and thus, they can share file segments from multiple IP addresses. The steganographic sender intentionally reorders the packets from different clients to encode the intended external information. To solve synchronizing issues caused by the poor network conditions, timestamp is also altered to indicate the steganographic order of the packets reception. This method can be considered as a hybrid technique that combines the features of PB and NAS based methods.

7. CONCLUSION AND FUTURE WORK

In this paper, we presented an overview of information hiding in general and it is extension to network protocols. We also differentiate steganography from cryptography. Finally, we classified network steganographic techniques in the literature into three categories which are packet data unit, protocol behavior and network application and service. Our future work in this area will include investigating steganalysis methods, which are utilized to detect steganographic communication in network protocols. We also aim to improve some of the existing techniques, targeting mainly on the packet data unit based methods.

REFERENCES

Anderson, R. 1996. Stretching the limits of steganography. In *Information Hiding* 1174: 39-48.

Arkin, O., and Anderson, J. 2003. EtherLeak: Ethernet frame padding information leakage. [Online]. Avail: http://www.rootsecure. net/content/downloads/pdf/atstake etherleak report.pdf.

Cabuk, S., Brodley, C. E., and Shields, C. 2004. IP covert timing channels: design and detection. In *Proceedings of the 11th ACM conference on Computer and communications security*: 178-187.

Danezis, G. 2011. Covert communications despite traffic data retention. In *Security Protocols XVI*: 198-214.

Fisk, G., Fisk, M., Papadopoulos, C., and Neil, J. 2003. Eliminating steganography in Internet traffic with active wardens. In *Information Hiding*: 18-35.

Frączek, W., Mazurczyk, W., and Szczypiorski, K. 2012. Hiding information in a stream control transmission protocol. *Computer Communications, 35*(2): 159-169.

Fridrich, J., Goljan, M., and Du, R. 2001. Invertible authentication watermark for JPEG images. In *Information Technology: Coding and Computing. International Conference on*: 223-227.

Jankowski, B., Mazurczyk, W., and Szczypiorski, K. 2010. Information hiding using improper frame padding. In *Telecommunications Network Strategy and Planning Symposium:* 1-6.

Ji, L., Jiang, W., Dai, B., and Niu, X. 2009. A novel covert channel based on length of messages. *International Symposium on Information Engineering and Electronic Commerce*: 551-554.

Johnson, N. F., Duric, Z., and Jajodia, S. 2001. *Information hiding: steganography and watermarking: attacks and countermeasures* (Vol. 1).

Kopiczko, P., Mazurczyk, W., and Szczypiorski, K. 2013. StegTorrent: a Steganographic Method for the P2P File Sharing Service. In *Security and Privacy Workshops (SPW):* 151-157.

Kundur, D., and Ahsan, K. 2003. Practical Internet steganography: data hiding in IP. In *Proceedings of the Texas workshop on security of information systems*.

Mazurczyk, W., Smolarczyk, M., and Szczypiorski, K. 2013. On information hiding in retransmissions. *Telecommunication Systems*, *52*(2): 1113-1121.

Mazurczyk, W., Karaś, M., and Szczypiorski, K. 2013. SkyDe: a Skype-based Steganographic Method. *International Journal of Computers, Communications & Control*, *8*(3): 389-400

Nair, A. S., Kumar, A., Sur, A., and Nandi, S. 2011. Length based network steganography using UDP protocol. In *Communication Software and Networks, IEEE*: 726-730.

Postel, J. 1981. Internet protocol.

Postel, J. 1981. Transmission control protocol.

Rowland, C. H. 1997. Covert channels in the TCP/IP protocol suite. *First Monday*, *2*(5).

Simmons, G. J. 1984. The prisoners' problem and the subliminal channel. In *Advances in Cryptology*: 51-67.

Shah, M. K., and Patel, S. B. 2011. Network based packet watermarking using TCP/IP protocol suite. In *Engineering (NUiCONE), 2011 Nirma University International Conference on*: 1-5.

Skype, Luxembourg *Skype*. [Online]. Avail: http://www.skype.com

Zander, S., Armitage, G., and Branch, P. 2007. An empirical evaluation of IP Time To Live covert channels. In *Networks, 2007. 15th IEEE International Conference on*: 42-47.

Digital Forensics: An Overview of the Current Trends

[1*]Nordiana Rahim, [2] Ainuddin Wahid Abdul Wahab, [3] Mohd Yamani Idna Idris and [4] Laiha Mat Kiah

[1, 2, ,3, 4] Faculty of Computer Science and Information Technology
University of Malaya Kuala Lumpur
Email: [1]nordiana@siswa.um.edu.my, [2]ainuddin@um.edu.my,
[3]yamani@um.edu.my
**Corresponding author*

ABSTRACT

There are various digital forensic models occupied in digital investigative processes. Developments of suitable Digital Forensic Framework (DFF) were required for presenting the digital evidence in a better way. Digital forensic investigation process is needed to search digital devices directed for relevant evidence. This paper explores the development of digital forensics framework that is based on the foundation of Digital Forensics Workshops (DFWRs). In addition, this paper also discusses the challenges of current digital forensic framework.

Keywords: Digital Forensic, Digital Forensic Investigation, Digital Forensic Challenges

1. INTRODUCTION

Organization commonly faced many problem in conducting this process due to the lack of skills, increasing in encryption data types and the corruption of digital data while preserving it (Shields *et al.*, 2011). The increasing number of electronic crime permits a current challenges in digital forensic practitioner to measure the forensically soundness of digital evidence (Guo *et al.*, 2009). Digital Forensic field consists of several procedures and steps called framework. Digital Forensic framework was employed to handle digital cases in different environment used today (Ieong, 2006).

The rest of the paper was structured as follows. Basics of digital forensics and its development were explained in Section 2 followed by the challenges in Digital Forensics in Section 3. All these issues were discussed in Section 4 and were concluded in Section 5.

2. DIGITAL FORENSICS AND FRAMEWORK DEVELOPEMENT

Digital Forensic is a growing area of high-tech crime investigation. It is a branch of forensic science that involves a practice of collecting digital data obtained in the digital devices. It is a task of gathering, analyzing and preserving the digital data in an acceptable form and can be presented in a court of law. Digital Forensic Research Workshop, (Palmer, 2001) has defined digital forensic as ;

> *"The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and preservation of digital evidence derived from the digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal or helping to anticipate unauthorized actions shown to be disruptive to planned operations."*

There are a lot of digital forensic frameworks has been developed all over the world. Most of the organization were tend to develop their own framework and focused on technology aspect in data acquisition and data analysis of investigation (Brill *et al.*, 2006). Researchers were presenting their own framework based on their experiences and knowledge. However, there is still no specific theory in digital forensic framework

Selamat *et al.* (2008) has listed several works occupy in digital forensics framework. Their works covered from year 1995 until 2007. In this paper we extended their work with the latest frameworks proposed by Kohn et al (2013) current digital forensics frameworks as shown in Table 1.

Model Name	Author	No. of phase
The Explore, Investigate and Correlate (EIC)) conceptual framework	(Osborne *et al.*, 2010)	**3 phases** - Explore - Investigate - Correlate
Systematic Digital Forensic Investigation Model (SDFIM)	(Agarwal *et al.*, 2011)	**11 phases** - Preparation - Securing the scene - Survey and recognition - Document the scene - Communication shielding - Evidence collection - Preservation - Examination - Analysis - Presentation - Result and view
Digital Forensic Readiness Model for PKI system	(Valjarevic *et al.*, 2011)	**10 phases** - Scenario phase - Source phase - Pre-incident collection phase - Pre-incident analysis phase - Incident detection phase - Post-incident collection phase - Post-incident analysis phase - Architecture-defining phase - Implementation phase - Assessment phase
Integrated Digital Forensic process model	(Kohn *et al.*, 2013)	**6 phases** - Preparation - Incident - Incident response - Physical investigation - Digital investigation - Presentation

Table 1: List of current Digital Forensics Frameworks

In this table, Osborne *et al.* (2010) suggested a high-level conceptual framework to notify the problems (scalability and comprehension) of digital evidence. Their work was mainly about how digital evidence can be presented in a good manner. The framework consists of 3 main phases; explore, investigate and correlate. The explore phase is a starting phase for any digital forensics framework. The main goal of this phase is to provide a general overview the data of digital evidence and also enable the investigator to be more focus on the related information. In investigation phase, investigators are more focus and visualize with greater information and details links

Meanwhile, Agarwal *et al.* (2011) paper, which was inspired Digital Forensics Research Working Group (DFRW,2001) has proposed a Systematic Digital Forensic Investigation Model (SDFIM). This model is consists of 11 phase and their aims were to helps forensic practitioners and organization to setting up an appropriate policies and procedures in systematic manners.

Valjarevic *et al.* (2011) found that, there is no specific Digital Forensic Readiness (DFR) for PKI system. Therefore, the authors proposed a new model called Digital Forensic Readiness for PKI system. This model consists of 10 phases that implement several basic concepts from the previous research. The main goal of this model is to help in maximizing the potential use of digital evidence, minimizes the cost of investigation process, and minimizes the interference from any investigation and improves the current level of information system in PKI system.

Currently, Kohn *et al.* (2013) has proposed a model called Integrated Digital Forensic Process Model (IDFPM). This model was based on several digital forensic process model (DF). The author has highlighted that existing Digital Forensics Process Model (DFPM) used different terms but the fact was it referred to the same process.

3. CHALLENGES IN DIGITAL FORENSICS

Digital investigator is working with digital data for analysis and examination. Digital data can be in a diverse forms and types. According to Raghavan (2013), digital data can be in a forms of pictures, audio, documents on computer, telephone contacts, video files, email conversations, encrypted data, instant messenger conversation and network traffic patterns. Advanced in technologies had brings several challenges into this field. Table 2 summarized our finding of challenges in digital forensics:

Author	**Challenges**
(Marcella *et al.*, 2002)	• Less ability to find and prosecute the criminal • Suitable laws and legal tool used for the investigation • Appropriate reason and critical investigation result during the prosecution
(Turner, 2005)	• Specific devices for different type of data
(Mercuri, 2005)	• Scaling technologies and the need to adapt scalable architecture • Adoption of certification program in Digital Forensic • Specific format of evidence to be presented in a court
(Adelstein, 2006)	• Increasing number of live memory forensics • Lack of suitable tools for investigation process
(Kaplan, 2007)	• Increasing accessibility • Usability of strong encryption solution
(Walters *et al.*, 2007)	• Size of evidence

	• The increasing use of encryption technologies • volatile memory is temporal proximity • Live memory forensic resulting an unclear snapshot image • Malicious adversary and anti-forensics • The susceptibility of these tools to false positive and decoys.
(Garfinkel, 2010)	• The emerging size of storage devices • Increasing prevalence of embedded flash storage and proliferation hardware interface • Increasing of proliferation of OS and file formats • Increased cases needed multiple devices • Used of “cloud ” remote processing and storage • Limitation of the scope of digital forensic investigation.
(Raghavan, 2013)	• Complexity problem • Diversity problem • Consistency and correlations • Quantity or volume problem • Unified time-lining problem
(Wazid *et al.*, 2013)	• Sheer amount of data • Various of digital media types • Online disks (storage data through online) • Anomonity of IP • Anti-Digital Forensics • Testing and validation • Size of evidence

Table 2: List of challenges in Digital Forensics

Based on our finding, 3 main issues in digital forensics has been considered which were; technical and procedure issue, Digital Forensic tools issues and legal enforcement issue. Table 3 explain the description of this 3 main issue and also show several authors that listing the issues in their papers.

Main Issues	**Description**	**Related Author**
Technical and procedure	Involving the technical skill and proper procedure to acquire and analyze the crime scene	Marcella and Greenfield (2002) Adelstein (2006) Kaplan (2007) Garfinkel (2010) Raghavan (2013) Wazid *et al.* (2013)
Digital Forensics Tool	Issuing the forensic tools to perform digital analysis	Turner (2005) Walters and Petroni (2007) Garfinkel (2010)
Legal Enforcement	Involving suitable and proper format to presenting evidence	Mercuri (2005) Raghavan (2013)

Table 3: Summarization of 3 main issues from the previous authors

As mention by Adelstein (2006) information can be obtain from live system which consists of running process, network connection, memory process, and system load. This information can be captured by using live analysis technique. However, the live system is not static – files and process continuously changing. This will prevent them taken as evidence. For instance, even the log files systems was frequently changing and new mail continuously arrives, this activity would not disturb email messages sent by the suspect. The evolution of technology devices required digital investigator to obtain knowledge of digital device architecture (Marcella and Greenfield, 2002). Due to current technologies, digital evidence can be in different type and can be more specific for different type of device (Turner, 2005) .Other issues that were listed includes increasing usage of encryption application (Kaplan, 2007), growing size of storage device and the usage of cloud remote processing application (Garfinkel, 2010).

Current digital forensic operation (Computing cryptographic hashes, thumbnail generation, file carving and string searches) resulting the low performance of digital forensic tool for investigation. There are several types of digital forensics tools available in this field. Enhancement of high technology devices has impacted the functionality of the tool. The increase number of computer devices and storage size resulting the increase of time in searching the evidence. It will be quite challenging since there is no specific tool for acquiring digital evidence from high end technology.

Law enforcement faced numerous significant challenges in developing and mastering the skills, tools and techniques of digital forensics. It is not easy in finding qualified forensics personnel, either in private sector or government sector. This is due to the limitations placed to the civilian access on training programs. In law enforcement, difficulties arise due to structural and cultural factors in certain communities. Even though some cases were well trained, there are still limitations in achieving successful prosecution. It includes the lack of suitable equipment and facilities to process digital evidence and unfamiliarity of prosecutor with the issues surrounding the seizure and processing the evidence (Yasinsac *et al.*, 2001). Even with these skills, there is no guarantee that the investigator will able to find enough evidence. In the next section, we discuss on the digital forensic framework to see how the aforementioned challenges and issues can be addressed in digital forensics framework.

4. DIGITAL FORENSICS FRAMEWORK

For ease of discussion, we first discussed about the development of digital forensics process model that have been developed since 1995 by (Pollitt, 1995). It consist of 4 basics steps; acquisition, identification, evaluation and admission of evidence. Since 2001, Digital Forensic Workshops (Palmer, 2001) has discussed the foundation of Digital Forensics Process. This foundation was used as guidance by digital forensic researcher to propose their own process model. Selamat *et al.* (2008) has summarized the previous digital forensics model .In the present study, their research were then expended by incorporate with the current digital process model. The current digital forensic process models from 2010 until to 2013 have been revised. We found that the number of stages for the process might be different as compared to others. However the content or the basic processes remain the same.

From our study, one particular research work that is related to our focus is Osborne *et al.* (2010), where he proposed digital forensics process model that may help investigators to manage and present their evidence in a proper way. The analysis will be developed through a critical review of Osborne's paper by discussing the issue of surrounding scalability and comprehension of digital evidence.

This paper utilized a high level conceptual framework in order to address the issues of scalability and knowledge of the digital evidence. The purpose of Explore, Investigate and Correlate framework is to provide forensic analyst a set of structured process which assists them to present the digital evidence in a proper way.

Osborne's has discussed a standard forensics tools that usually used by the investigators for analysis process. Forensic Toolkits (FTK) and Encase Forensic (Guidance Software) are good at preserving data capture from device, indexing data, keyword searching, and recovery deleted data or information from digital media (AccessData, 2009; GuidanceSoftware, 2009). Meanwhile, .XRY tool from Micro Systemation and Universal Forensics Extraction Device (UFED) is an effective tool in extracting information such as phone numbers, call history, short message service (SMS), multimedia message service (MMS) and digital photo from mobile phone. Osborne's highlighted that the capability and effectiveness of this tools is depending on current support of the phone model.

Osborne's also listed several issues regarding digital investigation process. According to Turnbull *et al.* (2009), the increasing number of cases has increased the number of examination process and it caused the increasing volume of data required for the analysis process. The second issue is about the time factors. The evidence might be found in a large volume of data, in multiple computer networks and also in the electronic system which make the investigation more difficult. Other issues highlighted by Osborne's is the pattern behavior of the evidence where it could not easily been detected using a traditional forensic analysis (Mohay, 2005). It is because ~~of~~ the document or evidence can be distributed to more than one computer or digital devices. Therefore, current analysis tools and techniques are not able to detect or highlight the relationships exist between the similar files access in multiple device or networks easily. Furthermore, the increasing number of consumer devices has extend the challenge where each of the devices consist of different software and operating system environment, structures and file storage formats. This may reliance of the techniques in the digital evidence sources. Osborne's claimed that the standard forensic tools (FTK and Encase) are not easily understandable by unskilled investigator without extensive training in digital forensics domain.

Osborne's has proposed his framework named EIC conceptual framework. His framework aims to help the investigator with or without knowledge to and understand the digital evidence. The concept of his framework is the progressive enrichment of a constantly evolving of digital evidence. Progressive enrichment is referring to the 'top-down' approach where investigator able to add data or information during the case progresses. Constantly evolving is referring to the framework where the information is 'real-time' update. It is more efficient compared to filtering and searching result. Osborne's do cite several authors that agreed with this idea (Blanchard *et al.*, 2007; Card *et al.*, 1999; Francia *et al.*, 2006; Paquet *et al.*, 2007).

Osborne's proposed frameworks are clearly mentioned the aims of his work. He also has provided figures that explained the processes and contents of his framework. The discussion of proposed framework consistently related with the aims of the framework discussion earlier.

From our observation, this particular framework is helping investigator to be focus on some issues. It also assists the investigator to present evidence in a proper manner. This make the evidence can be understandable and it may reduce workload of a forensics investigator. However, this framework need some extra modification where it can suitable used for high-tech crime investigation that involving high-end technology

5. DISCUSSION AND CONCLUSION

Several challenges that come across by the researchers have been identified and listed in this research. There were 3 main issues and challenges; technical and procedure, digital forensic issues and legal enforcement issues. It has been discussed and suggested that knowledge and skill in managing the digital evidence were an important criteria should be owned by digital investigators. This can be implemented by put in digital forensics syllabus in university. Other than that, digital forensics organization may perhaps conduct this course for newly fresh graduate to get employed by the organization. We noticed that, advanced of digital device is one of the challenges faced by the investigator. Digital forensics company and organization should improve digital forensics tools in order to make the evidence comply and useful for law enforcement.

In this work, current digital forensics framework and their challenges have been listed and discussed. There were several related research work that need to be enhanced have been found in order to accommodate the increasing demand in digital forensic field. There were numbers of issue and challenges faced by digital investigators. Further and future works are needed. Therefore new frameworks based on the listing challenges to overcome the current issues in digital forensics fields were proposed. Overall, this paper provides knowledge to other digital investigators to notify and understand the current digital forensics framework and also challenges came across in this field.

REFERENCES

AccessData. 2009. Forensic Toolkit® (FTK®), 2014, from http://www.accessdata.com/products/digital-forensics/ftk

Adelstein, F. 2006. Live forensics: diagnosing your system without killing it first. *Communications of the ACM, 49*(2), 63-66.

Agarwal, M. A., Gupta, M. M., Gupta, M. S., and Gupta, S. 2011. Systematic digital forensic investigation model. *International Journal of Computer Science and Security (IJCSS), 5*(1), 118.

Blanchard, J., Guillet, F., and Briand, H. 2007. Interactive visual exploration of association rules with rule-focusing methodology. *Knowledge and Information Systems, 13*(1), 43-75.

Brill, A. E., Pollitt, M., and Morgan Whitcomb, C. 2006. The evolution of computer forensic best practices: an update on programs and publications. *Journal of Digital Forensic Practice, 1*(1), 3-11.

Card, S. K., Mackinlay, J. D., and Shneiderman, B. 1999. *Readings in information visualization: using vision to think*: Morgan Kaufmann.

Francia, G., Trifas, M., Brown, D., Francia, R., and Scott, C. 2006, September. Visualization and management of digital forensics data. In *Proceedings of the 3rd annual conference on Information security curriculum development* (pp. 96-101). ACM.

Garfinkel, S. L. (2010). Digital forensics research: The next 10 years. *Digital Investigation, 7*, S64-S73.

GuidanceSoftware. (2009). EnCase Forensic Retrieved 16 April 2014, 2014, from http://www.guidancesoftware.com/products/Pages/encase-forensic/overview.aspx?cmpid=nav

Guo, Y., Slay, J., and Beckett, J. (2009). Validation and verification of computer forensic software tools—Searching Function. *digital investigation, 6,* S12-S22.

Ieong, R. S. (2006). FORZA–Digital forensics investigation framework that incorporate legal issues. *Digital Investigation, 3,* 29-36.

Kaplan, B. 2007. *Ram is key extracting disk encryption keys from volatile memory.* Master's thesis, Carnegie Mellon University.

Kohn, M., Eloff, M., and Eloff, J. 2013. Integrated Digital Forensic Process Model. *Computers & Security*.

Marcella Jr, A., and Greenfield, R. S. (Eds.). 2002. *Cyber forensics: a field manual for collecting, examining, and preserving evidence of computer crimes*. CRC Press.

Mercuri, R. 2005. Challenges in forensic computing. *Communications of the ACM, 48*(12), 17-21.

Mohay, G. 2005. *Technical challenges and directions for digital forensics.* Paper presented at the Systematic Approaches to Digital Forensic Engineering, 2005. First International Workshop on.

Osborne, G., Turnbull, B., and Slay, J. 2010, February. The" Explore, Investigate and Correlate'(EIC) Conceptual Framework for Digital Forensics Information Visualisation. In *Availability, Reliability, and Security, 2010. ARES'10 International Conference on* (pp. 629-634). IEEE.

Palmer, G. 2001. A road map for digital forensics research-report from the first Digital Forensics Research Workshop (DFRWS). *Utica, New York.*

Paquet, E., and Viktor, H. L. 2007. CAPRI-Content-based Analysis of Protein Structure for Retrieval and Indexing.

Pollitt, M. 1995. *Computer forensics: An Approach to evidence in cyberspace.* Paper presented at the Proceedings of the National Information Systems Security Conference.

Raghavan, S. 2013. Digital forensic research: current state of the art. *CSI Transactions on ICT, 1*(1), 91-114.

Selamat, S. R., Yusof, R., and Sahib, S. 2008. Mapping process of digital forensic investigation framework. *International Journal of Computer Science and Network Security, 8*(10), 163-169.

Shields, C., Frieder, O., and Maloof, M. 2011. A system for the proactive, continuous, and efficient collection of digital forensic evidence. *Digital Investigation, 8*, S3-S13. doi: DOI 10.1016/j.diin.2011.05.002

Turnbull, B., Taylor, R., and Blundell, B. 2009, March. The anatomy of electronic evidence–Quantitative analysis of police e-crime data. In *Availability, Reliability and Security, 2009. ARES'09. International Conference on* (pp. 143-149). IEEE.

Turner, P. 2005. Unification of digital evidence from disparate sources (digital evidence bags). *digital investigation, 2*(3), 223-228.

Valjarevic, A., and Venter, H. S. 2011. *Towards a Digital Forensic Readiness Framework for Public Key Infrastructure Systems.* Paper presented at the Information Security South Africa (ISSA), 2011.

Walters, A., and Petroni, N. L. 2007. Volatools: Integrating Volatile Memory into the Digital Investigation Process. *Black Hat DC 2007,* 1-18.

Wazid, M., Katal, A., Goudar, R. H., and Rao, S. 2013, April. Hacktivism trends, digital forensic tools and challenges: A survey. In *Information & Communication Technologies (ICT), 2013 IEEE Conference on* (pp. 138-144). IEEE.

Yasinsac, A., and Manzano, Y. 2001. *Policies to enhance computer and network forensics.* Paper presented at the Proceedings of the 2001 IEEE workshop on information assurance and security.

AUTOGRAPH