CONFERENCE PROCEEDINGS

Proceedings of the 8th International Cryptology and Information Security Conference 2022

26th - 28th July 2022
Putrajaya, MALAYSIA

Editors:
Muhammad Rezal Kamel Ariffin
Ramlan Mahmod
Hailiza Kamarulhaili
Goi Bok Min
Heng Swee Huay
Amir Hamzah Abd Ghafar

MSCR
Malaysian Society for Cryptology Research

UPM
UNIVERSITI PUTRA MALAYSIA
BERILMU BERBAKTI

CRYPTOLOGY 2022

Proceedings of the 8th International Cryptology and Information Security Conference 2022

26th – 28th July 2022
Malaysia

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

First Print 2022

ISSN 2716-6783

WELCOMING NOTES

I am very pleased to welcome speakers from countries across the world to the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022). It is our hope that participants will grab this opportunity and gain valuable experience either through formal or informal discussion during this intellectual meeting.

Cryptography is an area of research that has tremendous impact especially in the area of communication technology. In this respect, CRYPTOLOGY2022 will provide an avenue for participants to engage on current topics related to cryptology. It is also aimed at promoting and encouraging 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 our 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 extremely important that without continuous effort to conduct research in the area one would not be able to ascertain the degree of security being deployed. Therefore it is our responsibility to ensure this biennial gathering is held in a best possible manner such that pool of excellent ideas can be brought together to solve current and future problems.

In this conference, we have 13 papers scheduled to be presented encompassing various areas of cryptology such as theoretical foundations, applications, information security and other underlying technologies in this interesting mathematical field. I hope this conference will bring Malaysia further towards realizing and translating research into a good cryptography practices.

It goes without saying that a conference of this kind could not have been held without the committed efforts of various individuals and parties. I would like to take this opportunity to congratulate and thank everyone involved for their excellent work and in particular to Universiti Putra Malaysia (UPM) and CyberSecurity Malaysia for taking up the challenge of organizing this conference. I wish CRYPTOLOGY2022 will gives all participants great experience, enjoyable and meaningful moments. With that, I once again thank all speakers, presenters and participants in making this conference possible and a successful event.

Thank you.

PROF. DR. MUHAMMAD REZAL KAMEL ARIFFIN
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 pro-

cedures 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 using 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 CRYPTOLOGY2022 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,
CRYPTOLOGY2022

Table of Contents

General Chair	Muhammad Rezal Kamel Ariffin
Local Chair	Amir Hamzah Abd Ghafar
International Program	Amir Hamzah Abd Ghafar
	Amr M. Youssef
Committee	Arif Mandangan
	Chin Ji Jian
	Chris Liaw Man Cheon
	Geong Sen Poh
	Hailiza Kamarulhaili
	Heng Swee Huay
	Kamel Ariffin Mohd Atan
	Mohd Anuar Mat Isa
	Muhammad Asyraf Asbullah
	Muhammad Reza Z'aba
	Nur Azman Abu
	Terry Lau
	Thomas Studer
	Yap Wun She

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

Keynote Paper
Digital Ringgit: A New Digital Currency with Traditional Attributes

Nur Azman Abu[*1]

[1]*Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia*

E-mail: nura@utem.edu.my
[*]*Corresponding author*

ABSTRACT

Development on central bank digital currencies (CBDCs) are ongoing across the globe. They are much needed to modernize a national network of finance. While most countries are calling for proof of concept for CBDCs, central banks have taken a wide variety of distinct approaches to CBDC design and have paced their research and development efforts independently. Without CBDCs, private digital money would become increasingly dominated by non-financial corporations. An appeal and instant recognition of paper currency has been established in the last 100 years. It is designed to have an appearance of sovereign money. Even though there is a serial number written on it, a carrier is considered an owner without any sting attached to it. There are several serious challenges in embarking on CBDC. The first technical challenge is to provide an offline apparatus on a digital money transaction. Second technical challenge is referred to as a double spending problem. Third technical challenge here is to measure and control velocity of money in a digital transaction. This project will propose and develop a digital ringgit facing those challenges. A digital ringgit Malaysia (DRM) note can uniquely assigned to a multiple session number. A blockchain on a digital ringgit is called for here. A digital currency initiative will be

of significant benefits to developing economy; hence the cashless system will be helpful in the fight against corruption and money laundering. Eventually, the digital ringgit will provide a control mechanism to a central bank such as Bank Negara Malaysia on its monetary velocity and growth of electronic money in a cashless society. A digital ringgit has an enormous potential to spur a financial cycle in Malaysia.

Keywords: digital currency; digital money; blockchain.

Introduction

Development on central bank digital currencies (CBDCs) are ongoing across the globe. They are much needed to modernize a national network of finance. In a joint report, the International Monetary Fund (IMF), the World Bank and the Bank of International Settlements (BIS) have proposed to the G20 that a cross-border network of central bank digital currencies (CBDC), underpinned by efficient technological integration and proactive international cooperation, could be of significant benefit to the world economy (Huillet, 2021). While most countries are calling for proof of concept for CBDCs, central banks have taken a wide variety of distinct approaches to CBDC design and have paced their research and development efforts independently. China's digital yuan is well ahead of the international game, and multiple countries have piloted CBDCs for cross-border use, including France, Switzerland, Singapore and Bahrain, to name just a few.

Without CBDCs, private digital money would become increasingly dominated by non-financial corporations. Jack Ma's Ant Group is a good lesson for a banking sector. Ant has quickly evolved as a consumer lending platform, with RMB 1.7 trillion (USD 262 billions) on outstanding loans as of the end of June 2020 which hold more than any Chinese bank record (Yu and Mitchell, 2021). Bahamas has become the first to launch a general purpose CBDC, known as the Sand Dollar, in October 2020. China has a number of aggressive ongoing trials (Jones, 2021). Central Bank of Russia is also planning to issue a digital ruble ((The Central Bank of the Russian Federation), 2021). This paper will propose a digital ringgit as a part of Bank Negara direction into an immediate part toward an almost cashless society. As a small country, Malaysia can be easily left behind once CBDCs has been deployed around the globe.

Human is in essence a greedy being. Large numbers of human beings in modern history have become too greedy to resist printing more money than needed for free. This year alone, Federal Reserve is creating USD120 billion dollars out of thin air every month (Clark, 2021). While money should grow at the rate of population growth, most countries would print several times more than their national growth. Gold extraction rate from the ground has always been naturally governed by population growth rate.

Paper Currency

An appeal and instant recognition of paper currency has been established in the last 100 years. It is designed to have an appearance of sovereign money. It is made of light durable material. It comes with a fixed denomination. It can be carried along and passed on with ease. Even though there is a serial number written on it, a carrier is considered an owner without any sting attached to it. It can change hands multiple times until it wears off. A unit paper money is expected to last on average 2 years. A local currency in Malaysia is called ringgit. A unit ringgit Malaysia (RM 1) has the highest denomination in Malaysia.

Bretton Woods Agreement

Bretton Woods System has created a universal international currency exchange regime that lasted from 1944 to 1971. A currency is attached to a U.S. dollar which is in turn pegged to the price of gold. A detachment from gold reserve will give any country to print more money annually. This role has been taken by a central bank which also regulate financial entities and banks. A bank practices a fractional banking system. A bank keeps a small percentage of a deposit as reserves and loans out the rest. This system boosts a money supply and supports local economic growth.

Basel III

Due to an increase in fractional banking practice, Basel III will govern all capital and liquidity buffers a bank must carry in order to contain incoming systematic risk of 2008 financial collapse. During 2008 financial crisis, a large

bank has easily gone bankrupt. A strict measure is being imposed on various fractional practical reserve to avoid a run on any financial entities. Basel III is poised to shake up gold market and derivative banking sector. These new banking rules aims at averting another global financial crisis. These new changes also come at a time of accelerated monetary inflation. Full impact from Basel III will not be felt until January 2022.

Digital Ruble

Central Bank of Russia is planning to issue a digital ruble (Pshenichnikov Wladislav, 2020). Bank of Russia plans a phased launch of a pilot project for its digital currency in October 2020 via a Public Consultation Report on the "Digital Ruble" project and published the Concept of the digital ruble in April 2021. Bank of Russia will open a wallet to a financial institution and will in turn serve an individual on a digital ruble platform.

The April concept of the digital ruble assumes its following competitive advantages over other payment methods (Skorobogatova and Zabotkin, 2021):

1. use of digital ruble cheaper for making payments compared to current non-cash ruble.
2. broader financial access for economic agents by making payments in digital rubles in the offline mode.
3. better competition in the banking sector to facilitate transfer of fund from one form of ruble to another.

In 2020, Russia adopted four laws that are essential for digitalization: on the development of biometrics, on digital financial assets, on the marketplace, and on experimental legal regimes.

Digital Yuan

The digital yuan, a central bank digital currency (CBDC), will be used there to pay some workers. This digital yuan pilot is the country's first Blockchain

Fund Payment platform involving commercial banks and payment providers. During 2022 Winter Olympics, visitors can use the digital yuan as a currency. It is also being set up for cross-border payment use.

For a daily spending of less than RMB 5,000 (USD 782), it is possible to create a digital yuan wallet with just a mobile phone number. But as the volumes increase, proof of identity and a link to a bank account are required.

China's version of a sovereign digital currency will be used to simulate everyday banking activities including payments, deposits and withdrawals from a digital wallet. The e-yuan is part of the most liquid form of money supply that includes notes and coins in public circulation, but in digital form. It is issued and backed by the country's central bank. On June 2, Beijing released a total of 200,000 digital red packets, each containing 200 yuan, through a lottery system, to local residents.

Digital Rupee

Reserve Bank of India (RBI) is working on bringing out its own CBDC. A CBDC has similar functions as a paper currency and is digitally transferable. A digital rupee is considered to be a completely new element in vast land of India. After four years of an inter-ministerial committee's recommendation to launch fiat money in digital style, the RBI has said that preliminary plans to check its feasibility may be launched soon (Pandit, 2021).

Nevertheless, Prime Minister of India, Narendra Modi launched an e-RUPI digital payment system on 2 August 2021. It is a personal and purpose-specific digital voucher. It is sent either as a QR code or SMS string to a mobile phone of a beneficiary. A user of e-RUPI Digital Payment System can redeem a digital coupon at any service providers. The service provider will get a payment immediately after being processed.

Gold Reserve

To fully understand money, a good starting point is to understand gold. Gold is the key to a financial wisdom. Its primary role is to serve as a universal money. Gold, along with silver, is a the most trusted currencies based on its

intrinsic value. It has always been money throughout human civilizations. No one questions its values, and it has always been the real money.

A gold standard is when a central bank maintains a certain ratio of paper currency to gold. Since there has not been any gold standard for more than 50 years, there is no way to determine on what the right ratio is. Nevertheless, whatever the right amount of gold is, the feeling is that if a central bank had enough of it, then it could declare a new gold standard.

There are billions of ounces of gold bullion in the world, 3 billion oz to be precise (plus another 3 billion oz in non-bullion equivalent form). There is an ounce of gold for each person in the world. There are only 2 billion oz of silver available in the market. Russia and China have a strong reliable sovereign gold reserve to back their digital money distribution as listed in Table 1.

Table 1: Ten largest gold reserves by rich countries around a globe.

Country	Gold Reserve in Tones
USA	8133.5
Germany	3358.5
Italy	2451.8
France	2436.5
Russia	2301.6
China	1948.3
Switzerland	1040.0
Japan	846.0
India	760.4
Netherland	612.5
Malaysia	38.88

Presumably, United States has the most gold but no one alive has seen it. Germany, Italy and France have managed to accumulate so much gold since World War II. China and Russia have accumulated large amount of gold recently in the last few decades. There are speculations that they both may have collected higher amount via aggressive purchases. Unfortunately, Malaysia is not in the league and ranks at 52nd place. Malaysia's official reserve asset has amounted to more than USD100 billion in the last few years and should take

this opportunity to increase its gold reserve while it can.

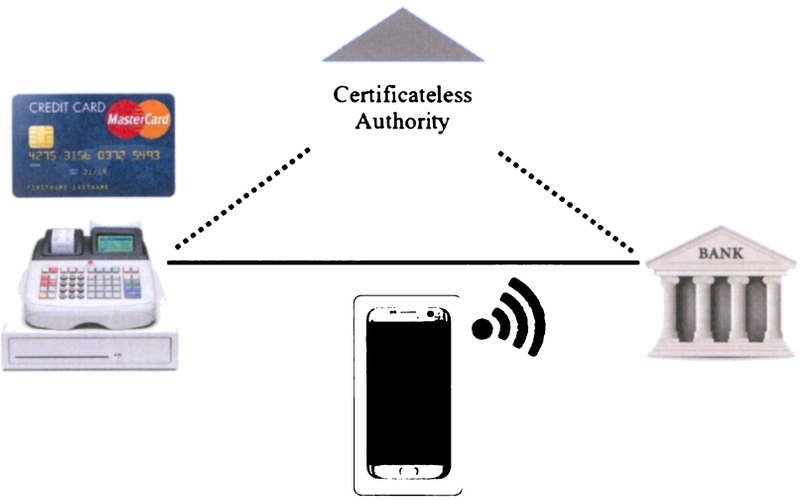


Figure 1: An E-commerce secure payment mode

Problem Statement: Current On-going Issues

Let us take specific problem on the use of debit/credit card online as shown in Figure 1. Once a debit/credit card has been used in an online transaction, it becomes vulnerable to be used or abused for another transaction due to anonymity issue (Hogan and Campbell, 2017). A technical challenge here is to provide an offline apparatus on digital money transaction.

Paper money has been around for a century. It has gained a feel and trust globally. While majority of the population is well connected, there are still a significant percentage of people who are not connected directly to a network infrastructure. They are presumably connected in almost real time. The first technical challenge here is to provide an offline apparatus on a digital money transaction. Practically, an almost online system can be presumed here.

Paper money with traditional attributes depends on its physical presence. A serial number is assigned to make it unique. A digital money can be replicated in its entirety without any loss of quality. A carrier of this paper money does not make it any less values on the money. Having a digital money, however, a

person may send a copy of his own money to ten individuals at almost the same time. A major issue here is at the receiving end. A payee cannot verify that this person did not send the same digital money to others. This second technical challenge here is referred to as a double spending. Back in 2009 on Saturday January 10, Satoshi Nakamoto prescribes the use of a peer-to-peer network as a solution to the problem of double-spending. Ideally, a receiver would need to verify an incoming digital money has not been spent. This problem can be sort out in the case of fully centralized online system.

While using digital electronic money, velocity of money tends to decrease during the same period. In order to overcome this downward spiral in its velocity, money supply (M1) has been increased to maintain an economic growth of a nation (Copic and Franke, 2020). The third technical challenge here is to measure and control velocity of money in a digital transaction. In this digital ringgit proposal, a technical mechanism will be set to allow a number of transaction on a digital ringgit as a measure to control its velocity.

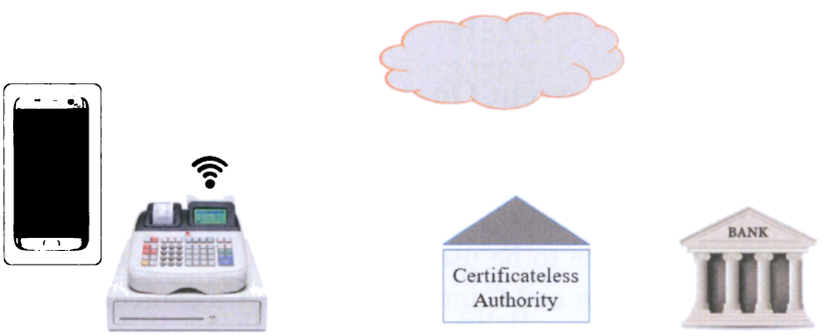


Figure 2: A basic payment system over a cloud.

Owner Privacy

In Part 10: Privacy, Satoshi discusses an idea of how banks achieve privacy for their customers, not to mention how Bitcoin might do the same. Banks simply limit access to the transactions taking place, and they are the only ones to record the identities of the participants. While Bitcoin publishes each transaction as it happens in real-time, a user on the blockchain must use a public-key infrastructure to identify themselves to the network and an associated private

key to sign the coins sent to them.

A Smartphone

A smartphone has become an important part of life. It is a source of individual networking and communication. An owner of a smartphone will protect and safeguard his/her smartphone at any cost all the time. It is more practical to digital money electronically on a smartphone.

A direct link a wallet to a subscriber number will enable more flexible management of digital currency. A smartphone will store or transfer funds and pay to another entity directly. Thanks to the massiveness of communication services, telco operators will be able to ensure a rapid launch and distribution of digital ringgit (Phillips, 2021). Mobile ID can act as an identifier to bind digital ringgit to a digital wallet to ensure individually secure entity. It is also possible to write a user 's crypto keys to a SIM card.

An Overall Goal

This project will come in 2 parts. Part One, one-time session digital ringgit with velocity one and Part Two, a multiple-session digital ringgit with velocity ten. This project shall propose a digital ringgit on a smartphone. This project will make use of a digital wallet. Traditionally, a physical wallet will carry some currency in several denominations. Each paper note can be passed to a merchant or used once only.

Current online payment systems have been vulnerable to database attack. Most of the databases are not securely encrypted; they are vulnerable to an open current and future attack such as a ransomware (Chesti and et.al, 2020).

First, one session digital ringgit payment system shall be introduced to minimize one-time payment risk. Figure 2 illustrates a basic secure payment mode over a cloud. This new model will pay special attention to a unique digital ringgit session number per paper ringgit note. Each (pseudo) random digital ringgit session number will only be used once. This session number will be dynamically changed and updated to a new number once a transaction has been executed. Therefore, it will be a randomly unique number per transaction

which is recognized by a financial service provider. Each new session digital ringgit number will also be digitally signed by the financial provider. Once it is used, it will be digitally coming back to its issuing financial provider.

Second, a multiple-session digital ringgit will be introduced into a local market. Naturally, after a user is familiar with and somewhat confident on a use of one-time session digital ringgit during the first year, he or she will be more open and ready to embark on a multiple-session digital ringgit transaction during the second year. A digital signature from an owner is hardly needed on one session digital ringgit to make a payment. However, in this multiple-session digital ringgit a digital signature from a sender or buyer is much needed to trace its validity and velocity of money. A blockchain on a digital ringgit is called for here.

Research Objectives

There are three research objectives in this project. First objective is to identify key success factors of CBDCs as a new digital money. Second objective is to develop and build a technical infrastructure on a digital ringgit. Third objective is to test and simulate a digital ringgit as a payment system as a proof of concept.

Digital Ringgit Malaysia

A digital ringgit Malaysia (DRM) note is uniquely assigned to a multiple session number. This DRM number is a 256-bit number. An owner of a digital ringgit Malaysia (DRM) is addressed as an ID. This ID is uniquely assigned to an individual person. In a specific DRM payment system, an ID is a 256-bit number. Original user identification must be unique by itself. In Malaysia, there is a MyKAD number which is recognised by a Registeration Authority (RA).

An issuing bank may use this unique user identification as an input an seed to generate an individual ID. An ownership code is stored in a blockchain which is controlled by this ID. An ID represents a person. And this person must carry a pair private-public key.

Initially, a DRM with specified amount comes from a paper currency note is validated by a digital signature from an issuing bank. This bank will issue a DRM note to an ID named Alice. This ID person Alice can spend the money by making a transaction to whom (another ID) he/she want to pay or send to Bob. A transaction consists of a current owner ID, an amount or total amount to another ID. A current owner will digitally sign this transaction. Any amount less than total amount on an original image of paper currency, a DRM must be returned back to the bank.

Later, an ID namely Bob who owns a DRM from a previous transaction, can spend this DRM as an input to a new transaction. A transaction must start from a full amount with a valid date prior to an expiry date. Any DRMs which is near an expiry date can be returned back to an issuing bank and be replaced by a new DRM. Bob can then sign and spend this DRM to another ID Carol.

A total amount of DRM in one's digital wallet is a total sum of paper note images. This visual output will give a traditional wallet looks and feels. A digital wallet in this case can still be hacked. However, an attacker needs to get full access to owner's private key in order to spend the money.

There are three research questions in this project. First, what are elements of success in developing an electronic payment system? Second, how to measure and control a velocity of electronic money? Third, what is the right technology to protect a digital money?

Elements of Success

This research project will evaluate current online payment systems according to seven element of success. Elementary membership factors consist of Claimability, Transferability, Recognition, Anonymity, Denomination, Validity Date and Velocity of Money.

Claimable Money

It measures an ability of an owner or carrier of a digital electronic money to claim it from the financial provider or bank in a case of any theft or loss. Founders of South African Bitcoin exchange, Africrypt with 69,000 bitcoins

disappeared to United Kingdom (Mason, 2021). The matter has been reported to the Financial Sector Conduct Authority (FCSA), the Hawks and the SA Reserve Bank.

Unfortunately, a formal investigation has not been launched since cryptocurrency is not considered a financial product in South Africa. This incident is the second big loss to hit South African cryptocurrency investors in two years, after Johann Steynberg, CEO of trading platform Mirror Trading International, disappearing with 23,000 bitcoins (Kilpatrick, 2021).

Transferable Money

It classifies whether an electronic money is transferable or not with ease to another person. The first kind of money transferable will be between users. The second kind of electronic money transfer will be between users without any claim from the bank. After that end user will make a payment to a merchant and the merchant will be able to claim the money from its financial provider or a bank. Non-transferable electronic money can only be used for one-time payment only.

A digital currency recipient must trust its status, he or she is not receiving a counterfeit money or a digital currency that had already been sent to someone else. In order to achieve this trust offline without requiring direct connection to a central server or a central authority, its status and complete transaction history must be publicly verifiable. Recognition of Valid Money: It measures an acceptance and recognition given toward an electronic money as a formal money by a central bank. A crypto currency especially Bitcoin is not recognized as a formal money by central banks. Any loss or theft will not be treated as monetary loss nor backed by a central bank.

Billionaire bitcoin owner Mircea Popescu has reportedly died off the coast of Costa Rica, leaving behind a cache of virtual currency valued at USD 1 billion. Word of Popescu's reported death was circulating in crypto circles, with some wondering where his holdings would go (DeCambre, 2021). Any losses can hardly be accessed or recovered via legally recognized title.

Anonymity

It measures an electronic money attachment to an owner or carrier of the money. A digital money in a financial account belongs to an account holder. Anonymity is an important element of privacy. A financial provider wants to trace to whom an electronic money belongs to at any given moment. However, a paper currency by itself is still anonymous to whom it belongs to without any reporting to the money issuer or financial provider.

Clearly, an open blockchain defies any basic understanding of anonymity since it literally publicizes the complete transaction histories of its users. A digital currency should make an attempt to provide the same anonymity similar to its traditional paper currency. A digital number assigned to it must be anonymous from any account numbers and origin of funds.

Denomination

Traditionally, paper money always comes along in certain denomination. A fixed denomination on new digital money is expected in a new electronic payment system. It is particularly crucial to introduce a fixed denomination on new electronic money in order to be successful and popular in a new electronic payment system. A fixed stable denomination in each electronic money rather than having an open amount depending on a transaction will limit any risk of loss or theft.

Validity Date

A metal coin comes with an issue year. Traditionally, money is expected to stay forever. A paper currency does not come with an expiry date even though it can be recalled at some later in the future. However, it can be recalled by a central bank at an expense of disruptive maneuver on a special occasion. In Malaysia, there was a recalled on big RM500 and RM1000 notes.

A credit card carries a validity date for periodical limit against future risk exposure. A digital money with expiry date will make sure that the currency will be recalled and coming back to a central bank on time.

A digital ringgit can trace all transactions and thus allow the central bank to measure rather than approximate its velocity. A validity date on a digital ringgit will give the central bank a better control on its expected velocity and volume of money in an economy.

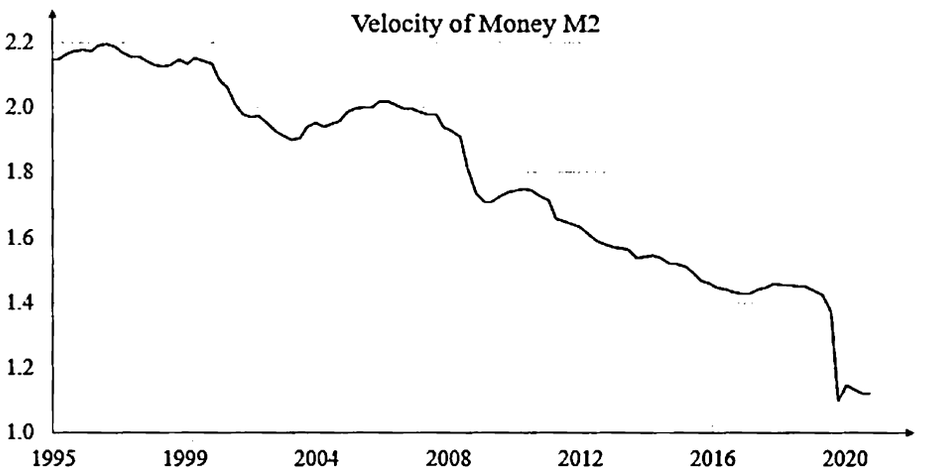


Figure 3: A declining trend on velocity of money in the past 25 years (FRED, 2021).

Velocity of Money

A velocity of money is central to the quantity theory of money (Pernice et al., 2020). Traditionally, a velocity of money denotes an average number of transactions per monetary unit within a year. In this instance, however, it measures the frequency of the use of a digital electronic money within a monetary payment system. An electronic money should support more than one-time use. A basic electronic money can only be used once only as a payment money.

In this case, number of times per unit digital money is transferred or expenses to purchase service and goods per unit note before it is claimed back to an issuing bank or financial provider. Velocity of money has been observed to slow down recently due to economic slowdown and an introduction of electronic money within a local economy. Velocity of money M2 has been observed to go a declining mode in the past quater of a century as shown in Figure 3. As a financial mean of adjustment for this decrease in monetary level, a

significant injection of money is needed to keep economic activities afloat.

An electronic money has followed on average a constant growth in the last decade. However, the velocity of paper money has gone down at the same time. An electronic money has been created by banks and non-financial corporation. They provides electroic money without any attachment to any paper money. Bank Negara Malaysia starts to lose its grasp from a moment of electronic money inception into a cashless transaction.

Stablecoin

There is an emergence of crypto currencies for online payments. A stablecoin is a specific subset of cryptocurrencies that have a value pegged to a real-world asset, such as a fiat currency like the U.S. dollar or a commodity like gold. These nongovernmental digital tokens are increasingly being used in domestic and international transactions where these digital space central banks do not have any sovereign regulations (Sigalos, 2021). Coming from a rigorous fluctuation within a cryptocurrency space, a stablecoin has emerged as a serious challenge to a central bank. At the same time, it is relatively safe from predation.

One-Session Digital Ringgit

In this research project, a new digital ringgit will be proposed. For a digital currency to be widely adopted, its users must believe its supply is strictly limited. A digital ringgit offers owner anonymity through the use of one-time unique identification and sender ambiguity by means of ring signatures.

An electronic money has followed on average a constant growth in the last decade. However, the velocity of paper money has gone down at the same time An electronic money has been created by various entities without any attachment to any paper money. Bank Negara Malaysia will start losing its grasp on the moment of electronic money in a cashless transaction.

Figure 4: A session Digital Ringgit 10 with an ECC Digital signature.

A session digital ringgit will also come along with the ECC digital signature as shown in Figure 4. This digital ringgit Malaysia(DRM) shall be honoured by the first merchant who claims its use once only. A digital signature from an issuing bank will make a DRM valid for an almost online transaction. This is an avenue to solve an off-line transaction.

This DRM will also have a validity date on it as written on bottom left corner of the DRM 10. Typically, it is valid for a month only. This monthly validity is meant to control a velocity of money respectively. A larger digital ringgit value may have shorter validity period in order to minimise its risk exposure. A user will slide a digital ringgit to a payment application during a transaction.

Once a digital ringgit is transferred to a service provider terminal, the payment system will first verify its digital signature on a digital ringgit image as depicted in Figure 4. Once verified, the payment system will request for certain amount of payment transaction from a financial provider. A threshold amount should be set on each one-time-use session digital ringgit. An encrypted update shall be prompted by the financial provider to deliver a replacement to an owner smartphone.

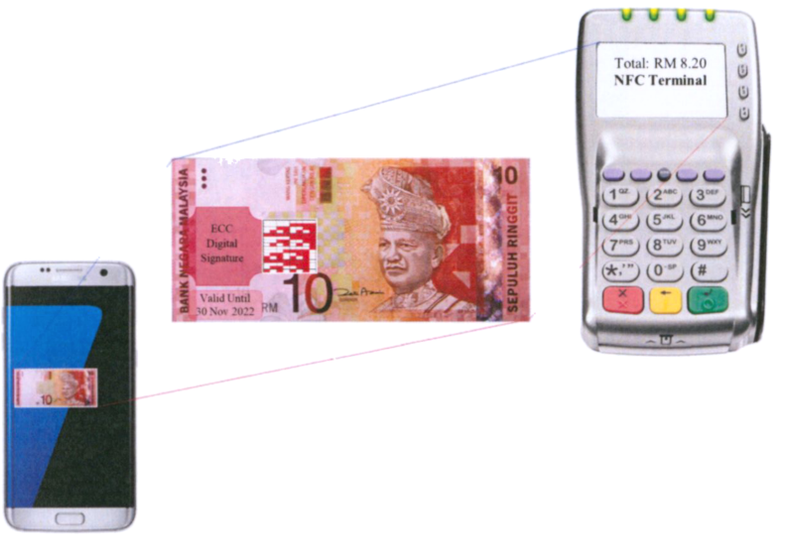


Figure 5: A user will slide an image of DRM 10 from his pocket money to an NFC cashier terminal within his smartphone Digital Ringgit application.

In Part Two, a new digital ringgit for multiple session will attach itself to real paper money note as shown in Figure 4 (Ali and et.al, 2018) A digital ringgit movement across multiple transactions can be tracked down by its issuing bank as digital ringgit will be embedded on the paper money kept by the banks as shown in Figure 5. At the same time, the velocity of the papers money can be traced from all the transactions it is going through. An electronic money has been in the current form an issued by a bank or financial provider with minimum controls from the central bank. In this digital ringgit system, the supply of money (M1) and the velocity of money will still be under control by the Bank Negara Malaysia.

A digital ringgit transaction is irreversible. Most of the time, an online purchase goes smoothly. In a case of something could go wrong, a threshold transaction can be opted to stop a payment request. An online shopper will feel safer knowing that his or her credit card company will reclaim a payment on his or her request. A digital ringgit must be able to have such an option. A third party is called for here to mediate a 2-out-of-3 threshold transantion between a buyer and a seller (Goodell and Noether, 2018).

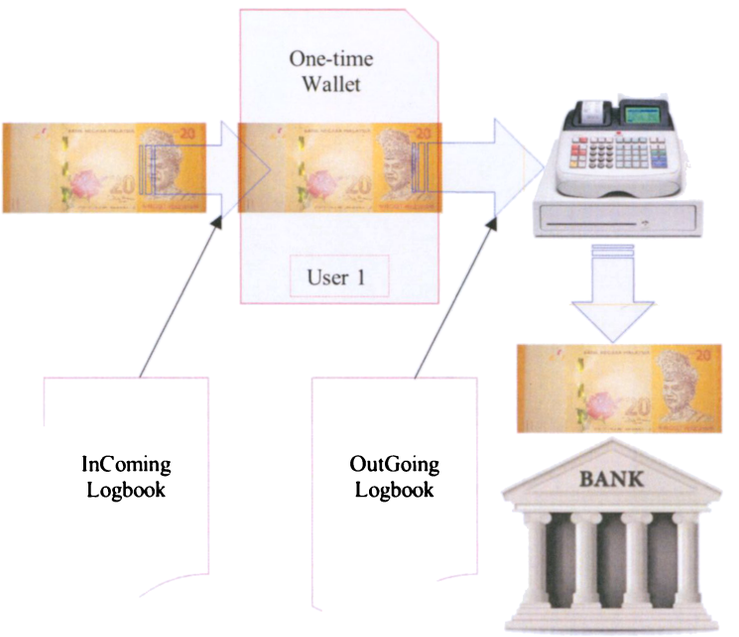


Figure 6: One-time wallet shall be proposed here to counter double spending problem.

One-time Wallet

Every entry of DRM shall be recorded in an one-time digital wallet. A DRM will be uniqely identified as its SHA-256 output. This distinct DRM shall be recorded in InComing Logbook and allowed to go into a wallet once only. Another DRM which carry the same SHA-256 shall be denied entry as depicted in Figure 6.

Similarly, a new DRM shall be recorded in OutGoing Logbook and allowed to exit once only. Another DRM which carry the same SHA-256 shall be denied an exit.An incoming DRM will carry a designated owner ID with a digital signature from a previous owner. An OutGoing DRM will be signed by the curent One-time wallet owner to a new owner. A certificateless PKI generation system is very much called for here.

Figure 7: A DRM 50 for multiple transaction with a digital signature from each transaction.

A Multiple Session DRM

In this project, a blockchain digital ringgit for multiple session will attach in self to real paper money note as shown in Figure 7 (Ali et al., 2018). A blockchain digital ringgit movement across multiple transactions can be tracked down by control authority as each blockchain DRM will be based on a real paper money kept by the banks as shown in Figure 8. At the same time, a velocity of this paper money can be traced from all the transactions. It is changing hand from one person to another but in the end will return back to a bank who issues it. A limit on the number of changing hands and expiry date will be explicitly written on it. An electronic money has been, in the current form, issued by the banks or financial provider with minimum controls from the central bank. In this digital ringgit system, the supply of money (M1) and the velocity of money will be under better control by a local central bank, the Bank Negara Malaysia.

Figure 8: A unique DRM 50 identification will also keep track of the serial number being used.

Figure 9: A DRM 20 for multiple transaction with a digital signature from each transaction with a yellow emblem.

A multiple-session digital ringgit will start from an original idea of Satoshi Nakamoto (2008) on a Bitcoin . He has started a basic idea to create a peer-to-peer digital transaction directly from one party to another without going through an intermediary or a financial institution.

A distinct part in this endeavor, a multiple-session DRM20 starts from a paper currency produced by a central bank as shown in Figure 9. It is a starting reference point as shown by a yellow arrow and an initial owner is an issuing bank as depicted in Figure 10. Initially, an issuing bank will digitally sign this DRM and assign it to the first owner. A current owner will sign his/her digital ringgit (DRM) to the next owner using his/her private key. The new owner may verify this signature from the previous owner public key.

One major issue here of course is at receiving party or a payee cannot verify that one of the owners did not double-spend the DRM. A common solution is to introduce a local trusted processing bank who checks on every transaction for double spending.

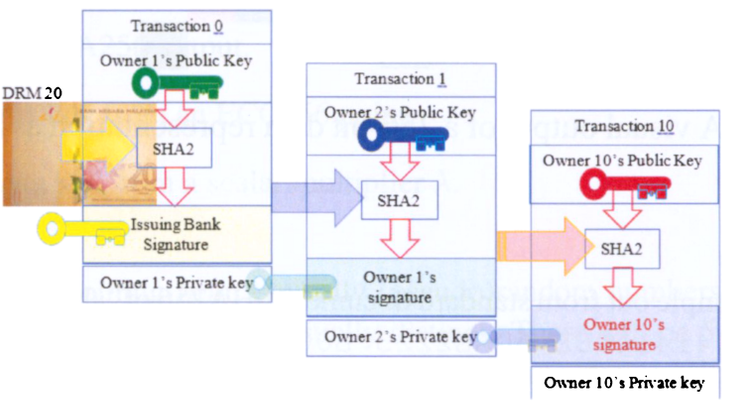


Figure 10: From original Satoshi Nakamoto's peer-to-peer- electronic cash to a digital ringgit Malaysia (DRM) design.

After each transaction, a one-session digital ringgit must be returned to an issuing bank to reissue a new digital ringgit, and only DRM issued directly from an issuing bank is trusted not to be double-spent. In A multiple-session digital ringgit, this issue is multiplying in ten folds. There is an urgent mechanism for a payee to know that the previous owners did not sign any earlier transactions prior to accepting a payment. An ideal way to confirm the absence of a transaction is to be aware of all transactions.

A 256-bit Emblem

In order to quickly visualise a 256-bit number, a perceptual digital image has been introduced here. A 256-bit number can be written in 32-byte array. Adding a zero byte in front, these 33 bytes will be assigned to red, green and blue pixel values. There will be 11 pixels and drawn in a basic shape below.

$$\begin{aligned} x &= (0x_0x_1)(x_2x_3x_4)\dots(x_{29}x_{30}x_{31}) \\ &= (0g_0b_0)(r_1g_1b_1)\dots(r_{10}g_{10}b_{10}) \\ &= a_0a_1\dots a_{10} \end{aligned}$$

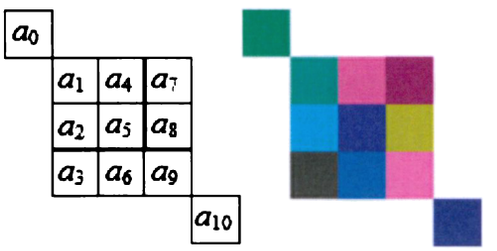


Figure 11: A visual output of a 256-bit data representation as an eleven-pixel image.

Take a sample out from standard document SHA256 of 'abc' as `BA7816BF 8F01CFEA 414140DE 5DAE2223 B00361A3 96177A9C B410FF61 F20015AD`. Then

$$x = \texttt{00BA78 16BF8F 01CFEA 414140 DE5DAE 2223B0 0361A3 96177A 9CB410 FF61F2 0015AD.}$$

An image of this hash value x is depicted on the right-hand side of Figure 11. An ECC256 point is typically referred to as a pair (x, y) where both x and y are 256-bit number. For each x coordinate, there are 2 possible values of y. An initial zero value in a_0 can be also used to indicate positive or negative algebraic value of y. Thus, this eleven-pixel image can also be used to represent an ECC256 point. Let a private key λ be the scalar multiplication from a base point $P_0(x_0, y_0)$ to a public key point $PA(x_A, y_A)$. A scalar multiplication operation is given by $PA(x_A, y_A) = \lambda \otimes P_0(x_0, y_0)$.

Figure 12: A high-fidelity visual emblem of a SHA256 of a simple string 'abc'.

There are several practical objects of use from this visual eleven-pixel image.

1. A login password
2. A basic SHA256 output.
3. A public key from an ECC256 point.
4. A private key from a scalar multiplier λ.

They are all coming from typically (pseudo)random numbers. Naturally, an eleven-pixel image will look visually distinct. There will be 5-by-5 image block. There are 14 more subblock to provide an indicator on what it is used. A basic SHA256 output will have a text SHA on top and 256 on the right most column as depicted in Figure 12. There are still seven spaces left on the bottom left column and row to put an identification on this emblem. This high-fidelity emblem will provide a compact visual representation of a 256-bit number specifically from (peudo)random elliptic points or secure hashing output.

A Login Application

A user will key in his password in an app which will produce a visual output of a SHA256 from the password. This visual image will be passed to a login system. Each pixel consists of 8-bit number per colour channel. A human visual system(HVS) is particularly sensitive to the first 4 significant bits on each pixel per colour channel. Since this visual image carry (pseudo) random 256-bit content, a simpler image which carry only 128 bits can stored as a visual future reference for the owner. An attacker still needs to guess the remaining 128 bits in order to capture the whole hashing output. A secure hashing algorithm will be strong enough to avoid any exhaustive brute force attack on the remaining 128 bits.

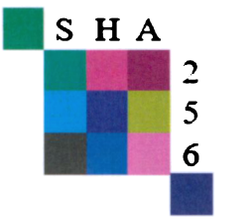


Figure 13: A partial 128-bit visual emblem of a SHA256 of a simple string 'abc'

A Partial 128-bit Emblem

A user might use several passwords during his or her lifetime. Whenever he or she forgot the password he or she could try all the password he has used before. A visual output image from a correct password will give perceptually to a partial visual image output. A partial visual image from a simple 'abc' string input is given in Figure 13. From a sample hashing value,

$$x = \texttt{00BA78 16BF8F 01CFEA 414140 DE5DAE 2223B0}$$
$$\texttt{0361A3 96177A 9CB410 FF61F2 0015AD.}$$

A partial hashing value will be used to generate a partial visual image output by discarding the 4 least significant bits on each pixel per colour channel,

$$y = \texttt{00B070 10B080 00C0E0 404040 D050A0 2020B0}$$
$$\texttt{0060A0 901070 90B010 F060F0 0010A0.}$$

Since hashing values are distinct from one another even from a pair of similar passwords, an attacker will have to resort to attacking the original password but the visual image output. Numerically, a partial hashing value will be slightly smaller. A partial visual image will look slightly darker.

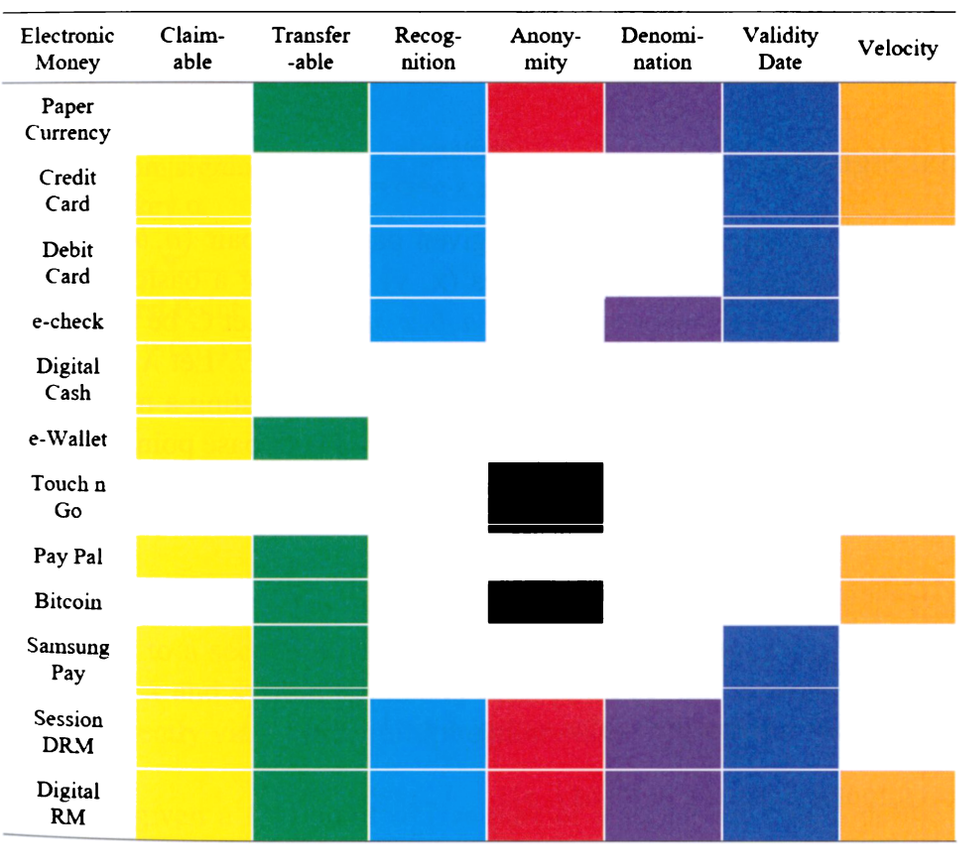
Comparison among Popular Digital Money in Malaysia

In this proposal, elements of success factors will be identified on traditional attributes in a digital payment system. A new mechanism on digital ringgit

will be proposed that can fulfill success factors in an online payment system. A brief evaluation on effectiveness of the proposed digital ringgit compared against popular digital money will be tabulated.

A dozen popular electronic money have been reviewed and compared to this proposal on a session and multiple session digital ringgits (Faeq Ali. et. al, 2018). A colour scheme as a membership criterion has been listed in Table 2.

Table 2: A Membership Evaluation of Electronic Money



A session digital ringgit satisfies all elements of success factors but velocity of money. A multiple session digital ringgit will have better chance to play a role on exercising velocity of money. A digital ringgit is expected to better represent a paper currency with an extra claimable feature. There have been

many crypto currency get lost or stolen without being able to recover.

Scalability and Performance

In an open public ledger, Bitcoin can process 7,000 transactions per second. A modern payment system needs to process about 100,000 transactions per second. At this initial design phase, it is crucial to project a growth on a blockchain ledger. In this digital ringgit, a ledger growth will be limited to the number of transactions allowed which limits the velocity of money and expiry date. Digital signing here in this project is by far the most time-consuming part. A selection on an elliptic curve will make a serious computational difference.

Digital Signature

Typically, an elliptic curve with a given parameter pair (a, b) is defined as the set of all points with coordinates (x, y) satisfying a basic Weierstraß equation, $E : y^2 = x^3 + ax + b$ where $a, b, x, y \in \mathbb{F}_q$. Let G be a base point generator, q be a prime and ϕ be an order on an elliptic E. Let λ be a private key. Then take precomputed λG as a public key. Computing a multiple λ of point G is considered as a one-way function. Given both base point G and λG, it is intractable to extract λ from them.

1. Generate random scalar α and compute αG.
2. Compute $c = \text{SHA2}(m)$.
3. Calculate a signature scalar $s \equiv \alpha + c \cdot \lambda \pmod{\phi}$.
4. Output a signature pair $(\alpha G, s)$.

An output pair $(\alpha G, s)$ is expected to be a digital signature on a message m from an owner of public key λG. It should be noted that α is a random session number in a traditional digital signature algorithm. A digital signature here consists of an EC point αG, a signature scalar s and a public key K. They can be compactly represented in 32+32+32 bytes. They will be visualized as 3 visual emblems in this project.

Signature Verification

From a signature pair $(\alpha G, s)$, public key λG and a message m.

1. Compute $c' = \text{SHA2}(m)$.
2. Compute $Q = sG$ and $Q' = \alpha G + c' \cdot \lambda G$.
3. Check on validation whether $Q = Q'$?.

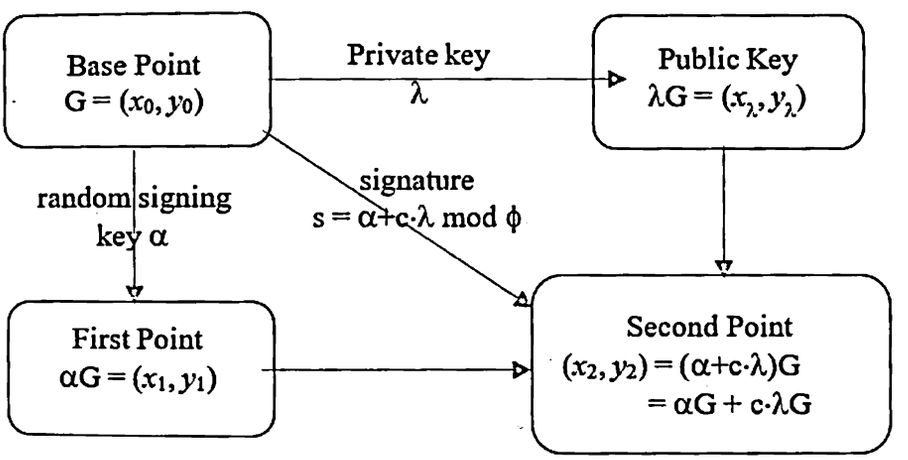


Figure 14: Point projection in a basic digital signing and verification.

Referring to Figure 14, there are 2 paths to compute and project from a base point G to a second point $(\alpha + c \cdot \lambda)G$. First, given a signature scalar $s = \alpha + c \cdot \lambda$ and system parameter base point G, the second point can be computed directly via a point multiplication sG.

Second, given a first point αG as part of a signature, take a public key λG and message m, then a scalar c can be independently computed as $c' = \text{SHA2}(m)$. Next, $c' \cdot \lambda G$ will be projected from a public key λG via a point multiplication. Thus, αG and $c' \cdot \lambda G$ will be added together to form $\alpha G + c' \cdot \lambda G = (\alpha + c' \cdot \lambda)G$.

In a case of both first and second paths will give the same answer, then the pair $(\alpha G, s)$ is considered a valid signature on a message m from an owner of

public key λG who must have used a private key λ in computing $s = \alpha + c \cdot \lambda$ to digitally sign it.

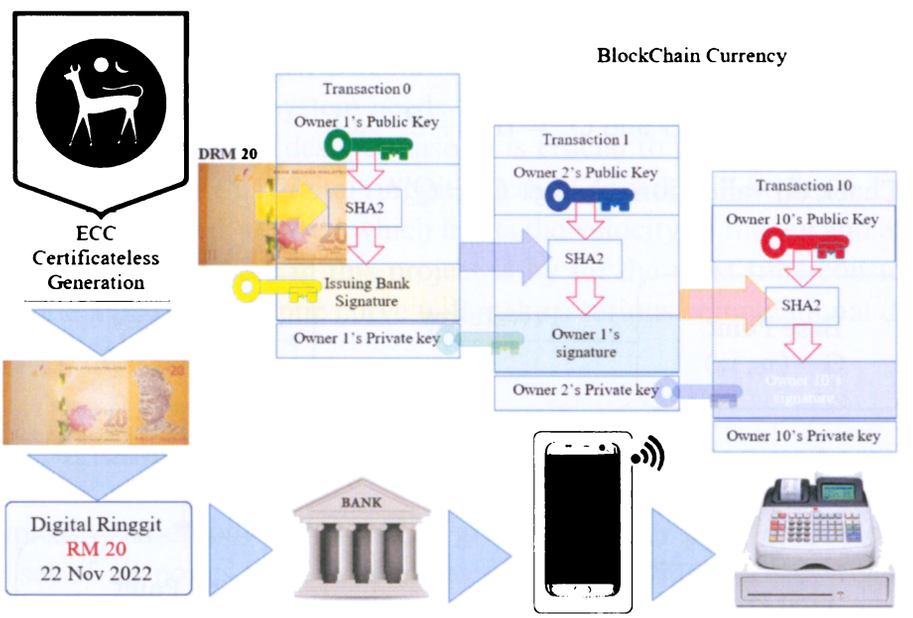


Figure 13 An overall insfrastructure of DRM from a paper currency to a merchant.

Figure 15: An overall insfrastructure of DRM from a paper currency to a merchant.

Impact Statement

A CBDCs is an idea whose time has come. If properly designed, they present an opportunity to improve payments with a technologically advanced representation of central bank money (Auer et al., 2021). Further exploration on CBDC design choices and their macrofinancial implications is essential.

The Federal Reserve has released a long-awaited review of the potential for creating a CBDC for US Dollar on 20 Jan 2022. An introduction of a CBDC would represent a highly significant innovation in US Dollars. The Fed report notes that the creation of a CBDC will seek to complement current financial systems, not replace them. Any digital currency the central bank issued would have to protect consumer privacy while supporting faster and

cheaper payments (Choo, 2022).

Money plays three main roles in a civilisation:

1. as a unit of account, the yardstick of economic activity;
2. a means of exchange to make payments; and
3. as a store of value to transfer purchasing power over time.

With a digital ringgit, BNM main goal is to provide a universal means of exchange for the digital economy. A digital ringgit will be crucial for a correct design of CBDCs as a new form of money of Malaysia in embarking into the digital era as illustrated in Figure 15. A digital ringgit should be introduced to public gradually.

First, one session digital ringgit payment system shall be introduced to minimize one-time payment risk. Each random digital ringgit session number will only be used once. This session number will be dynamically changed and updated to a new number once a transaction has been executed. Therefore, it will be a randomly unique number per transaction which is recognized by a financial service provider.

Each new session digital ringgit number will also be digitally signed by the financial provider. Once it is used, it will be digitally coming back to its issuing financial provider. Bank Negara will save the physical money printing since the future notes will be kept by the banks. Only digital ringgits will be floating around. This new era will save paper printing in the future.

Second, a blockchain digital ringgit for multiple session will attach in self to real paper money note. A blockchain digital ringgit movement across multiple transactions can be tracked down by control authority as each blockchain DRM will be based on a real paper money kept by the banks. At the same time, a velocity of this paper money can be traced from various transactions. It is changing hand from one person to another but in the end will return back to a bank who issues it.

During a post covid era, Malaysian society will learn how to interact in a cashless and contactless transactions. A digital ringgit will spur the growth of national economy. An industrial and financial will play an intermediary roles rather losing control from crypto currencies.

Ideally, a digital ringgit should be best serve as an electronic money via traditional attributes with privacy-protection, bank-mediation, transferability and also identity-verification. These attributes are taken from the latest publication on ((Federal Reserve Board), 2022) from Board of Governors of The Federal Reserve System.

1. **Privacy-Protection.** Even though Malaysia does not have a privacy act, protecting consumer privacy is critical. A digital ringgit would need to strike an appropriate balance between safeguarding the privacy rights of consumers and affording the transparency necessary to deter criminal activity.

2. **Bank Intermediation.** A digital ringgit should hardly add to a significant expansion of BNM role in the financial system and the economy. Under a bank intermediation, the private sector would offer accounts or digital wallets to facilitate the management of a digital ringgit payment system. These already in place potential intermediaries will operate in an open market for digital ringgit services.

3. **Transferable.** For a digital ringgit to serve as a widely accessible means of payment, it would need to be readily transferable between customers of different intermediaries. The ability to transfer value seamlessly between different intermediaries makes the payment system more efficient by allowing money to move freely throughout the economy.

4. **Identity-Verification.** Financial institutions in Malaysia are subject to robust rules that are designed to combat money laundering and the financing of terrorism. A digital ringgit would need to be designed to comply with these rules. In practice, an issuing bank needs to verify an identity of a person accessing a digital ringgit for a large sum of money, just as banks and other financial institutions currently verify the identities of their customers.

Latest Update

Central bankers are seizing on recent turmoil in cryptocurrency markets to push aggressively for central bank digital currencies (CBDCs). They made their case during the World Economic Forum 2022 in Davos. Someone must be responsible for the value, and it must be accepted universally as a means of exchange. Digital currencies issued by central banks, recognized officially by governments, and circulated into the economy in partnership with large commercial banks would supposedly represent safe, secure, and stable digital money.

A privately issued crypto token such as the Luna token, which was touted as being pegged to the U.S. dollar via TerraUSD, turned out to be pegged to zero once it crashed by over 99% in May (Gleason, 2022). The collapse of the Terra ecosystem — a much-hyped experiment in decentralized finance — began with its algorithmic stablecoin losing its peg to the US dollar, and ended with a bank run that made $40 billion of tokens virtually worthless (Nicolle and Kharif, 2022).

After crypto's last two-year hibernation ended in 2020, the sector spiked to around $3 trillion in total assets last November, before plunging to less than $1 trillion. Now back around $1 trillion, the crypto market is only marginally above the approximately $830 billion mark it reached in early 2018. As Bitcoin slipped almost 70% from its record high, a panoply of altcoins also plummeted. These unfortunate events have proven that digital money system must be properly taken care by a soverigne authority but individuals or small corporation.

Uses and Functions of a digital ringgit

A digital ringgit transaction would need to be final and completed in real time, allowing users to make payments to one another using a risk-free asset. Individuals, businesses, and governments could potentially use a CBDC to make basic purchases of goods and services or pay bills, and governments could use a digital ringgit to collect taxes or make benefit payments directly to citizens.

Additionally, a digital ringgit could potentially be programmed to, for example, deliver payments at certain times. A digital ringgit could potentially serve as a new foundation for the payment system and a bridge between various payment services, both legacy and new. It could also maintain the centrality of safe and trusted paper ringgit in a rapidly digitizing economy. An innovative effort should be made to graphically present touch-and-feel of paper ringgit digitally.

At a global stage, a joint international effort(IMF, Word bank and BIS) focuses on broadening the horizon beyond central banks' individual studies of CBDCs for domestic economy. An efficient technological integration and proactive international cooperation, could offer cheaper, faster and cleaner cross border payment and thus provide significant benefit to the world economy.

Conclusion

A cashless economy describes an economic state whereby financial transactions are not conducted with money in the form of physical banknotes or coins, but rather through the transfer of digital information between the transaction parties. the paper discovered that the adoption of the cashless economy policy can enhance the growth of financial stability in the country. It appears that much has already been done in making the people aware of the cashless economy and that a sizeable proportion of the people are actually awaiting the introduction of the cashless economy.

A digital currency initiative will be of significant benefits to developing economy; hence the cashless system will be helpful in the fight against corruption and money laundering. One most significant contribution of the cashless economy is that it is expected to reduce the risk associated with carrying cash. Since most transactions will now be settled electronically, people will have less need to move around with cash and therefore, loss of cash, theft and armed robbery will drastically reduce.

In this project, a digital ringgit for multiple session has been proposed which is directly pegged to a paper currency. Having the original paper money in vault, any dispute on the source of a digital ringgit can be checked. In-

directly, the digital ringgit will provide a control mechanism to Bank Negara Malaysia on its monetary velocity and growth of electronic money in a cashless society.

REFERENCES

Ali, M. F., Abu, N. A., and Harum, N. (2018). A Novel Multiple Session Payment System. *International Journal of Advanced Computer Science and Applications*, 9(6).

Auer, R., Frost, J., Gambacorta, L., Monnet, C., Rice, T., and Shin, H. S. (2021). Central bank digital currencies: motives, economic implications and the research frontier. *Annual Review of Economics, forthcoming*.

Choo, L. (2022). The Fed may create a US digital currency and wants your input. `https://www.protocol.com/bulletins/federal-reserve-cbdc`.

Clark, J. (2021). The Fiat Free-For-All: Currency Creation vs. Gold and Silver Production. `https://goldsilver.com/blog/the-fiat-free-for-all-currency-creation-vs-gold-and-silver-production/`. Online at GoldSilver.com.

DeCambre, M. (2021). One of the largest owners of bitcoin, who reportedly held as much as $1 billion, is dead at 41. `https://www.marketwatch.com/story/one-of-the-largest-owners-of-bitcoin-who-reportedly-held-as-much-as-1-billion-is-dead-at-41-reports-11624904721`. Online at Crypto Reports, Market Watch, 28th June 2021.

(Federal Reserve Board), F. (2022). Money and Payments: The U.S. Dollar in the Age of Digital Transformation, Research and Analysis. `https://www.federalreserve.gov/publications/files/money-and-payments-20220120.pdf`. Online at Federal Reserve website, January 2022.

FRED, F. R. B. o. S. L. (2021). Velocity of M2 Money Stock [M2V]. https://fred.stlouisfed.org/series/M2V. Online at FRED, Federal Reserve Bank of St. Louis, 12nd August 2021.

Gleason, S. (2022). Globalists Convene to Plan Central Bank Digital Currencies. https://goldseek.com/article/globalists-convene-plan-central-bank-digital-currencies. Online at Money Metals, 25th May 2022.

Goodell, B. and Noether, S. (2018). Thring signatures and their applications to spender-ambiguous digital currencies. *Cryptology ePrint Archive*.

Huillet, M. (2021). World Bank and BIS champion central bank digital currencies at G20. https://cointelegraph.com/news/imf-world-bank-and-bis-champion-central-bank-digital-currencies-at-g20. Online at Coin Telegraph: The Future of Money, 9th July 2021.

Jones, M. (2021). Central Bank Digital Currencies Get Full BIS Backing. https://www.reuters.com/business/central-bank-digital-currencies-get-full-bis-backing-2021-06-23/. Online at Business Reuters (London), 23rd June 2021.

Kilpatrick, M. (2021). Brothers who vanished with US$3.6 billion of investors' bitcoin can't be found. https://www.newshub.co.nz/home/technology/2021/06/brothers-who-vanished-with-us-3-6-billion-of-investors-bitcoin-can-t-be-found.html. Online at Newshub, 24th June 2021.

Mason, E. (2021). South African Brothers Disappear, Along with $2.2 Billion Worth of Bitcoin, Crypto and Blockchain. https://www.forbes.com/sites/emilymason/2021/06/23/south-african-brothers-disappear-along-with-22-billion-worth-of-bitcoin/. Online at Forbes, 23rd June 2021.

Nakamoto, S. (2008). Bitcoin: A peer-to-Peer Electronic Cash. https://bitcoin.org/bitcoin.pdf..

Nicolle, E. and Kharif, O. (2022). A $2 Trillion Free-Fall Rattles Crypto to the Core. https://www.bloomberg.com/news/articles/

2022-06-26/crypto-winter-why-this-bitcoin-bear-market-is-different-from-the-past? Online at Bloomberg, Markets-The Big Take, 27th June 2022.

Pandit, S. (2021). Coming Soon: The Digital Rupee. https://www.indialegallive.com/cover-story-articles/il-feature-news/coming-soon-the-digital-rupee/. Online at India Legal, 30th July 2021.

Pernice, I. G. A., Gentzen, G., and Elendner, H. (2020). Cryptocurrencies and the Velocity of Money. In *Cryptoeconomic Systems Conference.*

Phillips, T. (2021). Russian Carriers Seek Digital Ruble Mobile Wallet Role. https://www.nfcw.com/2021/02/22/370793/russian-carriers-seek-digital-ruble-mobile-wallet-role/. Online at NFC World : What's new in Payments, 22nd February 2021.

Pshenichnikov Wladislav, W. (2020). Prospects of issuing digital ruble and its functioning in the country's payment turnover. *π-Economy*, 86(6):101–109.

Sigalos, M. (2021). Why the Fed hates cryptocurrencies and especially stablecoins. https://www.cnbc.com/2021/07/16/jerome-powell-promotes-cbdc-digital-dollar-warns-against-stablecoins.html. Online at CRYPTO DECODED, 16th July 2021.

Skorobogatova, O. and Zabotkin, A. (2021). Presentation on the Digital Ruble Concept. https://www.cbr.ru/eng/press/event/?id=9739. Online at Bank of Russia, 8th April 2021.

(The Central Bank of the Russian Federation), C. (2021). Bank of Russia's Work: Results in Brief | The Bank of Russia's Annual Report for 2020. https://www.cbr.ru/Collection/Collection/File/35399/rb_2020.pdf. Online at Bank of Russia website.

Yu, S. and Mitchell, T. (2021). China's Central Bank Fights Jack Ma's Ant Group over Control of Data. https://www.ft.com/content/1dbc6256-c8cd-48c1-9a0f-bb83a578a42e. Financial Times, 23rd April 2021.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

Trapdoor Function from Weaker Assumption in the Standard Model for Decentralized Network

Anushree Belel*[1], Ratna Dutta[1], and Sourav Mukhopadhyay[1]

[1]*Department of Mathematics, Indian Institute of Technology Kharagpur, Kharagpur, India - 721302*

E-mail: anubelel@gmail.com
**Corresponding author*

ABSTRACT

Decentralized cryptography has emerged as an interesting area in modern cryptography to avoid a single point of failure with applications in cloud computing, electronic voting, the Internet of Things (IoT), ad-hoc networks, and many more. *Trapdoor function* (TDF) is an important building block in cryptography. *Threshold trapdoor function* (TTDF) supports sharing of master trapdoor key among multiple servers in such a way that only a threshold number of servers can invert the evaluated value of a randomly chosen function from the collection of TTDF. There exist realizations of TTDF from assumptions such as the *decisional Diffie-Hellman* (DDH) and the *learning with errors* (LWE) which are strong as compared to the *computational Diffie-Hellman* (CDH) assumption. Motivated by the surging importance of TTDF, we couple *Shamir's threshold secret sharing* with the CDH based *recyclable one-way function with encryption* (OWFE) of Garg and Hajiabadi (2018) and construct the *first* TTDF under the hardness of the CDH problem. Our scheme leads to shorter image size as compared to the existing DDH-based TTDF scheme of Tu et al. (2019) and features *one-wayness* against a selective adversary in the standard security model without using random oracle heuristic. More interestingly, when contrasted with previous TTDF, our candidate exhibits favourable results in terms of communication bandwidth.

Anushree Belel, Ratna Dutta & Sourav Mukhopadhyay

Keywords: Decentralized Network, CDH, OWFE, Selective security, Threshold secret sharing

1 INTRODUCTION

The emergence of ubiquitous computing has led to multiple heterogeneous devices with increased connectivity and has formed the Internet of Things (IoT). These IoT devices are often constrained regarding resources like memory, bandwidth, and computational power, thus requiring the assistance of more powerful but often untrusted servers in order to store, process, and perform computations on the collected data leading to cloud-assisted computing. A key challenge in this cloud-assisted computing paradigm is how to protect the security and privacy of participants considering the clients' resource constraints, especially in the multi-client setting. Although the classical cloud-computing paradigm traditionally involves one client, we argue that a multi-client environment is more realistic since often an aggregator has to perform computations from data collected from multiple users. For instance, consider the case when it is required to compute statistics for data collected from multiple users to monitor electricity consumption via smart metering, clinical data, or even the safety of buildings or environmental conditions from data collected through numerous sensors. Another example is network authentication protocols widely used to provide a single-sign-on experience to users by enabling them to authenticate periodically (e.g., once a day) to a ticket-granting service using their credentials and obtain a ticket-granting ticket (TGT). They can use TGT to get access to various services such as mail, printers, internal web, and so on. The recommended approach for generating a TGT is authenticated encryption using a master secret key to provide both confidentiality and integrity for the information contained in the ticket. This renders the master secret key an important attack target as it remains unprotected in memory over a long period. The proliferation of a wide range of IoT has provided users with new and convenient ways to interact with the world around them. Many IoT devices are employed to store secrets required to authenticate users or enable secure payments. Such devices are not equipped with proper environments to store secret keys and even when they are, provide developers with little programmability for their applications. It is therefore desirable to leverage the fact that users

own multiple devices such as smartphones, smart watches, smart TV, etc. to distribute the key material among them instead of keeping it entirely on any single device. This enables multi-device cryptographic functionalities without making strong assumptions about a device's security features. Given limited computation and communication power of IoT devices, such distributed primitives should require minimal interaction and limited cryptographic capabilities. Decentralized cryptosystems have rapidly gathered momentum. In many real-life situations, we do not believe that any individual can be trusted, though it is reasonable to assume that the majority of people are trustworthy. Similar is the case for online transaction. We may doubt to trust only one single server, but we hope that the majority of servers are working properly. An example of an application whose security could be improved greatly with a decentralized mechanism is a network certification authority, a trusted entity that certifies that a given public key corresponds to a given user. If we trust one server to perform this functionality, then it is possible that as a result of single server failure, no certificate can any longer be trusted. Thus, it is a good idea to distribute the functionality of the certification authority between many servers. *Threshold cryptography* [Desmedt and Frankel (1989), Frankel (1989), De Santis et al. (1994)] enhances the security by distributing some cryptographic functionality among several users in such a manner that at least t users can jointly compute the functionality but less than t users cannot break the security of the functionality. Over the past few decades of technological development, there has been a surge of new cryptographic results in threshold setting, including threshold ring signature, threshold authentication, threshold authenticated encryption, distributed pseudo-random function, functional secret sharing, homomorphic secret sharing, and many more.

The *trapdoor function* (TDF) has been a centerpiece of all these primitives. TDF [Diffie and Hellman (1976), Rivest et al. (1978)] consists of a collection of functions where each function is easy to compute if access to the function's index key is given and easy to invert if access to related trapdoor key is provided. Peikert and Waters (2011) provided instantiation of *lossy trapdoor function* (LTDF) under the *decisional Diffie-Hellman* (DDH) and the *learning with errors* (LWE) assumption and utilized it in designing TDF, CCA secure cryptosystem, oblivious transfer and collision-resistant hash function in a black-box manner. Freeman et al. (2010) presented LTDF based on the *d-linear* assumption by simplifying the DDH-based construction of Peikert and

Waters (2011). They have also designed LTDF based on the *quadratic residuosity* and the *composite residuosity* assumption. Boyen and Waters (2010) constructed LTDF with improved index key size under the *decisional bilinear Diffie-Hellman* (DBDH) assumption. However, building TDF based on the CDH problem was an open problem for more than three decades until the work of Garg and Hajiabadi (2018) that resolved this problem. They introduced the notion of *recyclable one-way function with encryption* (OWFE), realized it under the CDH assumption and used it to provide a black-box construction of TDF.

Threshold trapdoor function (TTDF) (Tu et al., 2019) makes TDF suitable in distributed setting. Each function in the family of TTDF is easy to compute if the function's index key is given and easy to invert if at least t users participate in the protocol. Tu et al. (2019) proposed two constructions of *threshold lossy trapdoor function* (TLTDF) by thresholdizing the DDH-based and the LWE-based LTDF of Peikert and Waters (2011). Tu et al. (2019) also came up with generic constructions of threshold and revocation encryption utilizing TTDF.

Our contribution. The goal of this paper is to address the problem of designing TTDF for secure decentralized protocols that achieve one-wayness under *weaker* underlying assumption, thereby providing stronger security guarantees. The existing construction of Tu et al. (2019) achieves one-wayness under the DDH and the LWE assumption. To the best of our knowledge, there is no construction of TTDF based on the CDH assumption which is weaker than the DDH assumption. This somewhat unsatisfactory state-of-affairs motivates our search for CDH-based TTDF. Inspired by the work of Garg and Hajiabadi (2018) that utilizes CDH-based *recyclable* OWFE to build a TDF, we share the master trapdoor key by (N, t)-threshold *Shamir secret sharing* (Shamir, 1979) and provide a shared trapdoor key to each identity holder, enabling each of them to compute an inversion share of the image of a domain element. A combiner can recover the pre-image if it has at least t inversion shares. We briefly summarize below the comparison of our scheme with the existing DDH-based TTDF of Tu et al. (2019). We exclude comparing our work with the LWE-based TTDF of Tu et al. (2019) which does not use any group structure and is constructed using matrices and introducing errors from some specific distribution (generally Gaussian distribution) in a controlled way.

Table 1: Comparative summaries of storage, communication bandwidth and security of TTDF schemes.

Scheme	Storage			Communication	Security	
	|ik|	|mtk|	|tk|	$\|Y\|$	HA	SM
(Tu et al., 2019)	$(l_1{}^2 + l_1)\|\mathbb{G}_1\|$	$l_1 t\|\mathbb{Z}_q\|$	$l_1\|\mathbb{Z}_q\|$	$(l_1+1)\|\mathbb{G}_1\|$	DDH	Adaptive
Ours	$(2n - 2nr + 1 + 4n^2r)\|\mathbb{G}\|$	$2ntr\|\mathbb{Z}_p\|$	$2nr\|\mathbb{Z}_p\|$	$\|\mathbb{G}\| + 2nr$	CDH	Selective

|ik| = size of index key, |mtk| = size of master trapdoor key, |tk| = size of shared trapdoor key, $|Y|$ = size of image, HA = Hardness Assumption, SM = Security Model, $|\mathbb{G}|$ = bit size of an element of the CDH-hard group $\mathbb{G}$,$|\mathbb{G}_1|$ = bit size of an element of the DDH-hard group $\mathbb{G}_1$, p = cardinality of the CDH-hard group $\mathbb{G}$, q = cardinality of the DDH-hard group $\mathbb{G}_1$, $|\mathbb{Z}_p|$ = bit size of an element of $\mathbb{Z}_p$ where p is a prime, $|\mathbb{Z}_q|$ = bit size of an element of $\mathbb{Z}_q$ where q is a prime, l_1 is domain size of the DDH-based TTDF Tu et al. (2019), n = domain size of recyclable OWFE, r = length of a binary string, $l = (n + nr)$ is domain size of our TTDF, t = threshold value

- Similar to the DDH-based scheme of Tu et al. (2019), we use *Shamir's threshold secret sharing* to share the master secret key. However, Tu et al. (2019) uses LTDF of Peikert and Waters (2011) and the order of the underlying group is set large to make DDH problem hard. On the other hand, we make use of *recyclable* OWFE of Garg and Hajiabadi (2018) and the order of the underlying group is set large to make CDH problem hard.

- The most appealing feature of our construction is that the underlying hardness assumption in our scheme is weaker than the DDH assumption. We support the conjectured security against a selective adversary by analysis and prove its one-wayness under the CDH assumption. More precisely, we obtain the following result.

  Theorem 1.1. *(Informal) Assuming perfect privacy of Shamir's threshold secret sharing scheme and security of recyclable* OWFE *under the hardness of the* CDH *problem, our* TTDF *achieves one-wayness against selective adversaries.*

- More interestingly as illustrated in Table 1, our scheme achieves better communication bandwidth than the work of Tu et al. (2019). The image size of our scheme is $\mathcal{O}(|\mathbb{G}| + l)$ where $l = (n + nr)$ is the domain size of our scheme and $|\mathbb{G}|$ is bit size of an element of CDH-hard group $\mathbb{G}$.

In contrast, the image size ($|Y|$) of the DDH-based scheme of Tu et al. (2019) is $\mathcal{O}(l_1|\mathbb{G}_1|)$ where l_1 represents the domain size of the DDH based TTDF and $|\mathbb{G}_1|$ denotes the bit size of an element of DDH-hard group $\mathbb{G}_1$.

- As demonstrated in Table 1, the index key size ($|\mathsf{ik}|$) of the DDH-based scheme of Tu et al. (2019) is $({l_1}^2 + l_1)|\mathbb{G}_1| = \mathcal{O}(l_1^2|\mathbb{G}_1|)$ whereas it is $\mathcal{O}(l^2|\mathbb{G}|)$ in our case. The size of master trapdoor key ($|\mathsf{mtk}|$) and shared trapdoor key ($|\mathsf{tk}|$) of the DDH-based scheme of Tu et al. (2019) are respectively $\mathcal{O}(l_1 t|\mathbb{Z}_q|)$ and $\mathcal{O}(l_1|\mathbb{Z}_q|)$ where t is the threshold, prime q is the cardinality of the DDH-hard group $\mathbb{G}_1$ and $|\mathbb{Z}_q|$ indicates the bit size of an element of $\mathbb{Z}_q$. The size of master trapdoor key ($|\mathsf{mtk}|$) and shared trapdoor key ($|\mathsf{tk}|$) of our scheme are respectively $\mathcal{O}(lt|\mathbb{Z}_p|)$ and $\mathcal{O}(l|\mathbb{Z}_p|)$ where prime p is the cardinality of the CDH-hard group $\mathbb{G}$ and $|\mathbb{Z}_p|$ represents the bit size of an element of $\mathbb{Z}_p$.
- Although our scheme is only selectively secure in contrast to the adaptive secure TTDF of Tu et al. (2019) and has higher complexity in terms of storage, we emphasize that our construction is the *first* to achieve security under the CDH assumption exhibiting stronger security guarantees and favourable communication bandwidth.

2 PRELIMINARIES

2.1 Notation

Let λ represent the security parameter. We use $\stackrel{c}{\equiv}$ to denote computational indistinguishability between two distributions and $\equiv$ to imply two distributions are identical. By $x \stackrel{u}{\leftarrow} S$ we mean that x is chosen uniformly from the set S . Let HC stands for a hardcore bit function and $[n]$ represents the set $\{1, 2, \ldots, n\}$ for any $n \in \mathbb{N}$. We say $f : \mathbb{N} \rightarrow \mathbb{R}$ is a *negligible* function of n if it is $\mathcal{O}(n^{-c})$ for all $c > 0$ and we use $\mathsf{negl}(n)$ for negligible function of n. Let the symbol $\perp$ indicates either failure or null value, which is clear from the context.

2.2 Threshold trapdoor function (TTDF)

We describe below the notion of *threshold trapdoor function* (TTDF) following Tu et al. (2019). It consists of algorithms Gen, Share, Func, Func^{-1}, Combine. Informally speaking, a trusted authority runs the probabilistic algorithm Gen to generate a master trapdoor key mtk and an index key ik. It also runs the deterministic algorithm Share to issue through a secure channel a shared trapdoor key $\mathsf{tk}_{\mathsf{id}}$ to an identity holder with identity id using the master trapdoor key mtk. An evaluator executes the deterministic algorithm Func that generates an image Y of an element X using the index key ik. An identity holder with identity id deterministically can generate an inversion share δ_{id} by invoking the algorithm Func^{-1} taking the shared trapdoor key $\mathsf{tk}_{\mathsf{id}}$ and the image element Y as input. The Combine algorithm is executed by a combiner which on input the index key ik, t inversion shares $S = \{\delta_{\mathsf{id}_j} | j \in L \subseteq [N] \text{ where } |L| = t\}$ and an image element Y, reconstructs the preimage X of Y. Formal definition of TTDF is given below.

Definition 2.1. (Threshold trapdoor function (TTDF). *Let $w = w(\lambda), p = p(\lambda)$ be polynomials in the security parameter λ. A family of (N, t)-threshold trapdoor function* $\mathsf{TTDF} = (\mathsf{Gen}, \mathsf{Share}, \mathsf{Func}, \mathsf{Func}^{-1}, \mathsf{Combine})$ *is a tuple of algorithms with domain $\{0,1\}^w$ and range $\{0,1\}^p$ defined as follows:*

- $\mathsf{Gen}(1^\lambda) \rightarrow (\mathsf{ik}, \mathsf{mtk})$: *A trusted authority runs this probabilistic algorithm which takes as input the security parameter 1^λ and generates an index key* ik *and a master trapdoor key* mtk. *It publishes* ik *and keeps* mtk *secret to itself.*
- $\mathsf{Share}(\mathsf{mtk}, \mathsf{id}_j) \rightarrow \mathsf{tk}_{\mathsf{id}_j}$: *On input* mtk *and an identity* $\mathsf{id}_j, j \in [N]$, *the trusted authority deterministically computes the shared trapdoor key* $\mathsf{tk}_{\mathsf{id}_j}$ *corresponding to the identity* id_j *and sends* $\mathsf{tk}_{\mathsf{id}_j}$ *to the user with identity* id_j *through a secure channel between them.*
- $\mathsf{Func}(\mathsf{ik}, X) \rightarrow Y$: *The evaluator takes input the index key* ik, *a domain element $X \in \{0,1\}^w$ and deterministically computes an image element $Y \in \{0,1\}^p$ of $X \in \{0,1\}^w$.*
- $\mathsf{Func}^{-1}(\mathsf{tk}_{\mathsf{id}_j}, Y) \rightarrow \delta_{\mathsf{id}_j}$: *On input* $\mathsf{tk}_{\mathsf{id}_j}$ *and $Y \in \{0,1\}^p$, the identity holder* id_j *generates the inversion share* δ_{id_j} *deterministically.*

- $\mathsf{Combine}(\mathsf{ik}, S, Y) \rightarrow X$: *On input the index key* ik, *image* Y *and any* t *inversion shares* $S = \{\delta_{\mathsf{id}_j} | j \in L \subseteq [N]$ *where* $|L| = t\}$ *of the image* Y, *the combiner outputs a value* $X \in \{0,1\}^w \cup \{\perp\}$.

2.2.1 Correctness

For all identity id_j, $(\mathsf{ik}, \mathsf{mtk}) \leftarrow \mathsf{Gen}(1^\lambda)$, $\mathsf{tk}_{\mathsf{id}_j} \leftarrow \mathsf{Share}(\mathsf{mtk}, \mathsf{id}_j)$, $j \in [N]$, $X \in \{0,1\}^w$, $Y \leftarrow \mathsf{Func}(\mathsf{ik}, X)$, we require the following:
$\mathsf{Pr}[\mathsf{Combine}(\mathsf{ik}, \{\mathsf{Func}^{-1}(\mathsf{tk}_{\mathsf{id}_j}, Y) | j \in L \subseteq [N], |L| = t\}, Y) \neq X] = \mathsf{negl}(\lambda)$.

2.2.2 One-wayness

A collection of (N,t)-TTDF is called one-way if the advantage of any *probabilistic polynomial time* (PPT) adversary $\mathcal{A}$ is negligible in the following experiment with a challenger.

- The adversary $\mathcal{A}$ declares to corrupt identities id_l for $l \in A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t-1$.
- The challenger runs $(\mathsf{ik}, \mathsf{mtk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
- The challenger samples a domain element $X \in \{0,1\}^w$, computes $Y \leftarrow \mathsf{Func}(\mathsf{ik}, X)$.
- The challenger generates the shared trapdoor keys $\mathsf{tk}_{\mathsf{id}_l} \leftarrow \mathsf{Share}(\mathsf{mtk}, \mathsf{id}_l)$ for $l \in A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$.
- The challenger provides $(\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_1}, \mathsf{tk}_{\mathsf{id}_2}, \ldots, \mathsf{tk}_{\mathsf{id}_{t-1}})$ to the adversary.
- Given the view $(\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \mathsf{tk}_{\mathsf{id}_{\gamma_2}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$, the adversary outputs its guess X' for the domain element X.

Advantage of the adversary in this game is defined as:

$$\mathsf{Adv}^{\mathsf{OW}}_{\mathcal{A},\mathsf{TTDF}}(1^\lambda) = \mathsf{Pr}[X = X' \leftarrow \mathcal{A}(\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \mathsf{tk}_{\mathsf{id}_{\gamma_2}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})]$$

2.3 Recyclable one-way function with encryption

An *one-way function with encryption* (OWFE) is a weakening notion of chameleon-encryption (Döttling and Garg, 2017) and consists of four algorithms Gen, func, Encaps, Decaps. A trusted entity runs the probabilistic algorithm Gen to generate public parameter pp. On input the public parameter pp and $x \in \{0,1\}^n$, an evaluator executes the deterministic algorithm func to obtain $y \in \{0,1\}^v$. The probabilistic algorithm Encaps is run by an encapsulator on input pp, y, (i,b), ρ to produce a ciphertext ct and a bit e where $i \in [n]$, b is a bit, ρ is a random value. On input pp, x, ct, a decapsulator invokes the algorithm Decaps and outputs a bit e_1 or aborts. Formally, we define OWFE as follows:

Definition 2.2. (One-way function with encryption (OWFE)). *An* OWFE *scheme is a tuple of four algorithms* (Gen, func, Encaps, Decaps) *with the following syntax where* $\{0,1\}^n$ *is the domain,* $\{0,1\}^v$ *is the range and* $n = n(\lambda)$, $v = v(\lambda)$ *are polynomials in the security parameter* λ.

- $\mathsf{Gen}(1^\lambda) \rightarrow \mathsf{pp}$: *A trusted authority executes this probabilistic algorithm which takes as input the security parameter* 1^λ *and publishes the public parameter* pp.
- $\mathsf{func}(\mathsf{pp}, x) \rightarrow y$: *On input the public parameter* pp *and a preimage value* $x \in \{0,1\}^n$, *an evaluator deterministically generates an image element* $y \in \{0,1\}^v$.
- $\mathsf{Encaps}(\mathsf{pp}, y, (i,b); \rho) \rightarrow (\mathsf{ct}, e)$: *An encapsulator on input the public parameter* pp, *a value* y, *an index* $i \in [n]$, *a bit* $b \in \{0,1\}$ *and randomness* ρ, *computes a ciphertext* ct *and a bit* e. *It publishes* ct *and keeps* e *secret to itself.*
- $\mathsf{Decaps}(\mathsf{pp}, x, \mathsf{ct}) \rightarrow e_1$: *On input the public parameter* pp, *a preimage value* x *and a ciphertext* ct, *a decapsulator deterministically outputs* $e_1 \in \{0,1\} \cup \{\perp\}$.

2.3.1 Correctness

For all $\mathsf{pp} \leftarrow \mathsf{Gen}(1^\lambda)$, $i \in [n]$, $x \in \{0,1\}^n$ and randomness ρ, if $y \leftarrow \mathsf{func}(\mathsf{pp}, x)$, $b = x_i$, $(\mathsf{ct}, e) \leftarrow \mathsf{Encaps}(\mathsf{pp}, y, (i,b); \rho)$ where x_i is the i-th bit of $x \in \{0,1\}^n$, then $e = e_1$ where $e_1 \leftarrow \mathsf{Decaps}(\mathsf{pp}, x, \mathsf{ct})$.

2.3.2 One-wayness

We say that an OWFE scheme is one-way if for any adversary $\mathcal{A}$, $\Pr[\mathsf{func}(\mathsf{pp}, \mathcal{A}(\mathsf{pp}, y)) = y] = \mathsf{negl}(\lambda)$, where $\mathsf{pp} \leftarrow \mathsf{Gen}(1^\lambda)$, $x \xleftarrow{u} \{0,1\}^n$ and $y \leftarrow \mathsf{func}(\mathsf{pp}, x)$.

2.3.3 Security for encryption

For all $i \in [n]$ and $x \in \{0,1\}^n$, we have $(x, \mathsf{pp}, \mathsf{ct}, e) \stackrel{c}{\equiv} (x, \mathsf{pp}, \mathsf{ct}, e')$, where $\mathsf{pp} \leftarrow \mathsf{Gen}(1^\lambda)$, $(\mathsf{ct}, e) \leftarrow \mathsf{Encaps}(\mathsf{pp}, \mathsf{func}(\mathsf{pp}, x), (i, 1-x_i); \rho)$, $e' \xleftarrow{u} \{0,1\}$, x_i is the i-th bit of $x \in \{0,1\}^n$, ρ is randomness.

Garg and Hajiabadi (2018) introduced the notion of *recyclable* OWFE defined below.

Definition 2.3. (Recyclability). *Let* Encaps_1 *and* Encaps_2 *denote the first and second output of* Encaps *of an* OWFE *scheme* $\mathsf{OWFE} = (\mathsf{Gen}, \mathsf{func}, \mathsf{Encaps} = (\mathsf{Encaps}_1, \mathsf{Encaps}_2), \mathsf{Decaps})$. *We say that the* OWFE *scheme is recyclable if the value of* $\mathsf{Encaps}_1(\mathsf{pp}, y, (i,b); \rho) \rightarrow \mathsf{ct}$ *is independent of* y*, only* $\mathsf{Encaps}_2(\mathsf{pp}, y, (i,b); \rho) \rightarrow e$ *depends on* y*. In other words, for any* $\mathsf{pp} \leftarrow \mathsf{Gen}(1^\lambda)$, $y_1, y_2 \in \{0,1\}^v$, $i \in [n]$, $b \in \{0,1\}$ *and randomness* ρ*, we have*

$$\mathsf{Encaps}_1(\mathsf{pp}, y_1, (i,b); \rho) = \mathsf{Encaps}_1(\mathsf{pp}, y_2, (i,b); \rho)$$

Henceforth, we omit y *as an input of* Encaps_1 *and write* $\mathsf{ct} \leftarrow \mathsf{Encaps}_1(\mathsf{pp}, (i,b); \rho)$.

2.3.4 Instantiation of recyclable OWFE

We describe below the construction of recyclable OWFE of Garg and Hajiabadi (2018).

- $\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$: A trusted authority runs a group generator scheme $\mathcal{G}(1^\lambda) \to (\mathbb{G}, p, g)$ where $\mathbb{G}$ is the description of a group, a prime number p is the order of the group and g is a generator of the group. Let $\{0,1\}^n$ be the domain and $\mathbb{G}$ be the range where $n = n(\lambda)$ is a polynomial in the security parameter λ. For each $j \in [n]$ and $b \in \{0,1\}$, the trusted authority chooses $g_{j,b} \xleftarrow{u} \mathbb{G}$ and sets the public parameter $\mathsf{pp} = \left(\mathbb{G}, p, g, \begin{pmatrix} g_{1,0} & g_{2,0} & \cdots & g_{n,0} \\ g_{1,1} & g_{2,1} & \cdots & g_{n,1} \end{pmatrix}\right)$.

- $\mathsf{func}(\mathsf{pp}, x) \to y$: An evaluator takes pp and $x = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$ and computes $y = \prod_{j \in [n]} g_{j,x_j} \in \mathbb{G}$.

- $\mathsf{Encaps}(\mathsf{pp}, y, (i,b); \rho) = (\mathsf{Encaps}_1(\mathsf{pp}, (i,b); \rho), \mathsf{Encaps}_2(\mathsf{pp}, y, (i,b); \rho)) \to (\mathsf{ct}, e)$: On input pp, $y, i \in [n]$, a bit $b \in \{0,1\}$, an encapsulator samples randomness $\rho \xleftarrow{u} \mathbb{Z}_p$ and proceeds as follows:

  1. Sets $\mathsf{ct} = \begin{pmatrix} c_{1,0} & c_{2,0} & \cdots & c_{n,0} \\ c_{1,1} & c_{2,1} & \cdots & c_{n,1} \end{pmatrix}$ where $c_{j,0} = (g_{j,0})^\rho$, $c_{j,1} = (g_{j,1})^\rho$ for every $j \in [n] \setminus \{i\}$ and $c_{i,b} = (g_{i,b})^\rho$, $c_{i,1-b} = \perp$.
  2. Computes $e = \mathsf{HC}(y^\rho) \in \{0,1\}$ where HC is a hardcore bit function.
  3. Sets $\mathsf{Encaps}_1(\mathsf{pp}, (i,b); \rho) \to \mathsf{ct}$ and $\mathsf{Encaps}_2(\mathsf{pp}, y, (i,b); \rho) \to e$.
  4. Sends ct to the decapsulator and keeps e secret to itself.

- $\mathsf{Decaps}(\mathsf{pp}, x, \mathsf{ct}) \to e_1$: On receiving ciphertext $\mathsf{ct} = \begin{pmatrix} c_{1,0} & c_{2,0} & \cdots & c_{n,0} \\ c_{1,1} & c_{2,1} & \cdots & c_{n,1} \end{pmatrix}$, a decapsulator uses pp and $x = (x_1, x_2, \ldots, x_n)$ and computes $e_1 = \mathsf{HC}(\prod_{j \in [n]} c_{j,x_j})$, where x_j is the j-th bit of $x \in \{0,1\}^n$.

2.3.5 Correctness

For all $\mathsf{pp} \leftarrow \mathsf{Gen}(1^\lambda), i \in [n], x \in \{0,1\}^n$ and randomness ρ, if $y \leftarrow \mathsf{func}(\mathsf{pp}, x)$ and $b = x_i$, then $\mathsf{Decaps}(\mathsf{pp}, x, \mathsf{ct}) \rightarrow e_1 = \mathsf{HC}(\prod_{j\in[n]} c_{j,x_j}) = \mathsf{HC}(\prod_{j\in[n]} (g_{j,x_j})^\rho) = \mathsf{HC}(y^\rho) = e$.

The above scheme achieves *one-wayness,* security for encryption assuming that Computational Diffie-Hellman (CDH) problem defined below is hard for $\mathbb{G}$ and $n \in \omega(\log p)$.

Definition 2.4. (Computational Diffie-Hellman (CDH) Assumption). *Let $\mathcal{G}$ be a group-generator scheme which on input 1^λ, outputs $(\mathbb{G}, p, g)$ where $\mathbb{G}$ is the description of a group, a prime number p is the order of the group and g is a generator of the group. We say that $\mathcal{G}$ is CDH-hard if for any adversary $\mathcal{A}$, $\mathsf{Pr}[\mathcal{A}(\mathbb{G}, p, g, g^{a_1}, g^{a_2}) = g^{a_1 a_2}] = \mathsf{negl}(\lambda)$ where $(\mathbb{G}, p, g) \leftarrow \mathcal{G}(1^\lambda)$ and $a_1, a_2 \xleftarrow{u} \mathbb{Z}_p$.*

2.4 Shamir's (N, t) -threshold scheme

Let t, N be positive integers with $t \leq N$ and $\mathbb{F}$ be a finite field with $|\mathbb{F}| > N$. Let $\mathsf{id}_i \in \mathbb{F}$ for $i \in [N]$ represent publicly available distinct nonzero elements. An (N, t)-threshold scheme (Shamir, 1979) is a method of sharing a secret $s \in \mathbb{F}$ among N users in such a way that any t users can recover s, but no colluding subset of $t-1$ users can do so. It is a tuple of two algorithms (Share, Combine) with the following syntax.

- $\mathsf{Share}(s, \mathsf{id}_i) \rightarrow s_i$: To share a secret $s \in \mathbb{F}$, the trusted authority randomly chooses $a_1, a_2, \ldots, a_{t-1} \in \mathbb{F}$ and constructs a polynomial $a(x) = s + \sum_{k=1}^{t-1} a_k x^k$. It provides the share $s_i = a(\mathsf{id}_i)$ of the secret s to the identity holder id_i for $i \in [N]$.
- $\mathsf{Combine}\{(s_i, \mathsf{id}_i) : i \in L \subseteq [N], |L| = t\} \rightarrow s$: On input any t identities with their associated shares, the combiner computes the polynomial $a(x)$ using Lagrange's interpolation and returns $s = a(0)$.

2.4.1 Correctness

By Lagrange's interpolation formula we have,

$$a(x) = \sum_{\substack{l \in L \\ |L|=t}} a(\mathsf{id}_l) \prod_{\substack{j \in L \\ j \neq l}} \frac{x - \mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j} = \sum_{\substack{l \in L \\ |L|=t}} s_l \prod_{\substack{j \in L \\ j \neq l}} \frac{x - \mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$$

So, the combiner correctly computes the polynomial $a(x)$ using t identities with their associated shares and recovers the correct value of the secret $s = a(0)$.

The above scheme achieves perfect privacy because no colluding subset of $t-1$ users learn any information of the secret s from their shares. These $t-1$ shares of the secret look like uniformly chosen random numbers.

3 CONSTRUCTION OF TTDF

Our construction of $\mathsf{TTDF} = (\mathsf{Gen}, \mathsf{Share}, \mathsf{Func}, \mathsf{Func}^{-1}, \mathsf{Combine})$ uses the CDH-based *recyclable* $\mathsf{OWFE} = (\mathsf{Gen}, \mathsf{func}, \mathsf{Encaps} = (\mathsf{Encaps}_1, \mathsf{Encaps}_2), \mathsf{Decaps})$ of Garg and Hajiabadi (2018) described in Section 2 with the following extended notation.

We define $\mathsf{OWFE.Decaps}(\mathsf{pp}, x, \mathsf{ct})$ to be the concatenation of $\mathsf{OWFE.Decaps}(\mathsf{pp}, x, \mathsf{ct}_i)$ for $i \in [r]$ where $\mathsf{ct} = (\mathsf{ct}_1, \mathsf{ct}_2, \ldots, \mathsf{ct}_r)$ is a sequence of encapsulated ciphertexts, pp is a given public parameter and $x \in \{0,1\}^n$ is a known value. The input space of TTDF is $\{0,1\}^{n+nr}$. We use an algorithm Perm that works as follows. For two lists u_1, u_2 and a bit b,

$$\mathsf{Perm}(u_1, u_2, b) = \begin{cases} (u_1, u_2) & \text{if } b = 0 \\ (u_2, u_1) & \text{if } b = 1 \end{cases}$$

- $\mathsf{TTDF.Gen}(1^\lambda) \rightarrow (\mathsf{ik}, \mathsf{mtk})$: A trusted authority does the following to generate an index key ik and a master trapdoor key mtk taking a security parameter 1^λ as input.

1. By running a group generator scheme $\mathcal{G}(1^\lambda)$ to generate $(\mathbb{G}, p, g)$ where $\mathbb{G}$ is the description of a group, a prime number p is the order of the group and g is a generator of the group and choosing $g_{j,b} \xleftarrow{u} \mathbb{G}$ for each $j \in [n]$, bit $b \in \{0,1\}$, generates the public parameter

$$\mathsf{pp} = \left(\mathbb{G}, p, g, \begin{pmatrix} g_{1,0} & g_{2,0} & \cdots & g_{n,0} \\ g_{1,1} & g_{2,1} & \cdots & g_{n,1} \end{pmatrix}\right) \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$$

2. For each $i \in [n]$ and bit $b \in \{0,1\}$, picks $\rho_{i,b} = (\rho_{i,b,1}, \rho_{i,b,2}, \ldots, \rho_{i,b,r}) \xleftarrow{u} \mathbb{Z}_p^r$ and sets $\mathsf{ct}_{i,b} = (\mathsf{ct}_{i,b,1}, \mathsf{ct}_{i,b,2}, \ldots, \mathsf{ct}_{i,b,r})$ where $\mathsf{ct}_{i,b,k} = \begin{pmatrix} c_{1,0,k} & c_{2,0,k} & \cdots & c_{n,0,k} \\ c_{1,1,k} & c_{2,1,k} & \cdots & c_{n,1,k} \end{pmatrix} \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,b); \rho_{i,b,k})$ with $c_{j,0,k} = (g_{j,0})^{\rho_{i,b,k}}$, $c_{j,1,k} = (g_{j,1})^{\rho_{i,b,k}}$ for every $j \in [n] \setminus \{i\}$ and $c_{i,b,k} = (g_{i,b})^{\rho_{i,b,k}}$, $c_{i,1-b,k} = \perp$.
3. For each $i \in [n]$ and bit $b \in \{0,1\}$, selects $d_{i,b} = (d_{i,b,1}, d_{i,b,2}, \ldots, d_{i,b,r}) \xleftarrow{u} \mathbb{Z}_p^{r(t-1)}$ where $d_{i,b,k} = (d_{i,b,k,1}, d_{i,b,k,2}, \ldots, d_{i,b,k,t-1}) \xleftarrow{u} \mathbb{Z}_p^{(t-1)}$ for $k \in [r]$, t is the required threshold with $t \leq N$. Here N represents the total number of parties.
4. Sets the index key ik as

$$\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), (\mathsf{ct}_{2,0}, \mathsf{ct}_{2,1}), \ldots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$$

and the master trapdoor key mtk as

$$\mathsf{mtk} = ((\rho_{1,0}, d_{1,0}), (\rho_{1,1}, d_{1,1}), \ldots, (\rho_{n,0}, d_{n,0}), (\rho_{n,1}, d_{n,1}))$$

- $\mathsf{TTDF.Share}(\mathsf{mtk}, \mathsf{id}_j) \rightarrow \mathsf{tk}_{\mathsf{id}_j}$: On input the master trapdoor key mtk and an identity $\mathsf{id}_j \in \mathbb{Z}_p^\star = \mathbb{Z}_p \setminus \{0\}$, $j \in [N]$, the trusted authority constructs polynomials

$$h_{i,b,k}(z) = \rho_{i,b,k} + d_{i,b,k,1} z + \cdots + d_{i,b,k,t-1} z^{t-1}$$

for $i \in [n], b \in \{0,1\}, k \in [r]$ and computes the shared trapdoor key $\mathsf{tk}_{\mathsf{id}_j} \in \mathbb{Z}_p^{2n \times r}$ as

$$\mathsf{tk}_{\mathsf{id}_j} = \begin{pmatrix} h_{1,0,1}(\mathsf{id}_j) & h_{1,0,2}(\mathsf{id}_j) & \cdots & h_{1,0,r}(\mathsf{id}_j) \\ h_{1,1,1}(\mathsf{id}_j) & h_{1,1,2}(\mathsf{id}_j) & \cdots & h_{1,1,r}(\mathsf{id}_j) \\ \vdots & \vdots & \vdots & \vdots \\ h_{n,0,1}(\mathsf{id}_j) & h_{n,0,2}(\mathsf{id}_j) & \cdots & h_{n,0,r}(\mathsf{id}_j) \\ h_{n,1,1}(\mathsf{id}_j) & h_{n,1,1}(\mathsf{id}_j) & \cdots & h_{n,1,r}(\mathsf{id}_j) \end{pmatrix}$$

It sends $\mathsf{tk}_{\mathsf{id}_j}$ through a secure channel to the user with identity id_j.

- $\mathsf{TTDF.Func}(\mathsf{ik}, X) \to Y$: An evaluator executes the following steps to generate an image Y of $X = (x = (x_1, x_2, \ldots, x_n), \beta = (\beta_1, \beta_2, \ldots, \beta_n))$ where $x_i \in \{0,1\}$ and $\beta_i \in \{0,1\}^r$ for $i \in [n]$ using the index key $\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), (\mathsf{ct}_{2,0}, \mathsf{ct}_{2,1}), \ldots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$.

  1. Parses $\mathsf{pp} = \left(\mathbb{G}, p, g, \begin{pmatrix} g_{1,0} & g_{2,0} & \cdots & g_{n,0} \\ g_{1,1} & g_{2,1} & \cdots & g_{n,1} \end{pmatrix}\right)$, extracts g_{j,x_j} for $j \in [n]$ from pp and computes $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ where $y = \prod_{j \in [n]} g_{j,x_j} \in \mathbb{G}$.
  2. Sets $e_i \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \mathsf{ct}_{i,x_i})$ where $e_i = (e_{i,1}, e_{i,2}, \ldots, e_{i,r}) \in \{0,1\}^r$, $e_{i,k} \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \mathsf{ct}_{i,x_i,k})$ for $k \in [r]$ where $e_{i,k} = \mathsf{HC}(\prod_{j \in [n]} c_{j,x_j,k})$ for $k \in \{1, 2, \ldots, r\}$ with $c_{j,0,k} = (g_{j,0})^{\rho_{i,x_i,k}}$, $c_{j,1,k} = (g_{j,1})^{\rho_{i,x_i,k}}$ for every $j \in \{1, 2, \ldots, n\} \setminus \{i\}$ and $c_{i,x_i,k} = (g_{i,x_i})^{\rho_{i,x_i,k}}$, $c_{i,1-x_i,k} = \perp$.
  3. Generates $Y = (y, \mathsf{Perm}(e_1, \beta_1, x_1), \ldots, \mathsf{Perm}(e_n, \beta_n, x_n))$ where for each $i \in [n]$,

$$\mathsf{Perm}(e_i, \beta_i, x_i) = \begin{cases} (e_i, \beta_i) & \text{if } x_i = 0 \\ (\beta_i, e_i) & \text{if } x_i = 1 \end{cases}$$

- $\mathsf{TTDF.Func}^{-1}(\mathsf{tk}_{\mathsf{id}_j}, Y) \to \delta_{\mathsf{id}_j}$: On input the shared trapdoor key

$$\mathsf{tk}_{\mathsf{id}_j} = \begin{pmatrix} h_{1,0,1}(\mathsf{id}_j) & h_{1,0,2}(\mathsf{id}_j) & & h_{1,0,r}(\mathsf{id}_j) \\ h_{1,1,1}(\mathsf{id}_j) & h_{1,1,2}(\mathsf{id}_j) & \cdots & h_{1,1,r}(\mathsf{id}_j) \\ \vdots & \vdots & \vdots & \vdots \\ h_{n,0,1}(\mathsf{id}_j) & h_{n,0,2}(\mathsf{id}_j) & \cdots & h_{n,0,r}(\mathsf{id}_j) \\ h_{n,1,1}(\mathsf{id}_j) & h_{n,1,1}(\mathsf{id}_j) & & h_{n,1,r}(\mathsf{id}_j) \end{pmatrix}$$

and $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \ldots, (\sigma_{n,0}, \sigma_{n,1}))$, the user with identity id_j sets the inversion share $\delta_{\mathsf{id}_j} \in \mathbb{G}^{2n \times r}$ as

$$\delta_{\mathsf{id}_j} = \begin{pmatrix} y^{h_{1,0,1}(\mathsf{id}_j)} & y^{h_{1,0,2}(\mathsf{id}_j)} & \dots & y^{h_{1,0,r}(\mathsf{id}_j)} \\ y^{h_{1,1,1}(\mathsf{id}_j)} & y^{h_{1,1,2}(\mathsf{id}_j)} & \dots & y^{h_{1,1,r}(\mathsf{id}_j)} \\ \vdots & \vdots & \vdots & \vdots \\ y^{h_{n,0,1}(\mathsf{id}_j)} & y^{h_{n,0,2}(\mathsf{id}_j)} & \dots & y^{h_{n,0,r}(\mathsf{id}_j)} \\ y^{h_{n,1,1}(\mathsf{id}_j)} & y^{h_{n,1,2}(\mathsf{id}_j)} & \dots & y^{h_{n,1,r}(\mathsf{id}_j)} \end{pmatrix}$$

- $\mathsf{TTDF.Combine}(\mathsf{ik}, S, Y) \rightarrow X$: On receiving the index key $\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), (\mathsf{ct}_{2,0}, \mathsf{ct}_{2,1}), \dots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$, a set $S = \{\delta_{\mathsf{id}_j} | j \in L \subseteq [N]$ where $|L| = t\}$ of t inversion shares and $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \dots, (\sigma_{n,0}, \sigma_{n,1}))$, the combiner reconstructs $x = (x_1, x_2, \dots, x_n) \in \{0,1\}^n$ bit-by-bit and $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ vector-by-vector as follows where $\beta_i \in \{0,1\}^r$ for $i \in [n]$.

  1. For each $i \in [n], b \in \{0,1\}, k \in [r]$, computes $y^{\rho_{i,b,k}} = \prod_{l \in L} \left(y^{h_{i,b,k}(\mathsf{id}_l)}\right)^{\lambda_l}$ by extracting $y^{h_{i,b,k}(\mathsf{id}_l)}$ from $\delta_{\mathsf{id}_l} \in S$ and computes Lagrange's coefficient $\lambda_l = \prod_{\substack{j \in L \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ using the public identities $\mathsf{id}_j, j \in L, j \neq l$. Note that $\rho_{i,b,k}$ is the constant coefficient of the polynomial $h_{i,b,k}(z)$ and so $\rho_{i,b,k} = h_{i,b,k}(0) = \sum_{l \in L} \lambda_l h_{i,b,k}(\mathsf{id}_l)$ for $k \in [r], i \in [n], b \in \{0,1\}$ by Lagrange's polynomial interpolation. Thus

$$y^{\rho_{i,b,k}} = y^{h_{i,b,k}(0)} = \prod_{l \in L} \left(y^{h_{i,b,k}(\mathsf{id}_l)}\right)^{\lambda_l}$$

  2. Returns $x_i = 0, \beta_i = \sigma_{i,1} = \sigma_{i,1-x_i}$ if $\sigma_{i,0} = \left(\mathsf{HC}(y^{\rho_{i,0,1}}), \mathsf{HC}(y^{\rho_{i,0,2}}), \dots, \mathsf{HC}(y^{\rho_{i,0,r}})\right), \sigma_{i,1} \neq \left(\mathsf{HC}(y^{\rho_{i,1,1}}), \mathsf{HC}(y^{\rho_{i,1,2}}), \dots, \mathsf{HC}(y^{\rho_{i,1,r}})\right)$ and returns $x_i = 1, \beta_i = \sigma_{i,0} = \sigma_{i,1-x_i}$ if $\sigma_{i,0} \neq \left(\mathsf{HC}(y^{\rho_{i,0,1}}), \mathsf{HC}(y^{\rho_{i,0,2}}), \dots, \mathsf{HC}(y^{\rho_{i,0,r}})\right), \sigma_{i,1} = \left(\mathsf{HC}(y^{\rho_{i,1,1}}), \mathsf{HC}(y^{\rho_{i,1,2}}), \dots, \mathsf{HC}(y^{\rho_{i,1,r}})\right)$
  3. Sets $x = (x_1, x_2, \dots, x_n)$ and $\beta = (\beta_1, \beta_2, \dots, \beta_n)$.

4. Computes $y' \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ by extracting pp from index key ik and returns $x = (x_1, x_2, \ldots, x_n), \beta = (\beta_1, \beta_2, \cdots, \beta_n)$ if $y' = y$. Otherwise, returns $\perp$.

3.1 Correctness

The inversion error of the constructed TTDF is at most $\frac{n}{2^r}$ as follows from the following argument. By choosing $r \in \omega(\log \lambda)$ we will have negligible inversion error.
Let $(\mathsf{ik}, \mathsf{mtk}) \leftarrow \mathsf{TTDF.Gen}(1^\lambda)$, $\mathsf{tk}_{\mathsf{id}_j} \leftarrow \mathsf{TTDF.Share}(\mathsf{mtk}, \mathsf{id}_j)$ for $j \in [N]$, $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \ldots, (\sigma_{n,0}, \sigma_{n,1})) \leftarrow \mathsf{TTDF.Func}(\mathsf{ik}, X)$ with $X = (x = (x_1, x_2, \ldots, x_n), \beta = (\beta_1, \beta_2, \ldots, \beta_n)) \in \{0,1\}^{n+nr}$, $\delta_{\mathsf{id}_j} \leftarrow \mathsf{TTDF.Func}^{-1}(\mathsf{tk}_{\mathsf{id}_j}, Y)$ where

$$\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), (\mathsf{ct}_{2,0}, \mathsf{ct}_{2,1}), \ldots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$$

$$\mathsf{mtk} = ((\rho_{1,0}, d_{1,0}), (\rho_{1,1}, d_{1,1}) \ldots, (\rho_{n,0}, d_{n,0}), (\rho_{n,1}, d_{n,1}))$$

$$\mathsf{tk}_{\mathsf{id}_j} = \begin{pmatrix} h_{1,0,1}(\mathsf{id}_j) & h_{1,0,2}(\mathsf{id}_j) & & h_{1,0,r}(\mathsf{id}_j) \\ h_{1,1,1}(\mathsf{id}_j) & h_{1,1,2}(\mathsf{id}_j) & \cdots & h_{1,1,r}(\mathsf{id}_j) \\ \vdots & \vdots & \vdots & \vdots \\ h_{n,0,1}(\mathsf{id}_j) & h_{n,0,2}(\mathsf{id}_j) & \cdots & h_{n,0,r}(\mathsf{id}_j) \\ h_{n,1,1}(\mathsf{id}_j) & h_{n,1,1}(\mathsf{id}_j) & & h_{n,1,r}(\mathsf{id}_j) \end{pmatrix}$$

$$\delta_{\mathsf{id}_j} = \begin{pmatrix} y^{h_{1,0,1}(\mathsf{id}_j)} & y^{h_{1,0,2}(\mathsf{id}_j)} & & y^{h_{1,0,r}(\mathsf{id}_j)} \\ y^{h_{1,1,1}(\mathsf{id}_j)} & y^{h_{1,1,2}(\mathsf{id}_j)} & \ldots & y^{h_{1,1,r}(\mathsf{id}_j)} \\ \vdots & \vdots & \vdots & \vdots \\ y^{h_{n,0,1}(\mathsf{id}_j)} & y^{h_{n,0,2}(\mathsf{id}_j)} & \ldots & y^{h_{n,0,r}(\mathsf{id}_j)} \\ y^{h_{n,1,1}(\mathsf{id}_j)} & y^{h_{n,1,2}(\mathsf{id}_j)} & \ldots & y^{h_{n,1,r}(\mathsf{id}_j)} \end{pmatrix}$$

We know that $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \ldots, (\sigma_{n,0}, \sigma_{n,1})) = (y, \mathsf{Perm}(e_1, \beta_1, x_1), \ldots, \mathsf{Perm}(e_n, \beta_n, x_n))$ where for each $i \in [n]$,

$$\mathsf{Perm}(e_i, \beta_i, x_i) = \begin{cases} (e_i, \beta_i) & \text{if } x_i = 0 \\ (\beta_i, e_i) & \text{if } x_i = 1 \end{cases}$$

$e_i \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \mathsf{ct}_{i,x_i}), e_i = (e_{i,1}, e_{i,2}, \ldots, e_{i,r}), e_{i,k} \leftarrow \mathsf{OWFE.Decaps}$ $(\mathsf{pp}, x, \mathsf{ct}_{i,x_i,k})$ for $k \in [r]$.
By the correctness of recyclable OWFE described in Section 2.3, we have $e_{i,k} = \mathsf{HC}(y^{\rho_{i,x_i,k}})$ for $k \in [r]$. Now the probability that $\mathsf{TTDF.Combine}(\mathsf{ik}, \{\mathsf{TTDF.Func}^{-1}(\mathsf{tk}_{\mathsf{id}_j}, Y) | j \in L \subseteq [N], |L| = t\}, Y) \neq X$ is the same as the probability that $\beta_i = \left(\mathsf{HC}(y^{\rho_{i,1-x_i,1}}), \mathsf{HC}(y^{\rho_{i,1-x_i,2}}), \ldots, \mathsf{HC}(y^{\rho_{i,1-x_i,r}})\right)$ for some $i \in [n]$ which is $\frac{1}{2^r}$ as β_i is chosen uniformly at random and is independent of x. Taking union over $i \in [n]$, we get the error bound as $\frac{n}{2^r}$.

4 SECURITY ANALYSIS

Theorem 4.1. *The threshold trapdoor function* $\mathsf{TTDF} = (\mathsf{Gen}, \mathsf{Share}, \mathsf{Func}, \mathsf{Func}^{-1}, \mathsf{Combine})$ *described in Section* 3 *is one-way. That is for any adversary* $\mathcal{A}$*, we have*

$$\Pr[x' = x] = \mathsf{negl}(\lambda)$$

where $x' \leftarrow \mathcal{A}(\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \mathsf{tk}_{\mathsf{id}_{\gamma_2}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$, $(\mathsf{ik}, \mathsf{mtk}) \leftarrow \mathsf{TTDF.Gen}(1^\lambda)$, $X = (x, \beta) \leftarrow \{0,1\}^{n+nr}$, $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \ldots, (\sigma_{n,0}, \sigma_{n,1})) \leftarrow \mathsf{TTDF.Func}$ (ik, X), $\mathsf{tk}_{\mathsf{id}_l} \leftarrow \mathsf{TTDF.Share}(\mathsf{mtk}, \mathsf{id}_l)$ *for* $l \in A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t - 1$.

Proof. We prove theorem 4.1 using two hybrids corresponding to the real and simulated view. Let view_i denote the view of the adversary in Hybrid i for $i \in \{0,1\}$. In Hybrid 0, the adversary's view $(\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$ is produced honestly by the challenger as described below following the real execution of the TTDF scheme.

- Samples $\mathsf{pp} = \left(\mathbb{G}, p, g, \begin{pmatrix} g_{1,0} & g_{2,0} & \cdots & g_{n,0} \\ g_{1,1} & g_{2,1} & \cdots & g_{n,1} \end{pmatrix}\right) \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$, $x \in \{0,1\}^n$ and sets $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ where $y = \prod_{j \in [n]} g_{j,x_j}$.

- For all $j \in [r]$, generates $(\mathsf{CT}_j, E_j, \mathsf{TK}_j) \leftarrow \mathsf{Real}(\mathsf{pp}, x, A)$ as described in Figure 1.

$$\mathsf{CT}_j = \begin{pmatrix} \widetilde{\mathsf{ct}}_{1,0,j} & \widetilde{\mathsf{ct}}_{2,0,j} & \cdots & \widetilde{\mathsf{ct}}_{n,0,j} \\ \widetilde{\mathsf{ct}}_{1,1,j} & \widetilde{\mathsf{ct}}_{2,1,j} & \cdots & \widetilde{\mathsf{ct}}_{n,1,j} \end{pmatrix}, E_j = \begin{pmatrix} \alpha_{1,0,j} & \alpha_{2,0,j} & \cdots & \alpha_{n,0,j} \\ \alpha_{1,1,j} & \alpha_{2,1,j} & \ldots, & \alpha_{n,1,j} \end{pmatrix}$$

$$\mathsf{TK}_j = \begin{pmatrix} f_{1,0,j}(\mathsf{id}_{\gamma_1}) & f_{1,0,j}(\mathsf{id}_{\gamma_2}) & \cdots & f_{1,0,j}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{1,1,j}(\mathsf{id}_{\gamma_1}) & f_{1,1,j}(\mathsf{id}_{\gamma_2}) & \cdots & f_{1,1,j}(\mathsf{id}_{\gamma_{t-1}}) \\ \vdots & \vdots & \vdots & \vdots \\ f_{n,0,j}(\mathsf{id}_{\gamma_1}) & f_{n,0,j}(\mathsf{id}_{\gamma_2}) & \cdots & f_{n,0,j}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{n,1,j}(\mathsf{id}_{\gamma_1}) & f_{n,1,j}(\mathsf{id}_{\gamma_2}) & \cdots & f_{n,1,j}(\mathsf{id}_{\gamma_{t-1}}) \end{pmatrix}$$

  Here for $i \in [n]$, $\alpha_{i,0,j} \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widetilde{\mathsf{ct}}_{i,0,j})$, $\alpha_{i,1,j} \xleftarrow{u} \{0,1\}$ if $x_i = 0$ and $\alpha_{i,0,j} \xleftarrow{u} \{0,1\}$, $\alpha_{i,1,j} \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widetilde{\mathsf{ct}}_{i,1,j})$ if $x_i = 1$.

- For all $i \in [n]$ and $b \in \{0,1\}$, sets $\mathsf{ct}_{i,b} = (\widetilde{\mathsf{ct}}_{i,b,1}, \widetilde{\mathsf{ct}}_{i,b,2}, \ldots, \widetilde{\mathsf{ct}}_{i,b,r})$, $\sigma_{i,b} = (\alpha_{i,b,1}, \alpha_{i,b,2}, \ldots, \alpha_{i,b,r}) \in \{0,1\}^r$.

- Forms the view $\mathsf{view}_0 = (\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$ by setting

  $\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), \ldots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$, $Y = (y, (\sigma_{1,0}, \sigma_{1,1}) \ldots, (\sigma_{n,0}, \sigma_{n,1}))$

  for $i \in [t-1]$,

$$\mathsf{tk}_{\mathsf{id}_{\gamma_i}} = \begin{pmatrix} f_{1,0,1}(\mathsf{id}_{\gamma_i}) & f_{1,0,2}(\mathsf{id}_{\gamma_i}) & \cdots & f_{1,0,r}(\mathsf{id}_{\gamma_i}) \\ f_{1,1,1}(\mathsf{id}_{\gamma_i}) & f_{1,1,2}(\mathsf{id}_{\gamma_i}) & \cdots & f_{1,1,r}(\mathsf{id}_{\gamma_i}) \\ \vdots & \vdots & \vdots & \vdots \\ f_{n,0,1}(\mathsf{id}_{\gamma_i}) & f_{n,0,2}(\mathsf{id}_{\gamma_i}) & \cdots & f_{n,0,r}(\mathsf{id}_{\gamma_i}) \\ f_{n,1,1}(\mathsf{id}_{\gamma_i}) & f_{n,1,2}(\mathsf{id}_{\gamma_i}) & \cdots & f_{n,1,r}(\mathsf{id}_{\gamma_i}) \end{pmatrix}$$

In Hybrid 1, the view $\mathsf{view}_1 = (\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$ is produced by the challenger similarly as Hybrid 0 except that for all $j \in [r]$, samples $(\mathsf{CT}_j, E_{\mathsf{Sim},j}, \mathsf{TK}_j) \leftarrow \mathsf{Sim}(\mathsf{pp}, y, A, B)$ as described in Figure 2 where $B \subseteq [N], |B| = t$. Note that $\mathsf{CT}_j, \mathsf{TK}_j$ are similar as Hybrid 0 and $E_{\mathsf{Sim},j}$ is defined as

$$E_{\mathsf{Sim},j} = \begin{pmatrix} \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_{1,0,j}(\mathsf{id}_l)}\right)^{\lambda_l}\right) & \ldots & \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_{n,0,j}(\mathsf{id}_l)}\right)^{\lambda_l}\right) \\ \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_{1,1,j}(\mathsf{id}_l)}\right)^{\lambda_l}\right) & \ldots & \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_{n,1,j}(\mathsf{id}_l)}\right)^{\lambda_l}\right) \end{pmatrix}$$

On input public parameter $\mathsf{pp} = \left(\mathbb{G}, p, g, \begin{pmatrix} g_{1,0} & g_{2,0} & \cdots & g_{n,0} \\ g_{1,1} & g_{2,1} & \cdots & g_{n,1} \end{pmatrix}\right) \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$, $x = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$, a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$, this algorithm proceeds as follows to generate a matrix CT of ciphertexts , $E \leftarrow \{0,1\}^{2\times n}$ and a matrix TK of trapdoor keys.

1. Samples $\begin{pmatrix} \widetilde{\rho}_{1,0} & \widetilde{\rho}_{2,0} & \cdots & \widetilde{\rho}_{n,0} \\ \widetilde{\rho}_{1,1} & \widetilde{\rho}_{2,1} & \cdots & \widetilde{\rho}_{n,1} \end{pmatrix} \xleftarrow{u} (\mathbb{Z}_p)^{2\times n}$ and $\begin{pmatrix} \widetilde{d}_{1,0} & \widetilde{d}_{2,0} & \cdots & \widetilde{d}_{n,0} \\ \widetilde{d}_{1,1} & \widetilde{d}_{2,1} & \cdots & \widetilde{d}_{n,1} \end{pmatrix} \xleftarrow{u} (\mathbb{Z}_p)^{2\times n(t-1)}$ where $\widetilde{d}_{i,b} = (\widetilde{d}_{i,b,1}, \ldots, \widetilde{d}_{i,b,t-1})$ for $i \in [n], b \in \{0,1\}$.

2. Constructs polynomials $f_{i,b}(z) = \widetilde{\rho}_{i,b} + \widetilde{d}_{i,b,1}z + \ldots + \widetilde{d}_{i,b,t-1}z^{t-1}$ for each $i \in [n], b \in \{0,1\}$.

3. Sets $\mathsf{CT} = \begin{pmatrix} \widetilde{\mathsf{ct}}_{1,0} & \widetilde{\mathsf{ct}}_{2,0} & \cdots & \widetilde{\mathsf{ct}}_{n,0} \\ \widetilde{\mathsf{ct}}_{1,1} & \widetilde{\mathsf{ct}}_{2,1} & \cdots & \widetilde{\mathsf{ct}}_{n,1} \end{pmatrix}$ where for $i \in [n]$, $b \in \{0,1\}$ $\widetilde{\mathsf{ct}}_{i,b} = \begin{pmatrix} c_{1,0} & c_{2,0} & \cdots & c_{n,0} \\ c_{1,1} & c_{2,1} & \cdots & c_{n,1} \end{pmatrix} \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,b); \widetilde{\rho}_{i,b})$ with $c_{j,0} = (g_{j,0})^{\widetilde{\rho}_{i,b}}$, $c_{j,1} = (g_{j,1})^{\widetilde{\rho}_{i,b}}$ for every $j \in [n] \setminus \{i\}$ and $c_{i,b} = (g_{i,b})^{\widetilde{\rho}_{i,b}}$, $c_{i,1-b} = \perp$.

4. Generates $E = \begin{pmatrix} \alpha_{1,0} & \alpha_{2,0} & \cdots & \alpha_{n,0} \\ \alpha_{1,1} & \alpha_{2,1} & \cdots & \alpha_{n,1} \end{pmatrix}$ where for $i \in [n]$ $\alpha_{i,0} \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widetilde{\mathsf{ct}}_{i,0})$, $\alpha_{i,1} \xleftarrow{u} \{0,1\}$ if $x_i = 0$ and $\alpha_{i,0} \xleftarrow{u} \{0,1\}$, $\alpha_{i,1} \leftarrow \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widetilde{\mathsf{ct}}_{i,1})$ if $x_i = 1$.

5. Sets $\mathsf{TK} = \begin{pmatrix} f_{1,0}(\mathsf{id}_{\gamma_1}) & f_{1,0}(\mathsf{id}_{\gamma_2}) & & f_{1,0}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{1,1}(\mathsf{id}_{\gamma_1}) & f_{1,1}(\mathsf{id}_{\gamma_2}) & & f_{1,1}(\mathsf{id}_{\gamma_{t-1}}) \\ \vdots & \vdots & & \vdots \\ f_{n,0}(\mathsf{id}_{\gamma_1}) & f_{n,0}(\mathsf{id}_{\gamma_2}) & & f_{n,0}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{n,1}(\mathsf{id}_{\gamma_1}) & f_{n,1}(\mathsf{id}_{\gamma_2}) & & f_{n,1}(\mathsf{id}_{\gamma_{t-1}}) \end{pmatrix}$ for a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t-1$.

Figure 1: Algorithm $\mathsf{Real}(\mathsf{pp}, x, A) \rightarrow (\mathsf{CT}, E, \mathsf{TK})$

where $B \subseteq [N], |B| = t$ and λ_l is the Lagrange's coefficient given by $\lambda_l = \prod_{\substack{j\in B \\ j\neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$.

To prove theorem 4.1 we have to show that the probability that an adversary $\mathcal{A}$ on input $\mathsf{view}_0 = (\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$ outputs $x' \in \{0,1\}^n$ such that $x = x'$ is negligible. This follows immediately from the following two

On input public parameter pp , $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$, a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$, a set $B \subseteq [N]$ with $|B| = t$, this algorithm does the following to generate a matrix CT of ciphertexts, $E_{\mathsf{Sim}} \leftarrow \{0,1\}^{2\times n}$, a matrix TK of trapdoor keys.

1. Samples $\begin{pmatrix} \widetilde{\rho}_{1,0} & \widetilde{\rho}_{2,0} & \cdots & \widetilde{\rho}_{n,0} \\ \widetilde{\rho}_{1,1} & \widetilde{\rho}_{2,1} & \cdots & \widetilde{\rho}_{n,1} \end{pmatrix} \xleftarrow{u} (\mathbb{Z}_p)^{2\times n}$ and $\begin{pmatrix} \widetilde{d}_{1,0} & \widetilde{d}_{2,0} & \cdots & \widetilde{d}_{n,0} \\ \widetilde{d}_{1,1} & \widetilde{d}_{2,1} & \cdots & \widetilde{d}_{n,1} \end{pmatrix} \xleftarrow{u} (\mathbb{Z}_p)^{2\times n(t-1)}$ where $\widetilde{d}_{i,b} = (\widetilde{d}_{i,b,1}, \ldots, \widetilde{d}_{i,b,t-1})$ for $i \in [n], b \in \{0,1\}$.

2. Constructs polynomials $f_{i,b}(z) = \widetilde{\rho}_{i,b} + \widetilde{d}_{i,b,1}z + \ldots + \widetilde{d}_{i,b,t-1}z^{t-1}$ for each $i \in [n], b \in \{0,1\}$.

3. Sets $\mathsf{CT} = \begin{pmatrix} \widetilde{\mathsf{ct}}_{1,0} & \widetilde{\mathsf{ct}}_{2,0} & \cdots & \widetilde{\mathsf{ct}}_{n,0} \\ \widetilde{\mathsf{ct}}_{1,1} & \widetilde{\mathsf{ct}}_{2,1,} & \cdots & \widetilde{\mathsf{ct}}_{n,1} \end{pmatrix}$ where for $i \in [n]$, $b \in \{0,1\}$ $\widetilde{\mathsf{ct}}_{i,b} = \begin{pmatrix} c_{1,0}, & c_{2,0}, & \ldots, & c_{n,0} \\ c_{1,1}, & c_{2,1}, & \ldots, & c_{n,1} \end{pmatrix} \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,b); \widetilde{\rho}_{i,b})$ with $c_{j,0} = (g_{j,0})^{\widetilde{\rho}_{i,b}}, c_{j,1} = (g_{j,1})^{\widetilde{\rho}_{i,b}}$ for every $j \in [n] \setminus \{i\}$ and $c_{i,b} = (g_{i,b})^{\widetilde{\rho}_{i,b}}, c_{i,1-b} = \perp$.

4. Computes $y^{f_{i,b}(\mathsf{id}_l)}$ for each $i \in [n], b \in \{0,1\}, l \in B$ and sets

$$E_{\mathsf{Sim}} = \begin{pmatrix} \mathsf{HC}\left(\prod_{l\in B}\left(y^{f_{1,0}(\mathsf{id}_l)}\right)^{\lambda_l}\right) & \mathsf{HC}\left(\prod_{l\in B}\left(y^{f_{n,0}(\mathsf{id}_l)}\right)^{\lambda_l}\right) \\ \mathsf{HC}\left(\prod_{l\in B}\left(y^{f_{1,1}(\mathsf{id}_l)}\right)^{\lambda_l}\right) & \mathsf{HC}\left(\prod_{l\in B}\left(y^{f_{n,1}(\mathsf{id}_l)}\right)^{\lambda_l}\right) \end{pmatrix}$$

where λ_l is the Lagrange's coefficient given by $\lambda_l = \prod_{\substack{j\in B \\ j\neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$.

5. Sets $\mathsf{TK} = \begin{pmatrix} f_{1,0}(\mathsf{id}_{\gamma_1}) & f_{1,0}(\mathsf{id}_{\gamma_2}) & & f_{1,0}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{1,1}(\mathsf{id}_{\gamma_1}) & f_{1,1}(\mathsf{id}_{\gamma_2}) & & f_{1,1}(\mathsf{id}_{\gamma_{t-1}}) \\ \vdots & \vdots & & \vdots \\ f_{n,0}(\mathsf{id}_{\gamma_1}) & f_{n,0}(\mathsf{id}_{\gamma_2}) & & f_{n,0}(\mathsf{id}_{\gamma_{t-1}}) \\ f_{n,1}(\mathsf{id}_{\gamma_1}) & f_{n,1}(\mathsf{id}_{\gamma_2}) & & f_{n,1}(\mathsf{id}_{\gamma_{t-1}}) \end{pmatrix}$ for a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t-1$.

Figure 2: Algorithm $\mathsf{Sim}(\mathsf{pp}, y, A, B) \rightarrow (\mathsf{CT}, E_{\mathsf{Sim}}, \mathsf{TK})$

Lemmas. □

Lemma 4.1. *The two views* view_0 *and* view_1 *described above are computationally indistinguishable.*

Proof. We first prove the following claim.

Claim 4.1. *For a fix polynomial* $r = r(\lambda)$ *and* $x \in \{0,1\}^n$, $(\mathsf{pp}, x, (\mathsf{CT}_1, E_1, \mathsf{TK}_1), \ldots, (\mathsf{CT}_r, E_r, \mathsf{TK}_r)) \stackrel{c}{\equiv} (\mathsf{pp}, x, (\mathsf{CT}_1, E_{\mathsf{Sim},1}, \mathsf{TK}_1), \ldots, (\mathsf{CT}_r, E_{\mathsf{Sim},r}, \mathsf{TK}_r))$ *where* $\mathsf{pp} \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$, $(\mathsf{CT}_i, E_i, \mathsf{TK}_i) \leftarrow \mathsf{Real}(\mathsf{pp}, x, A)$, $(\mathsf{CT}_i, E_{\mathsf{Sim},i}, \mathsf{TK}_i) \leftarrow \mathsf{Sim}(\mathsf{pp}, y, A, B)$ *for all* $i \in [r], y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$.

To prove Claim 4.1, we define two algorithms SReal and SSim in Figure 3 and Figure 4 respectively. Next we prove Claim 4.2 described below.

On input $i \in [n]$, public parameter pp , $x = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$, a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$, this algorithm proceeds as follows to generate a matrix $\widehat{\mathsf{ct}}$ of ciphertexts, $\widehat{e} \in \{0,1\}^{2\times 1}$, a matrix $\widehat{\mathsf{tk}}$ of trapdoor keys.

1. Samples $\widehat{\rho}_0, \widehat{\rho}_1 \stackrel{u}{\leftarrow} \mathbb{Z}_p, \widehat{d}_0, \widehat{d}_1 \stackrel{u}{\leftarrow} \mathbb{Z}_p^{t-1}$ where $\widehat{d}_0 = (\widehat{d}_{0,1}, \widehat{d}_{0,2}, \ldots, \widehat{d}_{0,t-1})$, $\widehat{d}_1 = (\widehat{d}_{1,1}, \widehat{d}_{1,2}, \ldots, \widehat{d}_{1,t-1})$.
2. Constructs polynomials f_j for $j \in \{0,1\}$ as $f_j(z) = \widehat{\rho}_j + \widehat{d}_{j,1} z + \ldots + \widehat{d}_{j,t-1} z^{t-1}$.
3. Sets $\widehat{\mathsf{ct}} = \begin{pmatrix} \widehat{\mathsf{ct}}_0 \\ \widehat{\mathsf{ct}}_1 \end{pmatrix}$ where $\widehat{\mathsf{ct}}_0 \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,0); \widehat{\rho}_0)$ and $\widehat{\mathsf{ct}}_1 \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,1); \widehat{\rho}_1)$.
4. Generates $\widehat{e}$ as follows.
   - if $x_i = 0$ then $\widehat{e} := \begin{pmatrix} \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widehat{\mathsf{ct}}_0) \\ b \end{pmatrix}$ where $b \stackrel{u}{\leftarrow} \{0,1\}$.
   - if $x_i = 1$ then $\widehat{e} := \begin{pmatrix} b \\ \mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widehat{\mathsf{ct}}_1) \end{pmatrix}$ where $b \stackrel{u}{\leftarrow} \{0,1\}$.
5. Sets $\widehat{\mathsf{tk}} = \begin{pmatrix} f_0(\mathsf{id}_{\gamma_1}) & f_0(\mathsf{id}_{\gamma_2}) & \ldots & f_0(\mathsf{id}_{\gamma_{t-1}}) \\ f_1(\mathsf{id}_{\gamma_1}) & f_1(\mathsf{id}_{\gamma_2}) & \ldots & f_1(\mathsf{id}_{\gamma_{t-1}}) \end{pmatrix}$ for a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t-1$.

Figure 3: Algorithm $\mathsf{SReal}(i, \mathsf{pp}, x, A) \rightarrow (\widehat{\mathsf{ct}}, \widehat{e}, \widehat{\mathsf{tk}})$

Claim 4.2. *For all* $i \in [n]$ *and* $x \in \{0,1\}^n$,

$$(\mathsf{pp}, x, (\widehat{\mathsf{ct}}, \widehat{e}, \widehat{\mathsf{tk}})) \stackrel{c}{\equiv} (\mathsf{pp}, x, (\widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}}, \widehat{\mathsf{tk}}))$$

where $\mathsf{pp} \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$, $(\widehat{\mathsf{ct}}, \widehat{e}, \widehat{\mathsf{tk}}) \leftarrow \mathsf{SReal}(i, \mathsf{pp}, x, A)$ *and* $(\widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}}, \widehat{\mathsf{tk}}) \leftarrow \mathsf{SSim}(i, \mathsf{pp}, y, A, B)$.

On input $i \in [n]$, public parameter pp, $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$, a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$, a set $B \subseteq [N]$ with $|B| = t$, this algorithm does the following to generate a matrix $\widehat{\mathsf{ct}}$ of ciphertexts, $\widehat{e}_{\mathsf{Sim}} \in \{0,1\}^{2\times 1}$, a matrix $\widehat{\mathsf{tk}}$ of trapdoor keys.

1. Samples $\widehat{\rho}_0, \widehat{\rho}_1 \xleftarrow{u} \mathbb{Z}_p, \widehat{d}_0, \widehat{d}_1 \xleftarrow{u} \mathbb{Z}_p^{t-1}$, where $\widehat{d}_0 = (\widehat{d}_{0,1}, \widehat{d}_{0,2}, \ldots, \widehat{d}_{0,t-1})$, $\widehat{d}_1 = (\widehat{d}_{1,1}, \widehat{d}_{1,2}, \ldots, \widehat{d}_{1,t-1})$.
2. Constructs polynomials f_j for $j \in \{0,1\}$ as $f_j(z) = \widehat{\rho}_j + \widehat{d}_{j,1}z + \ldots + \widehat{d}_{j,t-1}z^{t-1}$.
3. Sets $\widehat{\mathsf{ct}} = \begin{pmatrix} \widehat{\mathsf{ct}}_0 \\ \widehat{\mathsf{ct}}_1 \end{pmatrix}$ where $\widehat{\mathsf{ct}}_0 \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,0); \widehat{\rho}_0)$ and $\widehat{\mathsf{ct}}_1 \leftarrow \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,1); \widehat{\rho}_1)$.
4. Computes $y^{f_j(\mathsf{id}_l)}$ for $j \in \{0,1\}, l \in B$ and sets

$$\widehat{e}_{\mathsf{Sim}} = \begin{pmatrix} \mathsf{HC}\Big(\prod_{l\in B}\Big(y^{f_0(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \\ \mathsf{HC}\Big(\prod_{l\in B}\Big(y^{f_1(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \end{pmatrix}$$

where $\lambda_l = \prod_{\substack{j\in B \\ j\neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient.

5. Sets $\widehat{\mathsf{tk}} = \begin{pmatrix} f_0(\mathsf{id}_{\gamma_1}) & f_0(\mathsf{id}_{\gamma_2}) & & f_0(\mathsf{id}_{\gamma_{t-1}}) \\ f_1(\mathsf{id}_{\gamma_1}) & f_1(\mathsf{id}_{\gamma_2}) & & f_1(\mathsf{id}_{\gamma_{t-1}}) \end{pmatrix}$ for a set $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N], |A| = t-1$.

Figure 4: Algorithm $\mathsf{SSim}(i, \mathsf{pp}, y, A, B) \rightarrow (\widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}}, \widehat{\mathsf{tk}})$

Then combining the cases for $i = 1, 2, \ldots, n$ the following two distributions $\Bigg(\mathsf{pp}, x, \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (1,0); \widehat{\rho}_{0,1}) & \ldots & \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (n,0); \widehat{\rho}_{0,n}) \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (1,1); \widehat{\rho}_{1,1}) & \ldots & \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (n,1); \widehat{\rho}_{1,n}) \end{pmatrix},$

$$\begin{pmatrix} \widehat{e}_1 & \widehat{e}_2 & \ldots & \widehat{e}_n \end{pmatrix}, \begin{pmatrix} \widehat{\mathsf{tk}}_1 \\ \widehat{\mathsf{tk}}_2 \\ \vdots \\ \widehat{\mathsf{tk}}_n \end{pmatrix} \Bigg),$$

$$\Bigg(\mathsf{pp}, x, \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (1,0); \widehat{\rho}_{0,1}) & \ldots & \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (n,0); \widehat{\rho}_{0,n}) \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (1,1); \widehat{\rho}_{1,1}) & \ldots & \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (n,1); \widehat{\rho}_{1,n}) \end{pmatrix},$$

$\left(\widehat{e}_{\mathsf{Sim},1} \quad \widehat{e}_{\mathsf{Sim},2} \quad \cdots \quad \widehat{e}_{\mathsf{Sim},n}\right), \begin{pmatrix} \widehat{\mathsf{tk}}_1 \\ \widehat{\mathsf{tk}}_2 \\ \vdots \\ \widehat{\mathsf{tk}}_n \end{pmatrix}$ are computationally equivalent, where $\widehat{e}_1, \widehat{e}_2, \ldots, \widehat{e}_n$ are outputs of $\mathsf{SReal}(1, \mathsf{pp}, x, A), \mathsf{SReal}(2, \mathsf{pp}, x, A), \ldots, \mathsf{SReal}(n, \mathsf{pp}, x, A)$ and $\widehat{e}_{\mathsf{Sim},1}, \widehat{e}_{\mathsf{Sim},2}, \ldots, \widehat{e}_{\mathsf{Sim},n}$ are outputs of $\mathsf{SSim}(1, \mathsf{pp}, y, A, B), \mathsf{SSim}(2, \mathsf{pp}, y, A, B), \ldots, \mathsf{SSim}(n, \mathsf{pp}, y, A, B)$ respectively and $\widehat{\mathsf{tk}}_1, \widehat{\mathsf{tk}}_2, \ldots, \widehat{\mathsf{tk}}_n$ are respectively either outputs of $\mathsf{SReal}(1, \mathsf{pp}, x, A), \mathsf{SReal}(2, \mathsf{pp}, x, A), \ldots, \mathsf{SReal}(n, \mathsf{pp}, x, A)$ or $\mathsf{SSim}(1, \mathsf{pp}, y, A, B), \mathsf{SSim}(2, \mathsf{pp}, y, A, B), \ldots, \mathsf{SSim}(n, \mathsf{pp}, y, A, B)$. This establishes Claim 4.1.

Proof of Claim 4.2: Note that $\widehat{\mathsf{tk}}$ is independent of pp, x and its generation is similar in both the algorithms SReal and SSim. So, it is sufficient to prove that $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}) \stackrel{c}{\equiv} (\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}})$. First note that as $\mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widehat{\mathsf{ct}}_0) = \mathsf{OWFE.Encaps}_2(\mathsf{pp}, y, (i, 0); \widehat{\rho}_0)$ and $\mathsf{OWFE.Decaps}(\mathsf{pp}, x, \widehat{\mathsf{ct}}_1) = \mathsf{OWFE.Encaps}_2(\mathsf{pp}, y, (i, 1); \widehat{\rho}_1)$ by the correctness of OWFE, we have $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}) \equiv (\mathsf{pp}, x, \widehat{\mathsf{ct}}, e')$ where $\widehat{\mathsf{ct}}$ and $\widehat{e}$ are sampled according to $\mathsf{SReal}(i, \mathsf{pp}, x, A)$ and e' is sampled as:

- if $x_i = 0$, then $e' = \begin{pmatrix} \mathsf{OWFE.Encaps}_2(\mathsf{pp}, y, (i, 0); \widehat{\rho}_0) \\ b \end{pmatrix} = \begin{pmatrix} \mathsf{HC}(y^{\widehat{\rho}_0}) \\ b \end{pmatrix}$ where $b \xleftarrow{u} \{0, 1\}$.
- if $x_i = 1$, then $e' = \begin{pmatrix} b \\ \mathsf{OWFE.Encaps}_2(\mathsf{pp}, y, (i, 1); \widehat{\rho}_1) \end{pmatrix} = \begin{pmatrix} b \\ \mathsf{HC}(y^{\widehat{\rho}_1}) \end{pmatrix}$ where $b \xleftarrow{u} \{0, 1\}$.

Also $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, e') \equiv (\mathsf{pp}, x, \widehat{\mathsf{ct}}, e'')$ where e'' is sampled as:

- if $x_i = 0$, then $e'' = \begin{pmatrix} \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_0(\mathsf{id}_l)}\right)^{\lambda_l}\right) \\ b \end{pmatrix}$ where $b \xleftarrow{u} \{0, 1\}, B \subseteq [N], |B| = t$.
- if $x_i = 1$, then $e'' = \begin{pmatrix} b \\ \mathsf{HC}\left(\prod_{l \in B} \left(y^{f_1(\mathsf{id}_l)}\right)^{\lambda_l}\right) \end{pmatrix}$ where $b \xleftarrow{u} \{0, 1\}, B \subseteq [N], |B| = t$.

Note that $\widehat{\rho}_j = f_j(0) = \sum_{l \in B} \lambda_l f_j(\mathsf{id}_l)$ for $j = 0, 1$ by Lagrange's interpolation where $\lambda_l = \prod_{\substack{j \in B \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's co-efficient. Thus, $\mathsf{HC}\left(\prod_{l \in B} \left(y^{f_j(\mathsf{id}_l)} \right)^{\lambda_l} \right)$ $= \mathsf{HC}\left(y^{\sum_{l \in B} \lambda_l f_j(\mathsf{id}_l)} \right) = \mathsf{HC}(y^{f_j(0)}) = \mathsf{HC}(y^{\widehat{\rho}_j})$ and consequently e'' and e' have the same distribution.
Next we prove Claim 4.3 described below from the security for encryption requirement of the OWFE scheme. This will establish Claim 4.2.

Claim 4.3. $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, e'') \stackrel{c}{\equiv} (\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}})$

We know that security for encryption requirement assures that no probabilistic polynomial time adversary can distinguish between $(\mathsf{pp}, x, \mathsf{ct}_1, e_1)$ and $(\mathsf{pp}, x, \mathsf{ct}_1, e_2)$ where $\mathsf{pp} \leftarrow \mathsf{OWFE.Gen}(1^\lambda)$, $(\mathsf{ct}_1, e_1) \leftarrow \mathsf{OWFE.Encaps}(\mathsf{pp}, y, (i, 1 - x_i); \rho')$ with $e_1 = \mathsf{HC}(y^{\rho'}), y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ and $e_2 \stackrel{u}{\leftarrow} \{0,1\}$. Let us call $(\mathsf{pp}, x, \mathsf{ct}_1, e_1)$ the simulated challenge and $(\mathsf{pp}, x, \mathsf{ct}_1, e_2)$ the random challenge. To give the reduction we present a procedure Turn which turns a simulated challenge $(\mathsf{pp}, x, \mathsf{ct}_1, e_1)$ into a sample $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}})$ of Claim 4.3 and turns a random challenge $(\mathsf{pp}, x, \mathsf{ct}_1, e_2)$ into a sample $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, e'')$ of Claim 4.3.
The algorithm Turn $(\mathsf{pp}, x, \mathsf{ct}, e)$ returns (ct_3, e_3) formed as in Figure 5.

Observe that on input the random challenge $(\mathsf{pp}, x, \mathsf{ct}_1, e_2)$, the algorithm Turn generates ct_3, e_3 as follows:

- if $x_i = 0$ then $\mathsf{ct}_3 = \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i, 0); \rho) \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i, 1); \rho') \end{pmatrix}$,

$e_3 = \left(\mathsf{HC}\left(\prod_{l \in B'} \left(y^{g(\mathsf{id}_l)} \right)^{\lambda_l} \right) \right)_{e_2}$ where $e_2 \stackrel{u}{\leftarrow} \{0,1\}$, $B' \subseteq [N], |B'| = t$, $\lambda_l = \prod_{\substack{j \in B' \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient.

- if $x_i = 1$ then $\mathsf{ct}_3 = \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i, 0); \rho') \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i, 1); \rho) \end{pmatrix}$,

$e_3 = \left(\mathsf{HC}\left(\prod_{l \in B'} \left(y^{g(\mathsf{id}_l)} \right)^{\lambda_l} \right) \right)_{e_2}$ where $e_2 \stackrel{u}{\leftarrow} \{0,1\}$, $B' \subseteq [N], |B'| = t$,

1. Samples $\rho \xleftarrow{u} \mathbb{Z}_p, d = (d_1, d_2, \ldots, d_{t-1}) \xleftarrow{u} \mathbb{Z}_p^{t-1}$.
2. Constructs a polynomial $g(z) = \rho + d_1 z + \ldots + d_{t-1} z^{t-1}$.
3. Computes $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$, chooses $B' \subseteq [N]$ with $|B'| = t$ and generates (ct_3, e_3) as follows:
- if $x_i = 0$ then

$$\mathsf{ct}_3 = \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,0); \rho) \\ \mathsf{ct} \end{pmatrix}$$

$$e_3 = \begin{pmatrix} \mathsf{HC}\Big(\prod_{l \in B'} \Big(y^{g(\mathsf{id}_l)} \Big)^{\lambda_l} \Big) \\ e \end{pmatrix}$$

where $\lambda_l = \prod_{\substack{j \in B' \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient.

- if $x_i = 1$ then

$$\mathsf{ct}_3 = \begin{pmatrix} \mathsf{ct} \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,1); \rho) \end{pmatrix}$$

$$e_3 = \begin{pmatrix} e \\ \mathsf{HC}\Big(\prod_{l \in B'} \Big(y^{g(\mathsf{id}_l)} \Big)^{\lambda_l} \Big) \end{pmatrix}$$

where $\lambda_l = \prod_{\substack{j \in B' \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient.

Figure 5: Algorithm $\mathsf{Turn}(\mathsf{pp}, x, \mathsf{ct}, e) \rightarrow (\mathsf{ct}_3, e_3)$

$\lambda_l = \prod_{\substack{j \in B' \\ j \neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient.

It is easy to see that the distribution of $(\mathsf{pp}, x, \mathsf{ct}_3, e_3)$ is similar to that of $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, e'')$ of Claim 4.3.

Also note that by taking input the simulated challenge $(\mathsf{pp}, x, \mathsf{ct}_1, e_1)$, the algorithm Turn generates ct_3, e_3 in the following way:

- if $x_i = 0$ then $\mathsf{ct}_3 = \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,0); \rho) \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,1); \rho') \end{pmatrix}$,

$$e_3 = \begin{pmatrix} \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \\ e_1 \end{pmatrix} = \begin{pmatrix} \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \\ \mathsf{HC}(y^{\rho'}) \end{pmatrix}$$

$$= \begin{pmatrix} \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \\ \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g'(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \end{pmatrix}$$

where $B' \subseteq [N], |B'| = t$, $\lambda_l = \prod_{\substack{j\in B' \\ j\neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient and $g'(z) = \rho' + d'_1 z + \ldots + d'_{t-1} z^{t-1}, d' = (d'_1, d'_2, \ldots, d'_{t-1}) \leftarrow \mathbb{Z}_p^{t-1}$.

- if $x_i = 1$ then $\mathsf{ct}_3 = \begin{pmatrix} \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,0); \rho') \\ \mathsf{OWFE.Encaps}_1(\mathsf{pp}, (i,1); \rho) \end{pmatrix}$,

$$e_3 = \begin{pmatrix} e_1 \\ \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \end{pmatrix} = \begin{pmatrix} \mathsf{HC}(y^{\rho'}) \\ \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \end{pmatrix}$$

$$= \begin{pmatrix} \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g'(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \\ \mathsf{HC}\Big(\prod_{l\in B'}\Big(y^{g(\mathsf{id}_l)}\Big)^{\lambda_l}\Big) \end{pmatrix}$$

where $B' \subseteq [N], |B'| = t$, $\lambda_l = \prod_{\substack{j\in B' \\ j\neq l}} \frac{-\mathsf{id}_j}{\mathsf{id}_l - \mathsf{id}_j}$ is the Lagrange's coefficient and $g'(z) = \rho' + d'_1 z + \ldots + d'_{t-1} z^{t-1}, d' = (d'_1, d'_2, \ldots, d'_{t-1}) \leftarrow \mathbb{Z}_p^{t-1}$.

It is clear that $(\mathsf{pp}, x, \mathsf{ct}_3, e_3)$ is identically distributed to $(\mathsf{pp}, x, \widehat{\mathsf{ct}}, \widehat{e}_{\mathsf{Sim}})$ of Claim 4.3.

So, Claim 4.3 holds and hence Claim 4.2 follows. Thus Claim 4.1 is true. From Claim 4.1 it directly follows that $\mathsf{view}_0 \stackrel{c}{\equiv} \mathsf{view}_1$ because the view in either hybrid is obtained entirely based on $((\mathsf{CT}_1, E_1, \mathsf{TK}_1), (\mathsf{CT}_2, E_2, \mathsf{TK}_2), \ldots, (\mathsf{CT}_r, E_r, \mathsf{TK}_r))$ and the fact that this tuple is sampled from the distribution $\mathsf{Real}(\mathsf{pp}, x, A)$ in one hybrid and from $\mathsf{Sim}(\mathsf{pp}, y, A, B)$ in the other. □

Lemma 4.2. *Inverting the image* $y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ *under* view_1 *to recover* x *is computationally infeasible .*

Proof. We have to argue that for any adversary $\mathcal{A}$, $\Pr[x' = x] = \mathsf{negl}(\lambda)$ where $x' \leftarrow \mathcal{A}(\mathsf{view}_1)$. We know that $\mathsf{view}_1 = (\mathsf{ik}, Y, \mathsf{tk}_{\mathsf{id}_{\gamma_1}}, \mathsf{tk}_{\mathsf{id}_{\gamma_2}}, \ldots, \mathsf{tk}_{\mathsf{id}_{\gamma_{t-1}}})$ is the view in Hybrid 1 and the variables $\mathsf{pp}, y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$ are part of $\mathsf{ik} = (\mathsf{pp}, (\mathsf{ct}_{1,0}, \mathsf{ct}_{1,1}), (\mathsf{ct}_{2,0}, \mathsf{ct}_{2,1}), \ldots, (\mathsf{ct}_{n,0}, \mathsf{ct}_{n,1}))$ and $Y = (y, (\sigma_{1,0}, \sigma_{1,1}), \ldots, (\sigma_{n,0}, \sigma_{n,1}))$. The proof of Lemma 4.2 follows directly from the one-wayness of OWFE.func and perfect privacy of Shamir's secret sharing scheme. Also observe that view_1 is entirely based on $\mathsf{pp}, y \leftarrow \mathsf{OWFE.func}(\mathsf{pp}, x)$, $A = \{\gamma_1, \gamma_2, \ldots, \gamma_{t-1}\} \subseteq [N]$, $B \subseteq [N]$ with $|B| = t$. This is because in view_1 the underlying variables $(\mathsf{CT}_j, E_{\mathsf{Sim},j}, \mathsf{TK}_j)$ for all $j \in [r]$ are produced as $(\mathsf{CT}_j, \mathsf{E}_{\mathsf{Sim},j}, \mathsf{TK}_j) \leftarrow \mathsf{Sim}(\mathsf{pp}, y, A, B)$ without the knowledge of x. □

REFERENCES

Boyen, X. and Waters, B. (2010). Shrinking the keys of discrete-log-type lossy trapdoor functions. In *International Conference on Applied Cryptography and Network Security*, pages 35–52. Springer.

De Santis, A., Desmedt, Y., Frankel, Y., and Yung, M. (1994). How to share a function securely. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 522–533.

Desmedt, Y. and Frankel, Y. (1989). Threshold cryptosystems. In *Conference on the Theory and Application of Cryptology*, pages 307–315. Springer.

Diffie, W. and Hellman, M. (1976). New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654.

Döttling, N. and Garg, S. (2017). Identity-based encryption from the diffie-hellman assumption. In *Annual International Cryptology Conference*, pages 537–569. Springer.

Frankel, Y. (1989). A practical protocol for large group oriented networks. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 56–61. Springer.

Freeman, D. M., Goldreich, O., Kiltz, E., Rosen, A., and Segev, G. (2010). More constructions of lossy and correlation-secure trapdoor functions. In *International Workshop on Public Key Cryptography*, pages 279–295. Springer.

Garg, S. and Hajiabadi, M. (2018). Trapdoor functions from the computational diffie-hellman assumption. In *Annual International Cryptology Conference*, pages 362–391. Springer.

Peikert, C. and Waters, B. (2011). Lossy trapdoor functions and their applications. *SIAM Journal on Computing*, 40(6):1803–1844.

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.

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, 22(11):612–613.

Tu, B., Chen, Y., and Wang, X. (2019). Threshold trapdoor functions and their applications. *IET Information Security*, 14(2):220–231.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

SPA on Modular Multiplication in Rabin-p KEM

Amir Hamzah Abd Ghafar[*1], **Muhammad Rezal Kamel Ariffin**[1,2], **Hazlin Abdul Rani**[3], and **Siti Hasana Sapar**[1,2]

[1]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, Malaysia*
[2]*Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia*
[3]*Cryptography Development Department, CyberSecurity Malaysia, Malaysia*

E-mail: amir_hamzah@upm.edu.my
[*]*Corresponding author*

ABSTRACT

Rabin-p key encapsulation mechanism (KEM) is a variant of the Rabin encryption scheme, famously utilizing square root modular problem as its security strength. The Rabin-p KEM algorithm has been selected as one of the candidates by Malaysian's MySEAL project that aims to select new cryptographic algorithms. With suitable devices, the side-channel attack is a powerful attack that collects secret information via the physical data leaked out from a cryptographic device. It can expose the private key of a cryptosystem by targeting the cryptographic implementation of an algorithm. Since Rabin-p is a new key encapsulation mechanism, no previous side-channel attack has been known to cause its implementation to be vulnerable. Thus, this paper shows that the side-channel attack using simple power analysis on Rabin-p KEM results in its private key p to be known in feasible time. Also, a variation of this method has been shown to be effective against a single modular multiplication operation. Finally, this paper also suggests a randomized approach for future implementation of Rabin-p to prevent this kind of attack.

Keywords: cryptanalysis, side-channel analysis, simple power analysis, key encapsulation mechanism, Rabin-p KEM, MySEAL

1 INTRODUCTION

It is common for a nation to determine the most suitable set of cryptographic algorithms that meet specifically tailored specifications to ensure its digital sovereignty. Floridi (2020) asserted that protecting digital sovereignty among the European countries has been made possible through the endorsement of General Data Protection Regulation by European Unions (EU). Hence, it is natural for some countries to choose *non-US standard* cryptosystems as long as it is not proprietary; in order to be deployed within government agencies and the scientific community conducts sufficient cryptanalyses upon the algorithms.

Several past cryptanalyses exercises can be associated with these attempts, traced back to works by Matsumoto (2002), Menezes (2002) on HIME(R), a Japanese encryption algorithm. In addition, an attack by Shan et al. (2019) against the signing algorithm of the Chinese national ID-based algorithm, SM9 using horizontal side-channel analysis is another instance of this exercise. Other attacks on the same targeted algorithm were conducted by Cheng (2018), Zhang et al. (2018). Furthermore, Babenko and Ishchukova (2010) also showed a differential analysis on Russia's GOST 28147-89 encryption algorithm. More attacks on GOST can be read in Babenko et al. (2014, 2012, 2013). All these works emphasise the importance of analysing non-US standard cryptographic algorithms, especially if they are to be considered as a national standard in securing digital sovereignty.

In parallel to the above initiatives, MySEAL, which denotes for *Senarai Algoritma Kriptografi Terpercaya Negara* (the Malay translation for National Trusted Cryptographic Algorithms) is Malaysia's effort to develop requirements and guidelines on the usage of cryptographic algorithms in all trusted cryptographic products in Malaysia (CyberSecurity, 2019). The project is divided into two initiatives:

- **AKBA** (denotes for *Algoritma Kriptografi Baharu* or New Cryptographic Algorithms) **MySEAL.** The call for new cryptographic algorithms from local and international researchers; and
- **AKSA** (denotes for *Algoritma Kriptografi Sedia Ada* or Existing Cryptographic Algorithms) **MySEAL.** To study and determine security "levels" of existing cryptographic algorithms for usage in trusted cryptographic devices in Malaysia.

In the AKBA initiative, two cryptographic algorithms have been short-listed, namely Rabin-p Key Encapsulation Mechanism (KEM) (Asbullah et al., 2019) and TNC digital signature scheme (Yuan et al., 2019). Since the announcement of the AKBA listings, a call for cryptanalysis on both algorithms has been made.

The side-channel method is considered as a practical cryptanalytic strategy in enhancing the security of future implementation of any recently published algorithm. Introduced by Kocher (1996), the attack works by collecting the side-effect of cryptographic computations. These side-effects include its computational power Kocher et al. (2011), Messerges et al. (1999), computational time Kocher (1996), electromagnetic effect Yilmaz et al. (2019), acoustic Genkin et al. (2014) and other physical effects which are gained from the cryptographic devices. The impact of this method is significant in the last five years where exploits such as Spectre Kocher et al. (2019), and Meltdown Abu-Ghazaleh et al. (2019) has forced CPU-producing companies to redesign their products that can hinder both vulnerabilities. These incidents show that addressing side-channel vulnerabilities must be taken into consideration before implementing a cryptosystem such as Rabin-p.

Our contributions. This paper provides cryptanalysis on Rabin-p KEM. Specifically, we conduct a simple power analysis upon side-channel attacks on the cryptosystem and discover that the future implementation of Rabin-p KEM can be vulnerable to it. We also suggest improving the future implementation of Rabin-p KEM so that the attack that we discussed can be avoided.

Organization of the article. In Section 2, we introduce simple power analysis, which is the methodology of our attack. Then, in Section 3, we discuss the background and security of Rabin-p KEM. Section 4 provides the

main result of this paper where we attack Rabin-p KEM using simple power analysis. Next, in Section 5, we discuss a method called Rabin-p blinding to prevent the attack, before we conclude this paper in Section 6.

2 RELATED WORKS

Power analysis focuses on the computational power, which observes the power traces out of devices. Simple power analysis (SPA) is defined as the simplified version of power analysis in which direct observation can yield information regarding the key used in the cryptographic algorithm. From the information, an adversary may derive and reformulate the structures - especially the bits - of the private key by using different inputs of the algorithm. For example, using two different inputs, Y_1 and Y_2 into a device that computes RSA decryption algorithm, Kocher et al. (1999) showed there would be two distinct patterns of power traces, Δ_{Y_1} and Δ_{Y_2} where one input produced an output less than an RSA prime while another input produced an output greater than the same RSA prime. By choosing i-many different inputs, Y_i an adversary can eventually derive the position of the RSA primes (which is its private key), leading to the insecurity of the cryptosystem. To counteract SPA in RSA, Coron (1999) introduced a square-and-multiply-always approach during the modular exponentiation process in the RSA decryption algorithm. However, Yen et al. (2005) showed that this approach would succumb from a different type of side-channel attack called fault analysis. Due to this result, Giraud (2006) proposed a novel method to compute modular exponentiation that can resist both SPA and fault analysis.

The utilization of RSA for a source of cryptographic security is still a popular choice. This is due to its commutative structure. Deployments of embedded devices using RSA enables devices with both encryption and digital signing functionalities. As such, continuous assessment on RSA via SPA is of utmost importance. Devices adopting modular exponentiation algorithms via Barret and Montgomery reductions Courrege et al. (2010) has to be carefully scrutinized. This is due to the fact that uncareful deployment of modular exponentiation algorithms will lead to leakage of information through side channel attacks. More recent works of SPA can also be read in Luo et al. (2018), Ran-

dolph and Diehl (2020), Won et al. (2020). This chain of results showed that in the past SPA focused on modular exponentiations in RSA since the operation is a necessary computation in its decryption algorithm. However, in 2015, Ghafar and Ariffin (2016) showed that SPA could also be conducted on computation involving modular multiplication. The attack was conducted upon a variant of the original Rabin cryptosystem known as the AA_β cryptosystem (Ariffin et al., 2013). The cryptanalysis successfully factored AA_β's public key, $N = p^2q$ in polynomial time. The attack is significant since this is the first time SPA was conducted on modular multiplication instead of modular exponentiation.

3 RABIN-P KEM

The key encapsulation mechanism (KEM) is a mechanism to transport the key of a symmetric encryption scheme securely to establish secure communication between two entities. Using identical hard problem of integer factorization problem with Rabin Rabin (1979) and RSA Rivest et al. (1978) cryptosystems, Rabin-p KEM distinguishes itself by adopting construction proposed by Dent (2003) in building a KEM that satisfies One-Wayness against Chosen-Plaintext Attacks (OW-CPA). Furthermore, in the original submission of Rabin-p, Asbullah et al. (2019) proved that the KEM is IND-CPA secure in the random oracle model. Chin and Mohamad (2020) later provided the formal security proof that Rabin-p KEM also satisfies the IND-CCA2 security proof. The key generation algorithm of Rabin-p KEM is as follows.

Algorithm 1 Rabin-p KEM Key Generation Algorithm

Input: Security parameter, k.
Output: Public key N and private key p.

1: Generate two random and distinct primes p and q such that $p, q \equiv 3 \pmod 4$ where $2^k < p, q < 2^{k+1}$.
2: Compute $N = p^2q$.
3: Output public key N and private key p.

From Algorithm 1, we can see the integer factorization problem is embedded in the formulation of N. Typically, N with 3072-bit will have 128-bit

security level as shown in the original report. The encapsulation algorithm is presented as follows.

Algorithm 2 Rabin-p KEM Encapsulation Algorithm

Input: Public key, N
Output: A ciphertext tuple (K, C)

1: Choose a randomly integer $2^{\frac{3k}{2}} < x < 2^{2k-1}$.
2: Compute $c_1 \equiv x^2 \pmod{N}$.
3: Compute $c_2 = H(x)$. ▷ $H(x)$ is a hash function.
4: Set $C := (c_1, c_2)$.
5: Set $K := k(x)$. ▷ $k(x)$ is a key derivative function.
6: Output (K, C).

Rabin-p KEM generates the symmetric key, K, by using a key derivation function with the plaintext, x as its input. In order to ensure identical x is used, the sender generates the fingerprint of x using hash function, $H(x)$ (such as SHA-2) and sends it as parts of the ciphertext for the recipient to check during the decapsulation process. The decapsulation algorithm is shown in Algorithm 3.

Some designs in Algorithm 3 were originated from AA_β cryptosystem (Ariffin et al., 2013). The computation of only single modular exponentiation $x_p \equiv w^{(p+1)/4} \pmod{4}$ avoids the problem of distinguishing the correct plaintext that occurs in the original Rabin decryption algorithm. Rabin-p KEM defines two hard problems to be its security strengths. The problems are called the Rabin-p decryption problem and the Rabin-p private key problem. In Asbullah et al. (2019), both problems have been reduced to integer factorization problem (IFP). Specifically, given Rabin-p public key, $N = p^2q$, an adversary needs to factor N and find the values of p and q which are two random and distinct primes such that $p, q \equiv 3 \pmod{4}$ where $2^k < p, q < 2^{k+1}$. In the next section, we show that our attack using simple power analysis can solve IFP in a feasible time.

Algorithm 3 Rabin-p KEM Decapsulation Algorithm

Input: A ciphertext C and private key p.
Output: The value K.

1: Parse C as (c_1, c_2).
2: Compute $w \equiv c_1 \pmod{p}$.
3: Compute $x_p \equiv w^{\frac{p+1}{4}} \pmod{4}$.
4: Compute $i = \frac{c_1 - x_p^2}{p}$.
5: Compute $j \equiv \frac{i}{2x_p} \pmod{p}$.
6: $x_1 = x_p + jp$.
7: **if** $x_1 < 2^{2k-1}$ **then**
8: $\quad x = x_1$.
9: **else** $x = p^2 - x_1$.
10: **end if**
11: Compute $H(x)$ ▷ $H(x)$ is a hash function.
12: **if** $c_2 = H(x)$ **then**
13: $\quad$ Set $K := k(x)$ ▷ $k(x)$ is a key derivative function.
14: $\quad$ Output K.
15: **else** Output $\perp$ and halt.
16: **end if**

4 SPA ON RABIN-P KEM

In this cryptanalysis, we focus on the decapsulation algorithm of Rabin-p KEM. Particularly step 2 in Algorithm 3:

$$w \equiv c_1 \pmod{p}. \tag{1}$$

Our attack can be categorized under the chosen-ciphertext attack model. In this model, an adversary can modify the ciphertext that is the decryption algorithm's inputs and collect the algorithm's outputs. We also require the following assumptions in the attack:

Assumption 4.1. *The adversary is also able to monitor the power consumption used by the decryption algorithm.*

This assumption is realistic since leakages produced from side-channel attacks (described in Section 2) of most of the electronic devices can be monitored.

Assumption 4.2. *Given the power traces of the power consumptions of the targeted device, the adversary can distinguish correctly the region of power traces that execute modular reduction operation in (1).*

This assumption depends on the knowledge and experiences of the adversary to perform power analysis on that particular Rabin-p decapsulation device.

4.1 The Attack

By observing the power consumption of (1), the adversary needs to choose certain values to be the ciphertext. In a typical computation of Rabin-p, the value of c_1 will be greater than p since

$$c_1 = m^2 - \kappa_1 p^2 q$$

for some $\kappa_1 \in \mathbb{Z}$ where $2^{3k/2} < m < 2^{2k-1}$. Our hypothesis is the power traces from the computation of (1) will be different between $c_1 < p$ and $c_1 > p$. To outline this observation, we define several useful notations:

Definition 4.1. *Let p be the private key of Rabin-p KEM. Let c_1 be the ciphertext of the decapsulation algorithm of Rabin-p KEM. We define $\tau(c_1)$ as the target equation of computation in (1). $Reg(\tau_{c_1})$ is the region of power traces that executes $\tau(c_1)$.*

Definition 4.2. *Let $\Delta_{c_1}(\tau)$ be the computational power of $\tau(c_1)$ under $Reg(\tau)$. Its power traces can be observed by an adversary using any power observational devices (e.g. standard digital oscilloscope). For simplicity, we denote it as Δ_{c_1}. We also denote Δ_γ if we are only interested of discussing computational power of arbitrary function upon arbitrary input, γ.*

Next, we define our methodology to compare a pair of power traces and their consequences.

Definition 4.3. *An ϵ-power trace difference is the non-significant difference of two arbitrary power traces that execute τ with different inputs, namely Δ_{γ_1} and Δ_{γ_2}. That is*

$$|\Delta_{\gamma_1} - \Delta_{\gamma_2}| < \epsilon$$

as depicted in Figure 1.

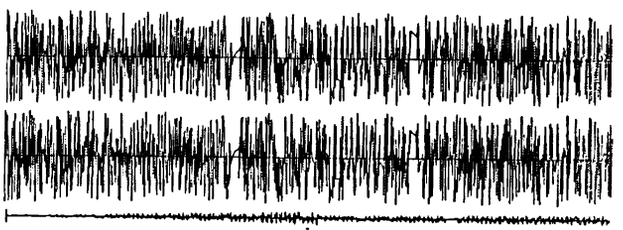


Figure 1: The top trace shows the power traces for Δ_{γ_1} while the middle trace show the power traces for Δ_{γ_2}. The bottom trace shows the difference between the both upper traces, $|\Delta_{\gamma_1} - \Delta_{\gamma_2}| < \epsilon$.

Definition 4.4. *A non ϵ-power trace difference is the significant difference of two power traces that execute the same operation with different inputs, namely Δ_{γ_1} and Δ_{γ_2}. That is*

$$|\Delta_{\gamma_1} - \Delta_{\gamma_2}| > \epsilon$$

as depicted in Figure 2.

Now, we can present our result in the next theorem:

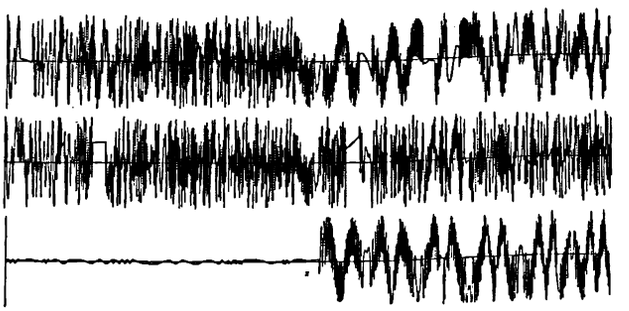


Figure 2: The top trace shows the power traces for Δ_{γ_1} while the middle trace show the power traces for Δ_{γ_2}. The bottom trace shows the difference between the both upper traces. From left to right it begins with $|\Delta_{\gamma_1} - \Delta_{\gamma_2}| < \epsilon$ and ends with $|\Delta_{\gamma_1} - \Delta_{\gamma_2}| > \epsilon$

Theorem 4.1. *Let c'_1 and c'_2 be the chosen ciphertexts in Rabin-p KEM decapsulation algorithm such that $c'_1 < c'_2$. Suppose $\Delta_{c'_1}$ and $\Delta_{c'_2}$ are the power traces from the computation of $\tau_{c'_1}$ and $\tau_{c'_2}$. If and only if*

(a) $c'_1 < c'_2 \leq p$ or $p < c'_1 < c'_2$ then $|\Delta_{c'_2} - \Delta_{c'_1}| < \epsilon$; and

(b) $c'_1 \leq p < c'_2$ then $|\Delta_{c'_2} - \Delta_{c'_1}| > \epsilon$.

for $\epsilon > 0$.

Proof. We will show that by choosing two ciphertexts, c'_1 and c'_2 such that $c'_1 < c'_2$, an adversary can predict the partial value of p using the power traces from the computation of (1) using c'_1 and c'_2 as their inputs.
(a) $\Rightarrow$:
If $c'_1 < c'_2 \leq p$, then the computations of

$$w'_1 \equiv c'_1 \pmod{p} \qquad \text{and} \qquad w'_2 \equiv c'_2 \pmod{p}$$

do not involve any modular reduction of p for both c'_1 and c'_2. In fact, $w'_1 = c'_1$ and $w'_2 = c'_2$. While if $p < c'_1 < c'_2$, then the computations of

$$w'_1 \equiv c'_1 \pmod{p} \qquad \text{and} \qquad w'_2 \equiv c'_2 \pmod{p}$$

involve modular reductions of p for both c_1' and c_2'. Since both $\tau(c_1')$ and $\tau(c_2')$ are going through the similar operation, their $\Delta_{c_1'}$ and $\Delta_{c_2'}$ will also be similar. Hence $|\Delta_{c_2'} - \Delta_{c_1'}| < \epsilon$.

$\Leftarrow$:

When an adversary obtains power traces such that

$$|\Delta_{c_2'} - \Delta_{c_1'}| < \epsilon$$

This shows that executing $\tau_{c_1'}$ and $\tau_{c_2'}$ utilize same amount of computational power. This is due to both are executing the same operations which:

(i) do not involve modular reduction process that implies $c_1' < c_2' \leq p$ or;

(ii) involve modular reduction process that implies $p < c_1' < c_2'$.

(b) $\Rightarrow$:

If $c_1' \leq p < c_2'$, the computation of

$$w_1' \equiv c_1' \pmod{p}$$

does not involve any modular reduction of p while

$$w_2' \equiv c_2' \pmod{p}$$

have a modular reduction of p. In fact, $w_1' = c_1'$ while $w_2 = c_2' - \kappa p$ for some $\kappa \in \mathbb{Z}$. Both $\tau(c_1')$ and $\tau(c_2')$ are going through the different operations, therefore their $\Delta_{c_1'}$ and $\Delta_{c_2'}$ will not be similar. Hence $|\Delta_{c_2'} - \Delta_{c_1'}| > \epsilon$.

$\Leftarrow$:

When an adversary obtains power traces such that

$$|\Delta_{c_2'} - \Delta_{c_1'}| > \epsilon$$

This shows that executing $\tau_{c_1'}$ and $\tau_{c_2'}$ utilize different amount of computational power. This implies $\tau_{c_2'}$ has an additional operation of modular reduction while $\tau_{c_1'}$ does not execute any modular reduction of p. This implies $c_1' \leq p < c_2'$. □

Remark 4.1. *$|\Delta_{c_2'} - \Delta_{c_1'}| < \epsilon$ shows similar power traces shown in Figure 1 and $|\Delta_{c_2'} - \Delta_{c_1'}| > \epsilon$ shows the similar power traces shown in Figure 2*

Result from Theorem 4.1 allows adversary to construct a method to determine the bits of p by analyzing the power traces of c_1' and c_2'. Before the method is introduced, we define a notation to indicate the position of the bits:

Definition 4.5. *Let b be a k-bit integer. We define $b[t]$ to indicate the $t-th$ bit (from the left) of b for $t = \{1, 2, \ldots, k-1, k\}$. Specifically, $b[1]$ indicates the most significant bit (MSB) of b while $a[k]$ indicates the least significant bit (LSB) of b.*

To determine the full bits of p using SPA, we assume that the adversary knows $t-1$ MSBs of p. The adversary also sets $t-1$ MSBs of c'_1 and c'_2 to follow $t-1$ MSBs of p.

Theorem 4.2. *Let p be the private key of Rabin-p cryptosystem. Let c'_1 and c'_2 be the ciphertexts in Rabin-p decryption algorithm such that $c'_1 < c'_2$. Suppose p, c'_2, c'_2 are k-bits in sizes. Suppose $c'_1[1] = c'_2[1] = p[1], c'_1[2] = c'_2[2] = p[2], \ldots, c'_1[t-2] = c'_2[t-2] = p[t-2], c'_1[t-1] = c'_2[t-1] = p[t-1]$ are known while $c'_1[t+1], c'_1[t+2], \ldots, c'_1[k]$ and $c'_2[t+1], c'_2[t+2], \ldots, c'_2[k]$ are set to 0 and $p[t+1], p[t+2], \ldots, p[k]$ are unknown. Set $c'_1[t] = 0$ and $c'_2[t] = 1$. Suppose $\Delta_{c'_1}$ and $\Delta_{c'_2}$ are the power traces from the computation of $\tau_{c'_1}$ and $\tau_{c'_2}$. If*

(a) $|\Delta_{c'_2} - \Delta_{c'_1}| < \epsilon$ *then* $p[t] = 1$; *and*

(b) $|\Delta_{c'_2} - \Delta_{c'_1}| > \epsilon$ *then* $p[t] = 0$.

Proof. We will show that by observing the differences of the power traces of $\tau_{c'_1}$ and $\tau_{c'_2}$, an adversary can determine the t-th bit of p.
(a) From Theorem 4.1(a), we know that if $|\Delta_{c'_2} - \Delta_{c'_1}| < \epsilon$ then there are two possibilities which are $c'_1 < c'_2 \leq p$ or $p < c'_1 < c'_2$. However, the possibility of $p < c'_1 < c'_2$ to occur is unlikely since the remaining bits of c'_1 which are $c'_1[t+1], c'_1[t+2], \ldots, c'_1[k]$ and the remaining bits of c'_2 which are $c'_2[t+1], c'_2[t+2], \ldots, c'_2[k]$ are set to 0 while $p[k] = 1$ (since p is an odd prime number). Therefore, $c'_1 < c'_2 \leq p$. This means $p[t] = c'_2[t] = 1$.
(b) From Theorem 4.1(b), we know that if $|\Delta_{c'_2} - \Delta_{c'_1}| > \epsilon$ then $c'_1 \leq p < c'_2$. Since $p < c'_2$, it is unlikely that $p[t] = c'_2[t] = 1$ when $c'_2[t+1], c'_2[t+2], \ldots, c'_2[k]$ are 0. On the other hand, $c'_1[t+1], c'_1[t+2], \ldots, c'_1[k]$ are 0 and $p[k] = 1$ (since p is an odd prime number). Thus, $p[t]$ must be equal to $c'_1[t]$ to satisfy $c'_1 \leq p < c'_2$. Therefore, $p[t] = c'_1[t] = 0$. □

Theorem 4.2 shows that by observing the differences of the power traces of $\tau_{c_1'}$ and $\tau_{c_2'}$, the adversary can determine the t-th bit of p. The adversary then can reiterate the processes to obtain the next MSBs of p.

Remark 4.2. *Not all the bits of p must be known for the adversary to obtain the value of p. An attack called partial key exposure attack shows that partial MSBs of a prime factor are sufficient to factor an RSA modulus using method from Coppersmith (1996). This approach could be replicated on Rabin-p using partial MSBs of p to factor N.*

An algorithm to execute SPA based on the previous theorems is as follows:

Algorithm 4 SPA on Rabin-p Algorithm

Input: Chosen ciphertexts, c_1' and c_2' which satisfy Theorem 4.2.
Output: t-th bit of private key of Rabin-p cryptosystem, $p[t]$.

1: Set $c_1'[t] = 0$ and $c_2'[t] = 1$
2: Use c_1' as input of Rabin-p decryption algorithm.
3: Collect its computational power traces, $\Delta_{c_1'}$ of computing $\tau_{c_1'}$.
4: Use c_2' as input of Rabin-p decryption algorithm.
5: Collect its computational power traces, $\Delta_{c_2'}$ of computing $\tau_{c_2'}$.
6: Observe the differences of $|\Delta_{c_2'} - \Delta_{c_1'}|$.
7: **if** $|\Delta_{c_2'} - \Delta_{c_1'}| < \epsilon$ **then**
8: $\quad p[t] = 1$
9: **else**
10: $\quad p[t] = 0$
11: **end if**
12: Output $p[t]$.

Example 4.1. *We considered the toy example from the original proposal of Rabin-p (Asbullah et al., 2019). Private key, p is a k-bit prime and $k = 15$, thus we need to determine the remaining 14 bits of p (its 15th bits is 1). Since our attack is categorized under the chosen-ciphertext model, the adversary can modify ciphertexts c_1' and c_2 as required. For $t = 14$, the adversary chooses $c_1'[14] = 0$ and $c_2[14] = 1$ while the remaining bits are set to 0. Both c_1' and c_2' are set as the inputs of Rabin-p decryption oracle and their corresponding power traces, $\Delta_{c_1'}$ in computing $\tau_{c_1'}$ and $\Delta_{c_2'}$ in computing $\tau_{c_2'}$, are collected.*

If

$$|\Delta_{c_2'} - \Delta_{c_1'}| < \epsilon \quad \textit{then 14-th bits of } p, \quad p[14] = 1, \textit{or}$$
$$|\Delta_{c_2'} - \Delta_{c_1'}| > \epsilon \quad \textit{then 14-th bits of } p, \quad p[14] = 0.$$

We reiterate the steps to determine all the remaining bits of p as shown in Table 1:

t-th bit	$\lvert\Delta_{c_2'} - \Delta_{c_1'}\rvert$	$p[t]$
15	-	1
14	$> \epsilon$	0
13	$> \epsilon$	0
12	$> \epsilon$	0
11	$> \epsilon$	0
10	$> \epsilon$	0
9	$> \epsilon$	0
8	$> \epsilon$	0
7	$> \epsilon$	0
6	$> \epsilon$	0
5	$> \epsilon$	0
4	$< \epsilon$	1
3	$> \epsilon$	0
2	$< \epsilon$	1
1	$< \epsilon$	1

Table 1: Finding the full bits of p in Rabin-p.

Finally, we recovered $p = (1000000000001011)_2 = 32779$.

Remark 4.3. *Based on the previous example, an adversary can find k-bits of p with k-iterations of Algorithm 4. Since Asbullah et al. (2019) suggested the size of p to be about $k = 1024$-bits, then Algorithm 4 supposedly to be iterated at most 1024 times. However, depending on the skills and experiences of the adversary to collect the computational power traces, $\Delta_{c_1'}$ and computational power traces, $\Delta_{c_2'}$, with the combination of partial key exposure method mentioned in Remark 4.2, the number of the iterations can be very much less. Thus, our attack is feasible to be mounted in polynomial time.*

Remark 4.4. *Since the value of p is feasible to be computed in polynomial time, the value of q can be obtained by computing $q = \frac{N}{p^2}$. Thus, the integer factorization problem represented in $N = p^2q$ also be solved in polynomial time.*

5 IMPROVEMENTS ON RABIN-P KEM

5.1 Rabin-p blinding

We propose a method to prevent the SPA attack shown in Section 4. We call this method **Rabin-p blinding**. Similar to RSA blinding (Kocher, 1996), the blinding method aims to introduce a random element into the targeted operation that makes the cryptosystem vulnerable to side-channel attack. In Rabin-p case, the output from the computation of (1) needs to be randomized.

Let p and c_1 be the private key and ciphertext respectively described in Algorithm 3. Let

$$w \equiv c_1 \pmod{p} \tag{2}$$

and

$$x_p \equiv w^{\frac{p+1}{4}} \pmod{p} \tag{3}$$

be the Steps 2 and 3 of the same algorithm. To accomplish Rabin-p blinding, we require r_1 and r_2 which are integers larger than p such that

$$r_1 + r_2 \equiv 1 \pmod{p}.$$

Then, we modify (2) to compute

$$w_1 \equiv c_1 \cdot r_1 \pmod{p} \tag{4}$$

and

$$w_2 \equiv c_1 \cdot r_2 \pmod{p}. \tag{5}$$

Based on the values from (4) and (5), observe that (3) will become

$$\begin{aligned} x_p &\equiv (w_1 + w_2)^{\frac{p+1}{4}} \pmod{p} \qquad (6) \\ &\equiv ((c_1 \cdot r_1 - \kappa_1 p) + (c_1 \cdot r_2 - \kappa_2 p))^{\frac{p+1}{4}} - \kappa_3 p \\ &\equiv ((c_1(r_1 + r_2) - \kappa_1 p - \kappa_2 p))^{\frac{p+1}{4}} - \kappa_3 p \\ &\equiv ((c_1(r_1 + r_2) \pmod{p}))^{\frac{p+1}{4}} - \kappa_3 p \qquad (7) \end{aligned}$$

for some $\kappa_1, \kappa_2, \kappa_3 \in \mathbb{Z}$. Since $r_1 + r_2 \equiv 1 \pmod{p}$, (7) become

$$\begin{aligned} x_p &= ((c_1 \pmod{p}))^{\frac{p+1}{4}} - \kappa_3 p \\ &\equiv w^{\frac{p+1}{4}} \pmod{p} \end{aligned}$$

which is equal to the Step 3 of original Rabin-p decapsulation algorithm. By introducing new computations of (4)-(6), we have randomized the computation that previously vulnerable to SPA. Since the new computations are only elementary computations, it can easily be adopted into the future implementation of Rabin-p KEM.

Remark 5.1. *Another method to avoid SPA in* $w \equiv c_1 \pmod{p}$ *is simply by omitting the computation. However, to ensure the decryption works, Step 4 in Algorithm 3 has to be modified to:*

1. $x_p \equiv c_1^{\frac{p+1}{4}} \pmod{4}$ *which will not be equal to* $w^{\frac{p+1}{4}} \pmod{4}$ *in the original algorithm; or*
2. $x_p \equiv (c_1 \pmod{p})^{\frac{p+1}{4}} \pmod{4}$ *which can be reduced to the same targeted equation of* $c_1 \pmod{p}$.

6 CONCLUSION

We have shown that a simple power analysis on Rabin-p can leads to the leakage of its private key, p. This is caused by the differences in power traces that computes the modular reduction step in Rabin-p decapsulation algorithm. Due to the result, we suggest that an improvement using a blinding method that randomize the parameter in the decapsulation algorithm should be added into the future implementation of Rabin-p KEM.

REFERENCES

Abu-Ghazaleh, N., Ponomarev, D., and Evtyushkin, D. (2019). How the Spectre and Meltdown hacks really worked. *IEEE Spectrum*, 56(3):42–49.

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*, 7:19–37.

Asbullah, M. A., Ariffin, M. R. K., and Mahad, Z. (2019). Rabin-p Key Encapsulation Mechanism A Proposal for Public Key Encryption for CyberSecurity Malaysia MySEAL Initiative. Technical report. `https://myseal.cybersecurity.my/en/files/Proposal%20AKBA%20MySEAL%20Rabin-p%20KEM_updated.pdf` (Accessed November 13th, 2019).

Babenko, L. et al. (2014). Algebraic cryptanalysis of GOST encryption algorithm. *Journal of Computer and Communications*, 2(04):10.

Babenko, L. and Ishchukova, E. (2010). Differential analysis of GOST encryption algorithm. In *Proceedings of the 3rd international conference on Security of information and networks*, pages 149–157.

Babenko, L., Ishchukova, E., and Maro, E. (2012). Research about strength of GOST 28147-89 encryption algorithm. In *Proceedings of the Fifth International Conference on Security of Information and Networks*, pages 138–142.

Babenko, L., Ishchukova, E., and Maro, E. (2013). GOST encryption algorithm and approaches to its analysis. In *Theory and Practice of Cryptography Solutions for Secure Information Systems*, pages 34–61. IGI Global.

Cheng, Z. (2018). Security analysis of SM9 key agreement and encryption. In *International Conference on Information Security and Cryptology*, pages 3–25. Springer.

Chin, J.-J. and Mohamad, M. S. (2020). Security Analysis of the Key Encapsulation Mechanism based on Rabin-p Encryption Scheme. *International Journal of Cryptology Research*, 10(2):23–37.

Coppersmith, D. (1996). Finding a small root of a bivariate integer equation; factoring with high bits known. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 178–189. Springer.

Coron, J.-S. (1999). Resistance against differential power analysis for elliptic curve cryptosystems. In *International workshop on cryptographic hardware and embedded systems*, pages 292–302. Springer.

Courrege, J.-C., Feix, B., and Roussellet, M. (2010). Simple power analysis on exponentiation revisited. In *International Conference on Smart Card Research and Advanced Applications*, pages 65–79. Springer.

CyberSecurity (2019). MySEAL - Senarai Algoritma Kriptografi Terpercaya Negara. `https://myseal.cybersecurity.my/en/index.html` (Accessed November 13th, 2019).

Dent, A. W. (2003). A designer's guide to KEMs. In *IMA International Conference on Cryptography and Coding*, pages 133–151. Springer.

Floridi, L. (2020). The fight for digital sovereignty: What it is, and why it matters, especially for the eu. *Philosophy & Technology*, 33(3):369–378.

Genkin, D., Shamir, A., and Tromer, E. (2014). Rsa key extraction via low-bandwidth acoustic cryptanalysis. In *Annual Cryptology Conference*, pages 444–461. Springer.

Ghafar, A. H. A. and Ariffin, M. R. K. (2016). SPA on Rabin variant with public key $N = p^2q$. *Journal of Cryptographic Engineering*, 6(4):339–346.

Giraud, C. (2006). An RSA implementation resistant to fault attacks and to simple power analysis. *IEEE Transactions on computers*, 55(9):1116–1120.

Kocher, P., Horn, J., Fogh, A., Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., Prescher, T., et al. (2019). Spectre attacks: Exploiting speculative execution. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 1–19. IEEE.

Kocher, P., Jaffe, J., and Jun, B. (1999). Differential power analysis. In *Annual International Cryptology Conference*, pages 388–397. Springer.

Kocher, P., Jaffe, J., Jun, B., and Rohatgi, P. (2011). Introduction to differential power analysis. *Journal of Cryptographic Engineering*, 1(1):5–27.

Kocher, P. C. (1996). Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Annual International Cryptology Conference*, pages 104–113. Springer.

Luo, C., Fei, Y., and Kaeli, D. (2018). Effective simple-power analysis attacks of elliptic curve cryptography on embedded systems. In *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–7. IEEE.

Matsumoto, T. (2002). Report on FY2001 Evaluation of Public-Key Cryptographic Techniques.

Menezes, A. (2002). Evaluation of Security Level of Cryptography: The HIME (R) Encryption Scheme.

Messerges, T. S., Dabbish, E. A., and Sloan, R. H. (1999). Power analysis attacks of modular exponentiation in smartcards. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 144–157. Springer.

Rabin, M. O. (1979). Digitalized signatures and public-key functions as intractable as factorization. Technical report, Massachusetts Inst of Tech Cambridge Lab for Computer Science.

Randolph, M. and Diehl, W. (2020). Power Side-Channel Attack Analysis: A Review of 20 Years of Study for the Layman. *Cryptography*, 4(2):15.

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.

Shan, W., Yu, J., Guo, L., Jiang, K., Wang, L., and Li, Q. (2019). A horizontal attack on sm9 signature generation. In *2019 15th International Conference on Computational Intelligence and Security (CIS)*, pages 306–309. IEEE.

Won, Y.-S., Sim, B.-Y., and Park, J.-Y. (2020). Key schedule against template attack-based simple power analysis on a single target. *Applied Sciences*, 10(11):3804.

Yen, S.-M., Lien, W.-C., Moon, S., and Ha, J. (2005). Power analysis by exploiting chosen message and internal collisions–vulnerability of checking

mechanism for RSA-decryption. In *International Conference on Cryptology in Malaysia*, pages 183–195. Springer.

Yilmaz, B. B., Prvulovic, M., and Zajić, A. (2019). Electromagnetic side channel information leakage created by execution of series of instructions in a computer processor. *IEEE Transactions on Information Forensics and Security*, 15:776–789.

Yuan, T. S., Sik, N. T., and Jian, C. J. (2019). Malaysian Digital Signature Algorithm Proposal: TNC Signature Scheme. Technical report. `https://myseal.cybersecurity.my/en/files/Malaysian_Digital_Signature_Algorithm_Proposal__TNC_Signature_Scheme.pdf` (Accessed November 13th, 2019).

Zhang, Q., Wang, A., Niu, Y., Shang, N., Xu, R., Zhang, G., and Zhu, L. (2018). Side-channel attacks and countermeasures for identity-based cryptographic algorithm SM9. *Security and communication networks*, 2018.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

The Post-Quantum Probabilistic Full Domain Hash

Mouhamed Lamine Mbaye*[1], Demba Sow[2], and Djiby Sow[3]

[1]*Faculté des Sciences et techniques, Université Cheikh Anta Diop de Dakar, Sénégal*
[2]*Département Mathématiques et Informatique*

E-mail: {mouhamedlamine1.mbaye, demba1.sow, djiby.sow}@ucad.edu.sn

ABSTRACT

Technology development allows cybercriminals to take their attack tools to unprecedented levels of sophistication and impact. Designing cryptographic protocols to guarantee the security of information systems is therefore always interesting. In this paper, we present an RSA-based signature scheme called post-quantum Probabilistic Full Domain Hash (pqPFDH) which is a variant of the RSA Full Domain Hash signature scheme. A random r is generated for each signature process allowing to compute the RSA exponentiation parameter. Under the RSA assumption, we prove that our scheme is unforgeable against adaptive chosen message attacks in the random oracle model. A much shorter random allows reaching a high-security level. We also show that it has a history-free reduction which implies its security against a quantum adversary.

Keywords: Signature scheme, Full Domain Hash, security reduction, Preimage Sampling trapdoor Function, history-free reduction

1 INTRODUCTION

Transmitting data or communicating securely remains more than ever a priority for companies or individuals. The hope of reaching the era of the quantum computer is a real motivation to direct research toward post-quantum cryptography to overcome the threat, caused by this famous computer on the problem of the factorization of integers, detected by Peter Shor since 1994 (Peter W. Shor , 1997). Advances in setting up alternative public-key cryptography algorithms called post-quantum algorithms that would resist quantum computers are undoubtedly notorious. In fact, the National Institute of Standard and Technology (NIST) launched the process of standardizing post-quantum algorithms in 2017. The objective is to evaluate post-quantum algorithms for cryptosystems that can withstand both classical and quantum computers and also interact with existing communication network protocols (see (L. Chen *et al.*, 2016, G. Alagic *et al.*, 2020)). Thereby, in order to keep the RSA public key system (R. Rivest , 1978) always in use, D. J. Bernstein *et al.* proposed the post-quantum RSA (D. J. Bernstein *et al.*, 2017) in that competition. It is a variant of the RSA cryptosystem designed to stop Shor's algorithm by using extremely large keys. The private key here is many primes, say a list of K (a power of 2) primes and the public key is the product of these primes. Although their submission didn't succeed for the second round, it is a good idea to think about it to preserve the use of RSA in a post-quantum era.

The digital signature appears to be an essential factor in the process of securing cryptographic protocols, in particular, to guarantee the integrity and authenticity of data. Thus, the implementation of applicable signature schemes in accordance with cryptographic principles will never be trivial.

The design of secure signature schemes has always been a concern of researchers. In this context, Bellare and Rogaway have developed a process for constructing signature schemes whose security is relative to that of the RSA. In 1996, they introduced the Full Domain Hash signature scheme and the Probabilistic Signature Scheme (see (M. Bellare and P. Rogaway, 1996)) which were proven secure against chosen message attack in the random oracle model (M. Bellare and P. Rogaway , 1993) assuming that inverting RSA is hard. The gap in this eminent work is that the security tightness of the FDH signature scheme can be enhanced. Then, it was revisited by other cryptographers like Coron (J. S. Coron, 2002, J. S. Coron , 2000) in order to strengthen the tightness security

by exhibiting different proof which provides tighter security reduction.
Our goal is to keep RSA still in use in a post-quantum era by generating large keys to establish a secure RSA-based signature scheme called Post-Quantum Probabilistic Full Domain Hash (pqPFDH). It's a variant of the Full domain Hash signature scheme with a random generated for each signature process. Its security, relatively close to that of RSA, will be proven in the random oracle model. In the security proof, we assume, like the FDH, that the hash function used is ideal and the RSA trapdoor permutation holds.
Contrary to the proof in the classical setting, each query made by a quantum adversary is a superposition of exponentially many states and the reduction algorithm have to evaluate the random oracle at all points in the superposition (Dan Boneh *et al.*, 2011). However, the security of a signature scheme in the classical random oracle model guarantees security in the quantum accessible random oracle model if the scheme is proven to have a history-free reduction. In this context, the classical proof of security simulates the random oracle and signature oracle in a history-free fashion. That is, its responses to queries do not depend on responses to previous queries or the query number. So showing that our scheme has this property implies its security in the quantum accessible random oracle.
The definition of a secure signature scheme ensures that an adversary, given the power to mount a chosen message attack, can not create an existential forgery, namely, the attacker can't output a valid signature pair for some new message. To prove the security of signature scheme, we generally proceeds by demonstrating that, if a polynomial-time adversary $\mathcal{A}$ can break the signature scheme, it can be used by a reduction algorithm $\mathcal{R}$ to invert in polynomial time some related one-way-function. Given an attacker $\mathcal{A}$ which can break the signature in time $\tau_{\mathcal{A}}$ with success probability at least $\varepsilon_{\mathcal{A}}$ for the reduction proof, $\mathcal{R}$ must simulate the environment of $\mathcal{A}$ and solve the problem (invert the one way function) with time $\tau_{\mathcal{R}} \geq \tau_{\mathcal{A}}$ and success probability $\varepsilon_{\mathcal{R}} \leq \varepsilon_{\mathcal{A}}$. For tightness of the reduction, it is required to have $\varepsilon_{\mathcal{R}} \approx \varepsilon_{\mathcal{A}}$ and $\tau_{\mathcal{R}} \approx \tau_{\mathcal{A}} + polynom(k)$, where k is a security parameter.
The paper is structured in six sections. The next section describes some previous works on the FDH signature scheme and its security tightness. In section 3, we present some preliminaries which help to understand the rest of the paper (trapdor function with preimage sampling 3.1, security model in 3.2, signature scheme in 3.3, hard problem for quantum computers 3.4 and post-quantum RSA problem 3.5). Section 4 prescribes the pqPFDH scheme which

is a post-quantum version of the PFDH signature scheme with a generated random allowing to compute the RSA exponent for each signature process. Subsection 4.1 illustrates the signature process, in subsection 4.2, we give the security proof of the scheme in the classical random oracle model and subsection 4.3 elucidates an approach allowing to translate the security of our scheme in the quantum accessible random oracle model. In section 5, we give a comparative security analysis and performance evaluation of our scheme to others which are both RSA-based signature schemes. Finally, section 6 concludes the paper.

2 RELATED WORKS

In order to strengthen the basic RSA signature, Bellare and Rogaway proposed in 1996 the Full Domain Hash signature scheme (M. Bellare and P. Rogaway, 1996). Assuming that the used hash function in the signature scheme is ideal, they proved that their scheme is secure against chosen message attack in the random oracle model. If f is a trapdoor permutation and $\mathcal{RO}$ is a random function from $\{0,1\}^*$ to the domain of f, they proved that signing a message m via $f^{-1}(\mathcal{RO}(m))$ is secure and proposed a security reduction for RSA-FDH where the reduction algorithm $\mathcal{R}$ will provide a perfect simulation and $(\varepsilon_{\mathcal{R}}, \tau_{\mathcal{R}})$-solves RSA trapdoor permutation with success probability $\varepsilon_{\mathcal{R}} \geq \frac{\varepsilon_{\mathcal{A}}}{q_h+q_{sig}+1}$ and time bound $\tau_{\mathcal{R}} \leq \tau_{\mathcal{A}}+(q_h+q_{sig}+1)polynom(k)$, where q_h and q_{sig} are respectively the number of hash queries and signature queries made by the attacker.

Coron studied the tightness of probabilistic FDH (J. S. Coron , 2000), and proposed a tight security reduction where the reduction algorithm $\mathcal{R}$ will provide a perfect simulation and $(\varepsilon_{\mathcal{R}}, \tau_{\mathcal{R}})$-solves RSA trapdoor permutation with success probability $\varepsilon_{\mathcal{R}} = \frac{\varepsilon_{\mathcal{A}}}{(1+6\frac{q_{sig}}{2^{k_0}})}$ and time bound $\tau_{\mathcal{R}} = \tau_{\mathcal{A}}+(q_h+q_{sig})\mathcal{O}(k^3)$ where q_h (respectively q_{sig}) is the number of queries made by the attacker to the oracle hash (respectively to the oracle signature).

To improve the tightness of RSA-FDH, Coron also proposed (J. S. Coron, 2002) a security reduction where the reduction algorithm $\mathcal{R}$ will provide a perfect simulation and $(\varepsilon_{\mathcal{R}}, \tau_{\mathcal{R}})$-solves RSA trapdoor permutation with success probability $\varepsilon_{\mathcal{R}} = \frac{\varepsilon_{\mathcal{A}}}{exp(1)q_{sig}}$ and time bound $\tau_{\mathcal{R}} = \tau_{\mathcal{A}} + (q_h + q_{sig} + 1)\mathcal{O}(k^3)$,

where q_h (respectively q_{sig}) is the number of queries made by the attacker to the oracle hash (respectively to the oracle signature).
In 2002, Dodis and Reyzin, generalizing Coron's work (Y. Dodis and L. Reyzin, 2003), showed that a similar result holds for any trapdoor permutation induced by a family of claw-free permutations. They proved that a tight security reduction is impossible for RSA-FDH, RSA-PFDH and PSS-R with small random. Moreover, they showed that, this cases, for any signature scheme outputs $sign(m) = \left(f^{-1}(\mathcal{RO}(m))\right)$ or $sign_r(m) = \left(f^{-1}(\mathcal{RO}(m,r)),r\right)$, where f^{-1} is the inverse of the trapdoor permutation, m is the message and r is a random, if the scheme is to be analyzed with a general "black-box" trapdoor permutation f.
In 2005, Dodis, Reyzin and Pietrzak, using previous work of Dodis *et al.*, proved (Y. Dodis and L. Reyzin, 2005) that one can't hope to prove $sign(m) = \left(f^{-1}(\mathcal{RO}(m))\right)$ secure under any assumption which is satisfied by random permutations. Pietrzak tells that their work does not mean that RSA-FDH with SHA1 is insecure, but it is just impossible to prove it (with a tight security reduction).
In 2009, Bégueline and Grégoire (S. Bégueline and B. Grégoire , 2009) used a general framework called CertiCrypt, allowing to formalize exact security proofs of cryptographic systems in the computational model, to prove the existential unforgeability under adaptive chosen-message attacks of the FDH and PFDH .
In 2011, Boneh *et al.* proved (Dan Boneh *et al.*, 2011) that FDH and PFDH signature schemes with preimage sampleable Trapdoor Functions has a history-free reduction. This means that their security in the classical random oracle model implies their security the quantum accessible random oracle model. They also proved the quantum security of a variant of FDH due to Katz and Wang (J. Katz , N. Wang , 2003) which has tight security reduction.
In 2018, Kiltz and Kakvi revisited Coron's work (J. Katz , N. Wang , 2018) and contradict the fact th at Coron said that the security loss of q_S in the proof of FDH is optimal and cannot possibly be improved. They showed that it only holds if the underlying trapdoor permutation is certified. They also use a stronger assumption called Phi-Hiding assumption introduced by Cachin *et al.* (C. Cachin *et al.*, 1999) to give a new tight security reduction of the FDH signature scheme.
In 2020, Yan Hue *et al.* proposed (Y. Hue *et al.*, 2020) a signature scheme called the Easy Simple Factoring-based (ESF) signature scheme proven se-

cure in the random oracle model. Compared to the FDH, their scheme has a more powerful efficiency performance in signing and verifying algorithms in the case that public keys are fixed to small bit lengths.

3 PRELIMINARIES

For a better understanding and consistency throughout the rest of the paper, we recall in this section some definitions and known results about signatures.

3.1 Trapdoor function with preimage sampling

A collection of trapdoor function with preimage sampling is a tuple of probabilistic polynomial time algorithms $(\mathcal{TG}, \mathcal{SD}, \mathcal{SP})$ that satisfies the following (Y. Dodis and L. Reyzin , 2008):
-$\mathcal{TG}$ is a generation algorithm which as input 1^n gives as output (s,t) where s specifies a function $f_s : D_n \longrightarrow R_n$. The domain D_n and the range R_n depend on n and t is a trapdoor for f_s;
- the algorithm $\mathcal{SD}$ with input 1^n samples an x from some distribution over D_n, for which the distribution of $f_a(x)$ is uniform over R_n;
- for every $y \in R_n$, $\mathcal{SP}(t,y)$ samples from the conditional distribution of $x \leftarrow \mathcal{SD}(1^n)$, given $f_s(x) = y$;
- one-wayness without trapdoor: for any PPT $\mathcal{A}$, the probability that $(\mathcal{A}, 1^n, a, y) \in f_a^{-1}(y) \subseteq D_n$ is negligible, where the probability is taken over the choice of a, the target value $y \leftarrow R_n$ chosen at random, and $\mathcal{A}$'s random coins.
This preimage sampling trapdoor function is not only one-way, but is also collision resistant.

3.2 Security model

Random Oracle Model: For any constant k, a random oracle is a function F_{rand} selected randomly in the set $\mathcal{F}_k$ of functions from $\{0,1\}^*$ to $\{0,1\}^k$.

Proof in the Random Oracle Model: Proof in the Random Oracle Model is described in (R. Canetti , O. Goldreich and S. Halevi , 2004, M. Bellare and P. Rogaway , 1993).
- Suppose that the hash function is a random function i.e in the simulation process, the hash function is replaced by a random oracle which outputs a random value for each new input.
- The only way to compute the hash function is to query the oracle hash.
- The reduction algorithm $\mathcal{R}$ must simulate the environment of the attacker $\mathcal{A}$ with her public key only.
- When the attacker $\mathcal{A}$ requests the oracle hash, the reduction algorithm $\mathcal{R}$ can choose the random to return as digest; hence $\mathcal{R}$ is able to embed the challenge (any information which able $\mathcal{R}$ to invert the related one way function at the end of the game) of his choice in the answer of a hash oracle query to $\mathcal{A}$.
- At the end of the simulation, if the attacker $\mathcal{A}$ outputs a valid forgery (which be never returned by the oracle signature) then $\mathcal{R}$ must be able to invert the related one-way function with good tightness.

3.3 Signature scheme

A randomized signature scheme is a tuple of three algorithms $\mathcal{S} = (\mathcal{KG}, \mathcal{SIG}, \mathcal{VER})$ described as follows:
- Key Generation algorithm($\mathcal{KG}$): with input a security parameter k, $\mathcal{KG}$ outputs a pair of keys (pub_{key}, sec_{key}).
- Signature algorithm ($\mathcal{SIG}$):
– with input a security parameter k, the signing algorithm produces a random r.
– with input (sec_{key}, m, r), the $\mathcal{SIG}$ outputs a signature σ.
- Verification algorithm ($\mathcal{VER}$): with input (m, σ, pub_{key}), the $\mathcal{VER}$ algorithm returns 1 if the signature is valid, and 0 otherwise.
Security of Randomized Signature Schemes: Goldwasser, Micali and Rivest introduced (S. Goldwasser *et al.*, 1988) the basic security notion for signatures called "existential unforgeability with respect to adaptive chosen-message attacks".
For this, a reduction algorithm $\mathcal{R}$ and an attacker $\mathcal{A}$ simulate the following game:
Setup: $\mathcal{R}$ runs the algorithm $\mathcal{SIG}$ with a security parameter k as input, to ob-

tain the public key pub_{key} and the secret key sec_{key}, and gives pub_{key} to the adversary.
Queries: Proceeding adaptively, $\mathcal{A}$ may request a signature on any message $m \in \mathcal{M}$ (multiple requests of the same message are allowed) and $\mathcal{R}$ will respond with (m, r, σ), where $\sigma = \mathcal{SIG}(sec_{key}, m, r)$ and r is a random. Let $Hist(\mathcal{S})$ be the signing data base (=set of signatures already outputted by the oracle signature to the queries of the $\mathcal{A}$).
Output: Eventually, $\mathcal{A}$ will output a pair (m, r, σ) and is said to win the game if $\mathcal{VER}(pub_{key}, m, r, \sigma) = 1$ and $(m, r, \sigma) \notin Hist(\mathcal{S})$ (this last condition forces the attacker $\mathcal{A}$ to output his own forgery).
The probability that $\mathcal{A}$ wins in the above game is denoted $Adv[\mathcal{A}, \mathcal{S}]$.
Unforgeability against Adaptive Chosen Message Attacks (EUF-CMA): A signature scheme $\mathcal{S} = (\mathcal{KG}, \mathcal{SIG}, \mathcal{VER})$ is existentially unforgeable with respect to adaptive chosen message attacks if for all probabilistic polynomial time attacker $\mathcal{A}$, $Adv[\mathcal{A}, \mathcal{S}]$ is negligible in the security parameter k.
BR-CMA: This adversary model is CMA where the number of random used by the probabilistic signature algorithm is fixed, says D. Hence the signer cannot sign (and outputs distinct values) the same message more than D times.

3.4 Hard problem for quantum computers

We briefly recall the definition of hard problem and history-free reduction for quantum computers introduced by D. Boneh et *al.* (Dan Boneh *et al.*, 2011).

Definition 3.1. *For a classical challenger $\mathcal{C}$ and a classical or quantum adversary $\mathcal{A}$, a problem $\mathcal{P}$ is a pair $(\mathcal{G}_\mathcal{P}, \alpha_\mathcal{P})$, where $\alpha_\mathcal{P}$ is a real number between 0 (inclusive) and 1 (exclusive) and $\mathcal{G}_\mathcal{P}$ specifies a game between $\mathcal{A}$ and $\mathcal{C}$ as follows:*
- given as input (1^n), $\mathcal{C}$ computes a value x and sends it to $\mathcal{A}$;
- $\mathcal{A}$ runs on x, and allowed to make classical queries to $\mathcal{C}$;
- $\mathcal{A}$ outputs a value y and send it to $\mathcal{C}$;
- $\mathcal{C}$, with x, y and the classical queries made by the adversary, outputs 1 or 0.
We say that $\mathcal{A}$ wins the game if $\mathcal{C}$ outputs 1. $\mathcal{A}$'s advantage with respect to the game $\mathcal{P}$ is defined as follows:

$$Adv[\mathcal{A}, \mathcal{P}] = |Pr[W] - \alpha_\mathcal{P}|$$

where W is the probability that $\mathcal{A}$ wins the game.

Definition 3.2. *Hard Problem. A problem $\mathcal{P} = (\mathcal{G}_{\mathcal{P}}, \alpha_{\mathcal{P}})$ is hard for quantum computer if, for all polynomial time quantum adversary $\mathcal{A}$, the value $Adv[\mathcal{A}, \mathcal{P}]$ is negligible.*

Definition 3.3. *History-free Reduction. Let $\mathcal{S} = (\mathcal{GEN}, \mathcal{SIG}, \mathcal{VER})$ be a signature scheme proven secure in the classical random oracle model and a hard problem $\mathcal{P}$. Then $\mathcal{S}$ has a history-free reduction from $\mathcal{P}$ if there is a proof of security that uses a classical probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ for $\mathcal{S}$ to construct a classical PPT adversary $\mathcal{B}$ for $\mathcal{P}$ such that:*
- The algorithm $\mathcal{B}$ internally runs these four algorithms $\mathcal{STR}$, $\mathcal{RND}$, $\mathcal{SIG}$, $\mathcal{FIN}$ which have access to a shared classical random oracle $\mathcal{O}_c$, except for $\mathcal{STR}$. These algorithms are used as follows:
** Given an instance x for problem P as input, algorithm $\mathcal{B}$ first runs $\mathcal{STR}(x)$ to obtain (pk, z) where pk is a signature public key and z is private state to be used by $\mathcal{B}$. Algorithm $\mathcal{B}$ sends pk to $\mathcal{A}$ and plays the role of challenger to $\mathcal{A}$.*
** When $\mathcal{A}$ makes a classical random oracle query to $\mathcal{O}(r)$, $\mathcal{B}$ responds with $\mathcal{RND}(r, z)$.*
** When $\mathcal{A}$ makes a classical signature query $Sig(sk, m)$, $\mathcal{B}$ responds with $\mathcal{SIG}(m, z)$.*
** When $\mathcal{A}$ outputs a signature forgery candidate (m, σ), $\mathcal{B}$ outputs $\mathcal{FIN}(m, \sigma, z)$.*
- There is an efficiently computable function $\mathcal{I}(pk)$ which produces an instance x of problem $\mathcal{P}$ such that $\mathcal{STR}(x) = (pk, z)$ for some z. Consider the process of first generating (sk, pk) from $\mathcal{GEN}(1^n)$, and then computing $x = \mathcal{I}(pk)$. The distribution of x generating in this way is negligibly close to the distribution of x generated in $\mathcal{G}_{\mathcal{P}}$.
- For fixed z, consider the classical random oracle $\mathcal{O}(r) = \mathcal{RND}(r, z)$. Define a quantum oracle $\mathcal{Q}$ which transforms a basis element $|x, y\rangle$ into $|x, y \oplus \mathcal{O}(x)\rangle$. We require that $\mathcal{Q}$ is quantum computationally indistinguishable queries abort is non-negligible.
- If (m, σ) is a valid signature forgery relative to the public key pk and oracle $\mathcal{O}(r) = \mathcal{RND}(r, z)$ the the output of $\mathcal{B}$ causes the challenger for problem $\mathcal{P}$ to output 1 with non-negligible probability.

Theorem 3.1. *Let $\mathcal{S} = (\mathcal{GEN}, \mathcal{SIG}, \mathcal{VER})$ be a signature scheme. Suppose that there is a history-free reduction that uses a classical PPT adversary $\mathcal{A}$ for $\mathcal{S}$ to construct a PPT algorithm $\mathcal{B}$ for a problem P. Furthermore, assume that $\mathcal{P}$ is hard for polynomial-time quantum computers, and quantum-accessible*

pseudorandom functions exist. Then S is secure in the quantum-accessible random oracle model.

For the proof of this theorem, one can refer to (Dan Boneh *et al.*, 2011).

3.5 Post quantum RSA

Post-quantum RSA consists of adjusting RSA parameters to make extremely large keys that will resist the power of quantum computers. The idea is to use many small primes which constitute the secret key and their product gives the public key. Clearly, a user generates K primes $q_1, q_2, ..., q_K$ of the form $q_i = 2p_i^{\beta_i} + 1$, where $1 \leq i \leq K$, p_i integers and establishes $n = \prod_{i=0}^{K} q_i$. Now, for a given pqRSA modulus n, an integer e coprime with $\varphi(n) = (q_1 - 1)(q_2 - 1)\cdots(q_K - 1)$ and $z \in (\frac{\mathbb{Z}}{n\mathbb{Z}})^*$, find x such that $x^e = z \mod n$, is the pqRSA problem. Then, an algorithm $\mathcal{R}$ is said to $(\tau_{\mathcal{R}}, \varepsilon_{\mathcal{R}})$-solve the pqRSA problem, if in at most $\tau_{\mathcal{R}}$ operations, $Pr\{(n, e) \leftarrow RSA(1^k), z \leftarrow (\frac{\mathbb{Z}}{n\mathbb{Z}})^*, x \leftarrow \mathcal{R}(n, e, z), x^e = z \mod n\} \geq \varepsilon_{\mathcal{R}}$, where the probability is taken over the distribution of (n, z) and over $\mathcal{R}$'s random tapes.

4 THE POST-QUANTUM PROBABILISTIC FULL DOMAIN HASH SIGNATURE SCHEME

In this section, we exhibit the pqPFDH signature scheme and prove that it is EUF-BR-CMA secure under the RSA assumption in the random oracle model. It is similar to the Probabilistic Full Domain Hash, but with a random exponent generated for each signature process and used to compute the encryption exponent.
Our main objective is to prove that the previous reduction technical can be adapted to RSA-PFDH in order to have the same success probability as above: $\varepsilon_{\mathcal{R}} = \varepsilon_{\mathcal{A}}(1 - \frac{1}{2^{|a|}})$, where $|a|$ is the size of the random.

4.1 Signature and vérification process

Here, we elucidate the different steps to follow to generate a signature according to our scheme.
Key generation algorithm $\mathcal{KG}$: the key generation algorithm works as follows:
– with input 1^k, $\mathcal{KG}$ generates a list of K primes $q_1, q_2, \ldots, q_K$ of the form $q_i = 2p_i^{\beta_i}+1$ for $1 \leq i \leq K$ and computes the modulus $n = q_1 \times q_2 \times \cdots \times q_K$. Then the public key is $p_{key} = n$ and the secret key is $s_{key} = (q_1, q_2, \ldots, q_K)$.
– It also uses a function G which, on input a random $a \in \{0,1\}^{k_a}$, outputs an odd integer e which is used to compute d such that $e \cdot d = 1 \ mod \ \varphi(n)$.
$G : \{0,1\}^{k_a} \longmapsto \{0,1\}^{k_e} \cap \mathbb{N}_{odd}$, where $\{0,1\}^{k_e}$ is the set of bits representation of the integer e such that $e.d = 1 \ mod \ \varphi(n)$.
– After concatenated the message m with the random a and the exponent e, one uses the hash function $H : \{0,1\}^{k_a+k_m+k_e} \longmapsto \mathbb{Z}_n^*$, where k_m is the length of the message m.

Remark 4.1. *To build G, one can proceed as follows:*
Let $G' : \{0,1\}^{k_a} \to \{0,1\}^{k_e-2}$ be a hash function where k_e is a security parameter, define the hash function G by $G(x) = 1||G'(x)||1 \in [0, n-1] \cap [2^{k_e}, 2^{k_e+1}[\cap \mathbb{N}_{odd}$, where $\mathbb{N}_{odd}$ is the set off odd integers.

Remark 4.2. *Since $G(x)$ is odd then, with a high probability, $G(x)$ is co-prime with $\varphi(n) = 2^K p_1^{\beta_1} p_2^{\beta_2} \ldots p_K^{\beta_K}$ because finding an odd integer which is not co-prime with $\varphi(n)$ is equivalent to factor n (which is known to be difficult).*

Signature algorithm $\mathcal{SIG}(m, s_{key})$**:** To sign a message m, the signer Bob should do the following steps:

(1) pick the private key $(q_1, q_2, \ldots, q_K)$ and the two hash functions G and H;

(2) select a random salt $a \in \{0,1\}^{k_a}$, compute $e = G(a)$ and $d = e^{-1} \mod \varphi(n)$;

(3) compute $y \longleftarrow H(e||m||a)$ and $\sigma \longleftarrow y^d \mod n$;

(4) The signature of m is (m, a, σ).

Verification algorithm $\mathcal{VER}(p_{key}, (m, a, \sigma))$**:** To verify the signature (m, a, σ) submitted by Bob, the verifier Alice should do the following steps:

(1) pick n, Bob's public key and the the signature (m, a, σ);

(2) compute $e = G(a)$ and $H(e||m||a) = y$;

(3) if $y = \sigma^e \mod n$, then return 1; else return 0.

4.2 Security proof: Reduction in the random oracle model

Assuming that inverting RSA is a hard problem, the following theorem proves the security of the pqPFDH signature in the random oracle model. It shows that with pqPFDH, we obtain tight security proof using much shorter random (a random of 5 bits).

Theorem 4.1. *If there exists an attacker $\mathcal{A}$ that $(q_H, q_G, q_{sig}, \tau_{\mathcal{A}}, \varepsilon_{\mathcal{A}})$-solves EUF-BR-CMA the pqPFDH signature scheme, then there exists a reduction $\mathcal{R}$ simulating the environment of $\mathcal{A}$ in the random oracle model that $(\tau_{\mathcal{R}}, \varepsilon_{\mathcal{R}})$-solves RSA with success probability $\varepsilon_{\mathcal{R}} = \varepsilon_{\mathcal{A}}(1 - \frac{1}{2^{|a|}})$ and time bound $\tau_{\mathcal{R}} \leq \tau_{\mathcal{A}} + q_G\mathcal{O}(1) + (q_H + q_{sig} + 2)\mathcal{O}(k^3)$, where $|a|$ is the size of random used for exponentiation, $\mathcal{R}$ receives q_{sig} signature queries, q_H and q_G queries respectively for the hash oracles H and G from $\mathcal{A}$ and k is a security parameter.*

Proof. Our reduction $\mathcal{R}$ behaves as follows:
- $\mathcal{R}$ is given ($n \leftarrow RSA(1^k)$, generates at random $t \leftarrow [0, n-1] \cap \mathbb{N}_{odd}$ and $y \leftarrow \frac{\mathbb{Z}}{n\mathbb{Z}}^*$, as well as an attacker $\mathcal{A}$ that $(q_H, q_G, q_{sig}, \tau_{\mathcal{A}}, \varepsilon_{\mathcal{A}})$-solves EUF-BR-CMA(pqPFDH);
- $\mathcal{R}$ simulates G_{key} and transmits some public key $p_{key} = n$ to $\mathcal{A}$;
- $\mathcal{R}$ receives queries for G from $\mathcal{A}$: it will have to simulate G at most q_G times;
- $\mathcal{R}$ receives queries for H from $\mathcal{A}$: it will have to simulate H at most q_H times;
- $\mathcal{R}$ receives signature queries from $\mathcal{A}$: it will have to simulate a signing oracle at most q_{sig} times;
- $\mathcal{A}$ outputs a forgery (m, a, S) for pqPFDH;
- $\mathcal{R}$ simulates a verification of the forgery which is valid with probability $\varepsilon_{\mathcal{A}}$;

- $\mathcal{R}$ outputs x such that $x^t = y \mod n$.

Simulation of oracle key generation $\mathbf{G}_{key}$: The reduction $\mathcal{R}$

- sets $Hist(S) = \emptyset$ (Signing oracle database);
- sets $Hist[G] = \emptyset$ (Oracle **G** database);
- sets $Hist[H] = \emptyset$ (Oracle **H** database);
- sends the pqPFDH public key n to $\mathcal{A}$;
- selects M (with $M < 2^{|a|}$) random integers $i_1, ..., i_M \in [1; 2^{|a|}]$ where $|a|$ is the size of random used for exponentiation in signature process and sets $L = \{i_1, ..., i_M\}$.

Simulation of oracle hash G: when $\mathcal{A}$ queries G with a random a_j, $1 \leq j \leq L$.

- $\mathcal{R}$ checks in $Hist[G]$, if a_j was queried in the past. If $e_j = G(a_j)$, is already defined to a value $e_j = G_{a_j}$, returned this value.
- If $j \notin L$ $\mathcal{R}$ picks at random $g_{a_j} \leftarrow [0, n-1] \cap \mathbb{N}_{odd}$ and defines $e_j = g_{a_j}$.
- If $j \in L$: $\mathcal{R}$ picks at random $g'_{a_j} \leftarrow [0, n-1] \cap \mathbb{N}_{odd}$ and defines $e_j = tg'_{a_j}$.
- memorizes (a_j, e_j) in $Hist[G]$.

Simulation of oracle hash H: when $\mathcal{A}$ queries H with message and a random (m, a_j),

– $\mathcal{R}$ invokes its own simulation to compute $e_j = G(a_j)$,

– checks in $Hist[H]$, if $e_j||m||a_j$ was queried in the past. If $H(e_j||m||a_j)$ is already defined as $h_{e_j||m||a_j}$, returned $h_{e_j||m||a_j}$;

– if $j \notin L$, $\mathcal{R}$;

* picks λ_{a_j} at random;
* defines and returns $H(e_j||m||a_j) = \lambda_{a_j}^{e_j} \mod n$ to $\mathcal{A}$;
* memorizes $(m, a_j, e_j, \lambda_{a_j}, \lambda_{a_j}^{e_j})$ in $Hist[H]$

– if $j \in L$, $\mathcal{R}$;

* picks λ_{a_j} at random;
* defines and returns $H(e_j||m||a_j) = \lambda_{a_j}^{e_j} y \mod n$ to $\mathcal{A}$;
* memorizes $(m, a_j, e_j, \perp, y)$ in $Hist[H]$;

Simulation of oracle signature $\mathbf{S}^{\mathbf{G},\mathbf{H}}$: when $\mathcal{A}$ requests the signature of some message m, $\mathcal{R}$,

- selects randomly j in $[1, 2^{|a|}] \setminus L$, after $\mathcal{R}$;
- invokes its own simulation of G and H to compute $e_j = G(a_j)$ and $H(e_j||m||a_j)$;
- search the unique $(m, a_j, e_j, \alpha, \beta)$ in $Hist[H]$ and returns (m, a_j, α);
- store (m, a_j, α) in oracle database $Hist[S]$.

Simulation of $\mathbf{V}^{\mathbf{G},\mathbf{H}}$: Given (m, a, S), $\mathcal{R}$

- invokes its own simulation of H and G to get $e = G(a)$ and $H(e||m||a)$;

- outputs 1 if $H(e||m||a) = S^e(\mod n)$ and $(m, a, S) \notin Hist(S)$ or 0 otherwise.

Final Outcome: assume that at the end of the game, $\mathcal{A}$ outputs (m^*, a_d, S^*) as a forgery. Then,

- $\mathcal{R}$ simulates $V^{G,H}$ to verify if (m^*, a_d, S^*) is a valid forgery which means that $H(e_d||m||a_d) = S^{e_d}(\mod n)$ and $(m^*, a_d, S^*) \notin Hist(S)$ (this last condition allows to force the attacker to output his own forgery instead of using a signature of the real signer as forgery);

-if (m^*, a_d, S^*) is invalid, $\mathcal{R}$ aborts;

- if $d \notin L$, $\mathcal{R}$ aborts;

- $\mathcal{R}$ sets $x = (\frac{S^*}{\lambda_{a_d}})^{e_d/t}$;

- $\mathcal{R}$ outputs x.

Tightness of this reduction:

- $\mathcal{R}$ perfectly simulates the scheme (oracles key generation, hash, signature and verification) with probability 1.

- $\mathcal{A}$ then outputs (m^*, a_d, S^*) with probability at least $\varepsilon_{\mathcal{A}}$ after time $\tau_{\mathcal{A}}$,

- since L is independent from $\mathcal{A}$, the event $d \in L$ occurs with probability $\frac{M}{2^{|a|}}$.

- Hence $\mathcal{R}$ then outputs a solution $(x; y)$ for $RSA[n; e; y]$ with probability one.

Summing up, $\mathcal{R}$ succeeds with probability $\varepsilon_{\mathcal{R}} = \varepsilon_{\mathcal{A}} \frac{M}{2^{|a|}}$ and time bound is given by $\tau_{\mathcal{R}} \leq \tau_{\mathcal{A}} + q_G\mathcal{O}(1) + (q_H + q_{sig} + 2)\mathcal{O}(k^3)$ where $|a|$ is the size of random used for exponentiation in pqPFDH signature scheme.

To have $\varepsilon_{\mathcal{R}} \geq \varepsilon_{\mathcal{A}}(1 - \frac{1}{2^{|a|}})$ it will suffice to choose the maximal value of $M = 2^{|a|} - 1$. □

4.3 Reduction in quantum accessible random oracle model:

In this part, we instantiate the pqPFDH signature scheme with a preimage Sampleable Trapdoor Function (PSF) before making it history-free. Consider the tuple $(\mathcal{TG}, \mathcal{SD}, \mathcal{SP})$ (as defined in 3.1) which specifies a collection of trapdoor collision-resistant hash functions that operates relative to a function $\mathcal{H} : \{0,1\}^n \longrightarrow \mathbb{Z}_n^*$ modeled as a random oracle.

- $\mathcal{KG}(1^n)$ runs $\mathcal{TG}$ which outputs the couple (s, t), where s specifies the function f_s and t is a trapdoor information for f_s. Then, a represents the verification key and t the signing key.

- $\mathcal{SG}(t,m)$: with a randomly chosen $r \in \{0,1\}^k$, compute $e = G(r)$ where $G : \{0,1\}^{k_a} \longmapsto \{0,1\}^{k_e} \cap \mathbb{N}_{odd}$, $y \leftarrow \mathcal{H}(r||e||m)$ and $\sigma \leftarrow \mathcal{SP}(t,y)$. The signature is the tuple (r,σ).
- $\mathcal{VR}(s,(r,m,\sigma))$: if $\sigma \in D_n$, $e = G(r)$ an odd integer and $f_s(\sigma) = \mathcal{H}(r||e||m)$, then accept, else reject.
The scheme described above is UF-CMA secure in the random oracle model. To ensure the security of pqPFDH in the quantum accessible random oracle model, we make its reduction to history-free fashion. So, let $\mathcal{A}$ be an adversary for this defined pqPFDH-PSF scheme and $\mathcal{B}$ the reduced classical collision finder for the trapdoor function PSF.
- On input $a \in \{0,1\}^{k_a}$, $\mathcal{B}$ computes $\mathcal{STR}(a) := G(a) = e$, and simulates $\mathcal{A}$ on e.
- When $\mathcal{A}$ queries $\mathcal{O}(r)$, $\mathcal{B}$ responds with $\mathcal{RND}(r,e) = f_s(e, \mathcal{SD}(1^n, \mathcal{O}(r||e)))$.
- When $\mathcal{A}$ queries $S(s_{key,m})$, $\mathcal{B}$ responds with $\mathcal{SIG}(m,e) = \mathcal{SD}(1^n, \mathcal{O}(r||e||m))$.
- When $\mathcal{A}$ outputs (m,σ), $\mathcal{B}$ outputs $\mathcal{FIN}(m,\sigma,r) := (\mathcal{SD}(1^n, \mathcal{O}(r||e||m)),\sigma)$.
The above reduction ensure the history-free criterion for pqPFDH. So, according to Theorem 1, this implies the security in the quantum accessible random oracle model .

5 SECURITY ANALYSIS AND PERFORMANCES

The pqPFDH signature scheme presented in this work uses a generated random salt to compute the exponentiation in the RSA trapdoor permutation at each signature process. The reduction algorithm has as success probability $\varepsilon_{\mathcal{R}} = \varepsilon_{\mathcal{A}}(1 - \frac{1}{2^{|a|}})$ and time bound $\tau_{\mathcal{R}} \leq \tau_{\mathcal{A}} + q_G\mathcal{O}(1) + (q_h + q_{sig} + 2)\mathcal{O}(k^3)$, where $2^{|a|}$ is the number of random allowed to use for the exponentiation in the signature process. To have this good success probability, we use signature with random such that the trapdoor permutation is randomly chosen and inverted at each signature, namely, the general form of our signature is $sign_a(m) = \big(m, a, f_a^{-1}(\mathcal{RO}(m,a,e))\big)$.
We remark that the security reduction for the pqPFDH signature scheme is tight because it is equivalent to those of RSA. We see that with a random of 5 bits, we have $\varepsilon_{\mathcal{R}} = 0.96875\ \varepsilon_{\mathcal{A}}$, where $\varepsilon_{\mathcal{R}}$ is the success probability of an algorithm $\mathcal{R}$ that can break RSA by using an attacker $\mathcal{A}$ that breaks pqPFDH with probability $\varepsilon_{\mathcal{A}}$. But for RSA-PFDH, it is necessary to use random with

r	1	2	3	4	5	6	7	8
$\frac{\varepsilon_{\mathcal{R}}}{\varepsilon_{\mathcal{A}}}$	0.5	0.75	0.875	0.937	0.968	0.992	0.996	0.998

Table 1: Security gap between pqPFDH and RSA

size greater than $\log_2 q_{sig} + 8$ (≥ 38), where q_{sig} is the number of signatures queries, in order to have $\varepsilon_{\mathcal{R}} = 1.04\ \varepsilon_{\mathcal{A}}$. We also remark that the success probability of the simulation is independent of the number of signing and hashing oracles queries. Our new signature scheme is more secure than RSA-PFDH relatively to all known reductions, but it is also slower. So we win in terms of safety but, in terms of efficiency, we lose.

Compared with the FDH and PFDH, the advantage of our scheme lies in the choice of the random. With a much smaller random size, we obtain a security level much closer to those of the RSA. In fact, Bellare and Rogaway showed (M. Bellare and P. Rogaway, 1996) that the security reduction for FDH bounds the probability ε of breaking FDH in time τ by $(q_h + q_s) \cdot \varepsilon'$, where ε' is the probability of inverting RSA in time τ' close to τ and where q_h and q_s are the numbers of hash queries and signature queries performed by the forger. This was improved by Coron (J. S. Coron , 2000) to $\varepsilon = q_s \cdot \varepsilon'$, which is better since in practice q_s happens to be smaller than q_h. However, the security reduction is still not tight, and FDH is still not secure as inverting RSA.

To prove the security of pqPFDH in the quantum accessible random oracle model, the trick is, as elucidated for FDH (M. Zhandry, 2015) (theorem 5.3), to use a quantum one way trapdoor permutation. We use the notion with preimage sampling trapdoor function which is built upon a collection of trapdoor collision-resistant hash functions given by $(\mathcal{TG}, \mathcal{SD}, \mathcal{SP})$ and operates relative to the function $H : \{0,1\}^* \rightarrow \mathbb{Z}_n^*$ that is modelled as a random oracle. It can be considered for certain random lattices that are as hard ad worst-case lattices (Y. Dodis and L. Reyzin , 2008). Then, by transforming it in a history form ensure that the scheme is quantum UF-CMA-secure since H is modeled as a random oracle.

6 CONCLUSION

In this work, we describe a new signature scheme called post-quantum Probabilistic Full Domain Hash (pqPFDH) which is a variant of the RSA-FDH signature scheme. A random is generated for each signature process allowing to compute the RSA exponentiation. Our scheme is proven secure against chosen message attacks in the random oracle model, under the RSA assumption. The security reduction has a better security proof than PFDH because much shorter random salt (a random of size 5; see **Table 1**) is sufficient to achieve the same security level.
We also prove the security of our scheme against an adversary that has access to a quantum computer. We use the notion of history-free reduction which ensures that security in the classical random oracle model implies security in the quantum-accessible random oracle model. So, showing that pqPFDH has a history-free reduction from a given hard problem called preimage sampling trapdoor permutation highlights roughly speaking its security in the quantum-accessible random oracle model, which means that pqPFDH is secure against an attacker that has access to a quantum computer.

REFERENCES

D. J. Bernstein *et al.* (2017). Post quantum rsa. *International Workshop on Post-Quantum Cryptography*, pages 311–329.

J. Katz , N. Wang (2003). Efficiency improvements for signature schemes with tight security reductions. *Conference on Computer and communications security*.

J. Katz , N. Wang (2018). Optimal security proofs for full domain hash, revisited. *Journal of Cryptology*, 31:276–306. `https://doi.org/10.1007/s00145-017-9257-9`.

J. S. Coron (2002). Optimal security proofs for pss and other signature schemes. *Proceedings of Eurocrypt'02, lncs, vol. 2332, Springer-Verlag*, 2332:272–287.

L. Chen *et al.* (2016). Report on post-quantum cryptography. *Proceedings of Eurocrypt'02, lncs, vol. 2332, Springer-Verlag.* `http://dx.doi.org/10.6028/NIST.IR.8105`.

M. Zhandry (2015). Secure identity-based encryption in the quantum random oracle model. *International Journal of Quantum Information*, 13(04):1484–1509.

R. Canetti , O. Goldreich and S. Halevi (2004). The random oracle methodology, revisited. *Journal of the ACM*, 51:557–594.

R. Rivest (1978). A method for obtaining digital signatures and public key cryptosystems. *CACM*, 21.

Y. Dodis and L. Reyzin (2003). On the power of claw-free permutations. *In Security in Communication Networks (SCN 2002), Lecture Notes in Computer Science, Springer*, 2576:55–73.

C. Cachin *et al.* (1999). Computationally private information retrieval with polylogarithmic communication. *Advances in Cryptology EUROCRYPT 99 , Lecture Notes in Computer Science, Springer Berlin Heidelberg (1999)*, pages 402–414.

Dan Boneh *et al.* (2011). Random oracles in a quantum world. *International conference on the theory and Application of Cryptology and Information Security, ASIACRYPT 2011. Springer*, pages 41–69.

G. Alagic *et al.* (2020). Status report on the second round of the nist post-quantum cryptography standardization process. 8309. `https://doi.org/10.6028/NIST.IR`.

J. S. Coron (2000). On the exact security of full domain hash. *Proccedings of Crypto's 2000, LNCS, Springer Verlag 2000*, 1880:229–235.

M. Bellare and P. Rogaway (1993). Random oracles are practical: a paradigm for designing efficient protocols. *Proceedings of the First Annual Conference on Computer and Communications Security, ACM.*

M. Bellare and P. Rogaway (1996). The exact security of digital signatures: How to sign with rsa and rabin. *Proceedings of Eurocrypt 1996, lncs, Springer-Verlag*, 1070:399–416.

Peter W. Shor (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *Soicety for Industrial Applied Matheematics (SIAM), J. Comput*, 26(5):1484–1509.

S. Bégueline and B. Grégoire (2009). Formally certifying the security of digital signature schemes. *30th IEEE Symposium in Security and Privacy.*

S. Goldwasser *et al.* (1988). A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal of computing*, 17(2):281–308.

Y. Dodis and L. Reyzin (2008). Trapdoors for hard lattices and new cryptographic constructions. *Proceedings of the fortieth annual ACM symposium on Theory of computing, May 2008*, pages 197–206. `https://doi.org/10.1145/1374376.1374407`.

Y. Dodis and L. Reyzin (2005). On the generic insecurity of full-domain hash. *Advances in Cryptology-CRYPTO 2005. 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18.*

Y. Hue *et al.* (2020). Easy simple factoring-based digital signature scheme. *15th International Conference for Internet Technology and Secured Transactions (ICITST), IEEE.*

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

Analysis of Permutation Functions for Lightweight Block Cipher Design

Abdul Alif Zakaria[*1,3], **A. H. Azni**[*1,2], **Farida Ridzuan**[1,2], **Nur Hafiza Zakaria**[1,2], and **Maslina Daud**[3]

[1]*Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai, 71800, Negeri Sembilan, Malaysia*
[2]*CyberSecurity and System Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai, 71800, Negeri Sembilan, Malaysia*
[3]*CyberSecurity Malaysia, Cyberjaya, 63000, Selangor, Malaysia*

E-mail: alif@cybersecurity.my; ahazni@usim.edu.my;
[*]*Corresponding author*

ABSTRACT

In this paper, the permutation functions of the lightweight block ciphers are analyzed to observe their impact on the cryptographic strength. Three types of permutation functions are presented in the analysis which includes the Feistel-based permutation, formulation-based permutation, and table-based permutation. In order to execute the avalanche effect and correlation coefficient analysis, one hundred 64-bit plaintexts are generated as the input using a pseudorandom bit generator. From the analysis, the avalanche effect results show that all permutation methods have the ability to maintain the optimum output changes with minor or major modifications of the input. In addition, the correlation coefficient results indicate a weak linear relationship between the cipher input and its corresponding output, thus the output produced from all permutation methods is not linear with the input data.

Keywords: Permutation, lightweight, block cipher, encryption, avalanche effect, correlation coefficient

1 INTRODUCTION

Lightweight block cipher is a standard when it comes to security protections on resource-constrained devices. The prospect of lightweight algorithm has caught the interest of academics, industry, and government. Development of lightweight algorithms started as early as the 1990's with the emergence of IDEA (Lai and Massey, 1991), Blowfish (Schneier et al., 1993), and TEA (Wheeler and Needham, 1994). Hundreds of lightweight block ciphers including the recently published SCENERY (Feng and Li, 2022) and ESIT (Nayak and Swain, 2022) algorithms have been proposed up until this point. Existing block ciphers such as AES and PRESENT have inspired the new designs of lightweight algorithms despite their age of development (Girija et al., 2020). Cryptographic components of both mentioned algorithms are still relevant and may be used to build new lightweight block ciphers.

Two main fundamental structures of lightweight block ciphers are the Feistel network and Substitution-Permutation Network (SPN). In Feistel network lightweight block cipher, the encryption and decryption processes are similar but the round function has weak diffusion property (Liu et al., 2019). On the contrary, the SPN-based algorithm offers high confusion and diffusion properties, however, the encryption and decryption operations are non-identical that need additional cycles and codes for execution (Singh et al., 2019).

A lightweight block cipher should provide confusion and diffusion properties in its design components to provide sufficient security to the algorithm (Sehrawat et al., 2019). Confusion obscures the relationship between the plaintext and ciphertext with substitution function, while diffusion spreads the plaintext statistics through the ciphertext using permutation function.

In this paper, the permutation functions of the lightweight block ciphers are discussed. Three types of permutation functions are presented to observe their impact on cryptographic strength. To the best of our knowledge, there is no analysis has been conducted in evaluating the individual permutation functions at the component level to test their performance on a single round function. Therefore, our finding has a new contribution to the body of knowledge, especially to those who intended to develop an encryption algorithm.

The organization of the paper is structured as follows. Section 2 defines the permutation functions of lightweight block ciphers. Section 3 presents the analysis of permutation functions. Section 4 provides the results and discussion of the analysis. Lastly, Section 5 concludes the research work.

2 PERMUTATION FUNCTION

Permutation or linear layer is a core component in SPN block cipher (Turan et al., 2021). The component can influence the security and efficiency of the cryptographic algorithm. Permutation function consists of three methods that include the Feistel-based permutation, formulation-based permutation, and table-based permutation as shown in Figure 1. From the literature, every lightweight block cipher must have implemented at least one of the three permutation methods. These permutation methods would add diffusion property to the algorithm which is one of the most important characteristics of the cryptographic design (Zakaria et al., 2020). The classification of the permutation methods of each lightweight block cipher as shown in Table 1, Table 2, and Table 3 are based on the presentation of the original algorithm papers which described their permutation layers.

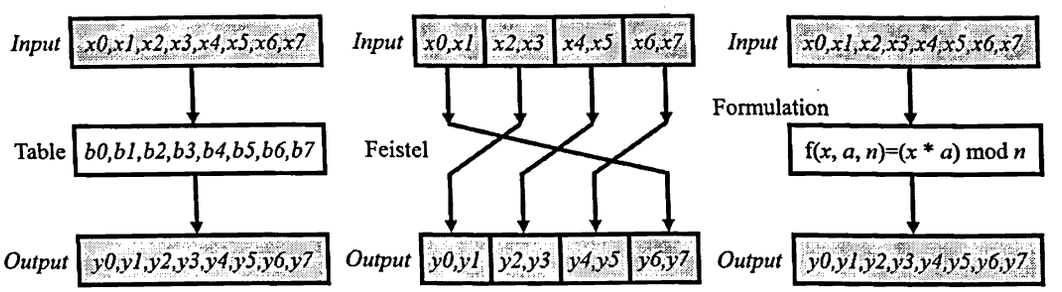


Figure 1: Permutation Functions

2.1 Table-based Permutation

Table-based permutation is a common function in block ciphers that relocates the cipher data to a specific position defined by the permutation table. The

Table 1: Types of Table-based Permutation

Type	Data/Block (bit)	Data Distribution Ratio (%)	Algorithm
Bit	1/8	12.50%	DoT and ICEBERG
	1/16	6.25%	HERMES, LBC-IoT, MANTIS, NUX, and QTL
	1/32	3.13%	ANU, SFN, and VAYU
	1/64	1.56%	ACT, GIFT, Halka, PICO, PRESENT, and PUFFIN
Nibble	4/32	12.50%	MIBS
	4/64	6.25%	LILLIPUT, Midori, and TWINE

table can be structured for various forms of cipher data such as bits or nibbles to suit the algorithm. There are also many sizes of permutation tables being implemented in lightweight block ciphers such as 8, 16, 32, and 64 bits.

Table-based bit permutation is implemented in PRESENT given the simplicity of the method which is being adopted in many algorithms (Bogdanov et al., 2007). The regular table-based bit permutation helps VAYU to improve the robustness of the algorithm round function (Bansod et al., 2016). Cipher bits that are permuted using a permutation table give complete diffusion with just three rounds of ACT algorithm (Jithendra and Kassim, 2020). The table-based bit permutation is easy to be implemented in hardware and software environments of SFN algorithm (Li et al., 2018). ICEBERG table-based permutation is designed to disturb the bit alignment of the algorithm to provide resistance against cryptanalytic attacks (Standaert et al., 2004). Lastly, the table-based bit permutation of Halka algorithm proved that no trails can be constructed thus preventing structural attacks (Das et al., 2014).

Besides the table-based bit permutation, a table-based nibble permutation is also being used such as Midori which improved diffusion speed and increased the number of active S-boxes in each encryption round (Banik et al., 2015). TWINE is another algorithm that applied the table-based nibble permutation that achieved hardware efficiency (Suzaki et al., 2011). The adopted table-based permutation in lightweight algorithms are presented in Table 1.

Table 2: Types of Feistel-based Permutation

Type	Data/Block (bit)	Data Distribution Ratio (%)	Algorithm
Bit	16/32	50.00%	JAC_Jo, LBC-IoT, and MISTY
	16/64	25.00%	μ2, LILLIPUT, NUCLEAR, NUX, and QTL
	32/64	50.00%	Blowfish, LBlock, MIBS, MISTY, and VAYU
Nibble	4/16	25.00%	Khudra

2.2 Feistel-based Permutation

In Feistel network structure algorithm, the plaintext block is split into two or many equal sizes of cipher which is known as halves. These halves will be swapped or permutated into a different cipher position before operating other cryptographic functions. In this research, only the permutation function will be considered in the analysis without including the other functions in the Feistel network structure such as F-function and XOR operation. This analysis aims to evaluate the effect of different types of Feistel-based permutation functions used in lightweight block ciphers.

Implementations of the permutation technique are divided into block and nibble types. Block type permutation layer enhanced diffusion property of LBC-IoT by applying two blocks swap of the Feistel structure (Ramadan et al., 2021). In VAYU, the block permutation layer increased its active S-boxes (Bansod et al., 2016). The block permutation in the Feistel design of MIBS used simple wiring that does not require an extra gate (Izadi et al., 2009). Blowfish Feistel network discards the bitwise permutation to introduce simple operations that are efficient on microprocessors (Schneier et al., 1993).

Apart from that, the nibble permutation of Feistel structure in Khudra reduced the number of S-box usage (Kolay and Mukhopadhyay, 2014). Although the S-box usage is reduced, the block cipher does not require an additional diffusion layer to secure the algorithm. The Feistel-based permutation can be applied using multiple methods in lightweight block ciphers by splitting the plaintext block into 4, 16, or 32 bits as shown in Table 2.

Table 3: Types of Formulation-based Permutation

Type	Data/Block (bit)	Data Distribution Ratio (%)	Algorithm
Bit	1/16	6.25%	μ2
	1/32	3.13%	ESF and TED
Nibble	4/16	25.00%	LRBC
Byte	8/64	12.50%	ILEA

2.3 Formulation-based Permutation

Formulation-based permutation executes on-the-fly mathematical computation to obtain the permutation parameters. Mathematical operations such as addition, subtraction, multiplication, and modulation are used to compute the permutation parameters before executing the permutation operation. Formulation-based bit permutation provides fast diffusion without hardware implementation cost in ESF algorithm (Liu et al., 2014). The bitwise formulation-based permutation is designed to maximize the number of active S-boxes of μ2 block cipher (Yeoh et al., 2020).

Besides the formulation-based bit permutation method, ILEA (Jha et al., 2020) applied formulation-based byte permutation while LRBC block cipher (Biswas et al., 2020) implemented round transposition operation using formulation-based nibble permutation to provide additional security to the algorithms. Implementations of formulation-based permutations in existing lightweight block ciphers are listed in Table 3.

3 ANALYSIS OF PERMUTATION FUNCTION

This section presents the analysis used to evaluate permutation functions used in lightweight block ciphers. The evaluation techniques are implemented to distinguish the cryptographic strength of each method. Two types of tests are conducted including the avalanche effect and correlation coefficient tests.

For the experimental setup, the input data is assigned as bit representation defined as x_0, x_1, ..., x_n, while the output data is defined as y_0, y_1, ..., y_n,

where n is the size of the permutation function. In this experiment, we analyzed 64 bits cipher blocks. Therefore, permutation functions that have smaller than 64-bit size as shown in Table 1, Table 2, and Table 3 are repeated using different sets of input data to complete the specified cipher block size. To conduct the analysis, the appended input data is defined as p_0, p_1, ..., p_{63}, while the appended output data is defined as p_0, p_1, ..., p_{63} are analyzed using the evaluation methodologies that will be discussed in the following sections.

The objective of this research is to inspect the effectiveness of each permutation function in distributing the cipher bits. Therefore only a single function round is conducted in this analysis instead of executing the full encryption rounds of the lightweight block ciphers. From there, readers can distinguish the strength of the individual permutation function in order to design a lightweight block cipher that has an optimum number of encryption rounds.

3.1 Avalanche Effect Test

Permutation function of a lightweight block cipher can offer diffusion property to the algorithm (Zakaria et al., 2020). This security feature is dependent on each output produced from the cipher input. Avalanche effect can be used to measure and analyse the non-linear characteristics of a lightweight block cipher. In order to execute the avalanche effect test, one hundred 64-bit plaintexts are generated as the input using a pseudo-random bit generator. Each input is fed into the permutation methods to obtain the output and the avalanche effect is computed using equation (1) (Astuti et al., 2019). The optimum value for the avalanche effect result should be 50% of the total number of the cipher bits.

$$E = \frac{1}{s} \sum_{i=1}^{s} |c_i - p_i| \tag{1}$$

where s is the length of input while p_i and c_i are the i^{th} input and output bit.

Table 4: Correlation Coefficient Test Results Indication

Condition	Result
$r_{pc} = 0$	Non-linear relationship
$0 < r_{pc} \leq 0.3$ or $-0.3 \leq r_{pc} < 0$	Weak positive/negative linear relationship
$0.3 < r_{pc} < 0.7$ or $-0.7 < r_{pc} < -0.3$	Moderate positive/negative linear relationship
$0.7 \leq r_{pc} < 1$ or $-1 < r_{pc} \leq -0.7$	Strong positive/negative linear relationship
$r_{pc} = 1$ and $r_{pc} = -1$	Perfect positive/negative linear relationship

3.2 Correlation Coefficient Test

Correlation coefficient test aims to analyse the non-linear association of the permutation function input and its corresponding output (Imdad et al., 2022). Formulation of the correlation coefficient is given by equation (2). The coefficient takes values from +1 to -1 where the accepted ranges of the results are listed in Table 4 (Kanjo et al., 2017). Similar to the avalanche effect test, one hundred 64-bit plaintexts are generated as the input to execute the experiment.

$$r_{pc} = \frac{\sum_{i=1}^{s}(p_i - E)(c_i - E)}{\sqrt{\sum_{i=1}^{s}(p_i - E)^2}\sqrt{\sum_{i=1}^{s}(c_i - E)^2}} \tag{2}$$

where E is the avalanche effect while p_i and c_i are the i^{th} input and output bit.

4 RESULTS AND DISCUSSION

This section presents the results of the analysis conducted on the permutation functions implemented in lightweight block ciphers. Comparison of permutation methods that include the table-based permutation, Feistel-based permutation, and formulation-based permutation are briefly discussed to observe the impact of each method on the strength of the cryptographic algorithm.

4.1 Results of Table-based Permutation

There are six table-based permutations extracted from existing algorithms to conduct the avalanche effect and correlation coefficient tests that include ACT (Table-1/64), ANU (Table-1/32), TWINE (Table-4/64), HERMES (Table-1/16), DoT (Table-1/8), and MIBS (Table-4/32) algorithms. From the avalanche effect results shown in Figure 2, all tables except ACT obtained the optimum results which are 32 bits or equivalent to 50% of the total cipher size. Meanwhile, for the correlation coefficient, all results achieved lower than 0.3 which indicates a weak linear relationship between the cipher input and output.

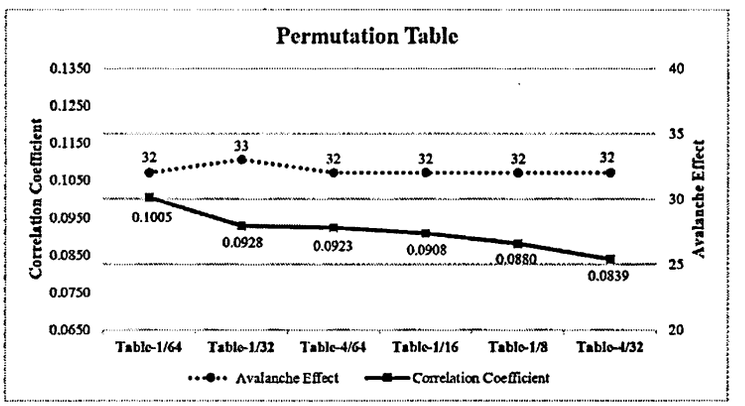


Figure 2: Results of Table-based Permutation Functions

4.2 Results of Feistel-based Permutation

Feistel-based permutation implemented four methods such as Feistel-4/16 (Khudra), Feistel-16/64 (LILLIPUT), Feistel-16/32 (MISTY), and Feistel-32/64 (LBlock) as presented in Figure 3. The avalanche effect shows that only Khudra and LILLIPUT achieved optimum results, while the other algorithms obtained acceptable results which deviate slightly from the threshold value of 32 bits. On the other hand, the correlation coefficient test recorded all types of Feistel-based permutation obtained the weak linear relationship indicator which is 0.3.

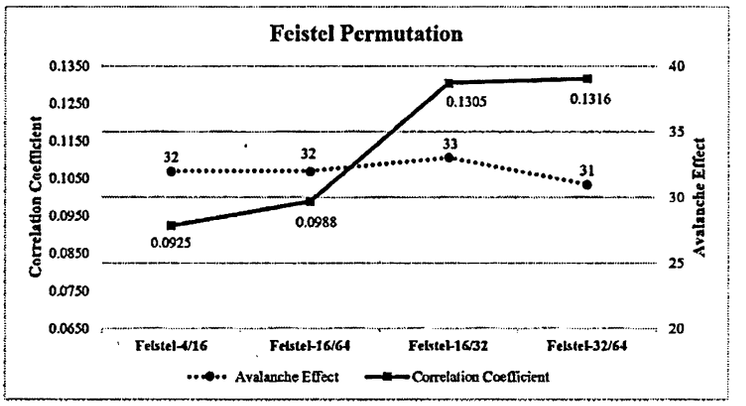


Figure 3: Results of Feistel-based Permutation Functions

4.3 Results of Formulation-based Permutation

Formulation-based permutation method consists of four implementations that include Formulation-1/32 (ESF), Formulation-1/16 (μ2), Formulation-8/64 (IL-EA), and Formulation-4/16 (LRBC) as displayed in Figure 4. Similar to the table-based permutation and Feistel-based permutation, all methods obtained close to the optimum avalanche effect results which are 32 bits and achieved less than 0.3 correlation coefficient results which indicate a weak linear relationship of the cipher.

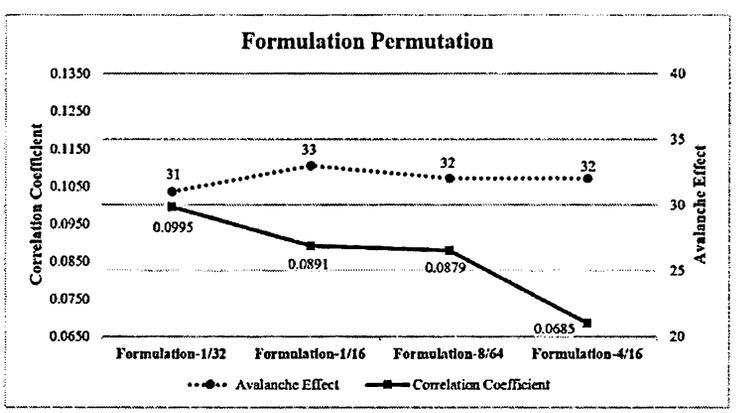


Figure 4: Results of Formulation-based Permutation Functions

4.4 Comparison of Permutation Methods

Results of permutation methods are sorted according to the data length which indicates the data being processed that can be represented in the form of bit, nibble, byte, or block as shown in Figure 5. By only considering the table-based permutation and formulation-based permutation using 1-bit data, the correlation coefficient increased as the block sizes increased. Looking at the 16-bit data, increasing the block size would decrease the correlation coefficient of the Feistel permutation. Contrary to the mentioned results, the correlation coefficient of the table-based permutation in 4-bit data increased in line with the block sizes.

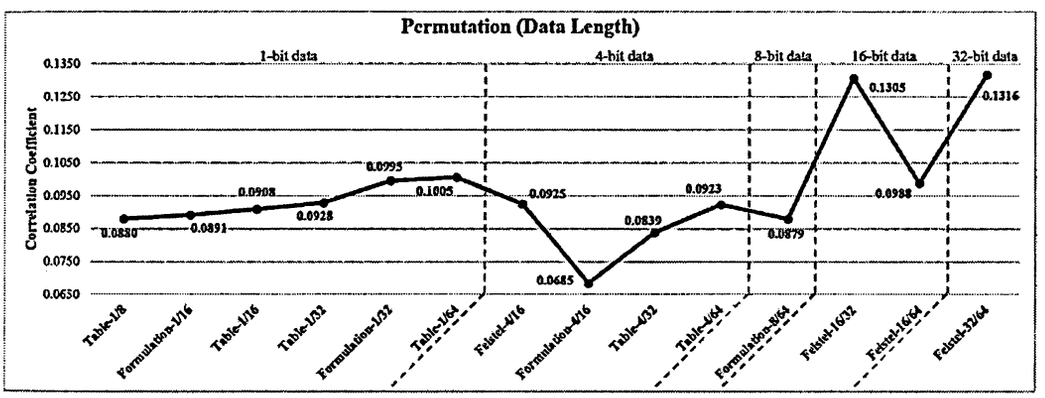


Figure 5: Results of Permutation Functions According to Data Length

From another angle, the results are sorted according to the block length as shown in Figure 6. For the 32-bit block, the table-based permutation achieved a better result than the formulation-based permutation. Meanwhile, increasing the data size would reduce the correlation coefficient of the table-based permutation. In the 16-bit block, the formulation-based permutations obtained better results than the table-based permutation. On the other hand, the formulation-based permutation attained a better correlation coefficient compared to the Feistel-based permutation. Increasing the data size of the permutation table in the 64-bit block has increased the result while increasing the data size of Feistel-based permutation has reduced the correlation coefficient.

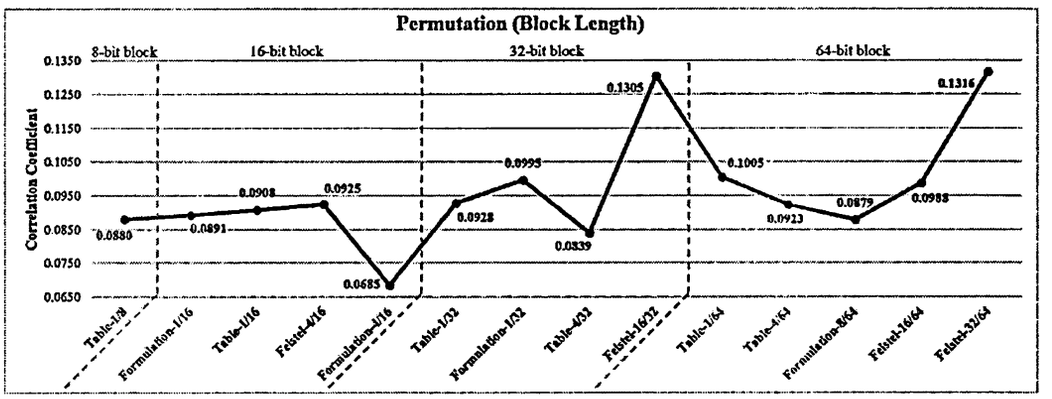


Figure 6: Results of Permutation Functions According to Block Length

4.5 Discussion

The overall avalanche effect results achieved by all three permutation methods prove their ability to provide diffusion property to the cipher output. Although some of the results are slightly deviate from the threshold value of 32 bits, it can be concluded that the permutation methods have the ability to maintain the optimum output changes with minor or major modifications of the input.

On the other hand, the highest correlation coefficient result is 0.1316 which indicates a weak linear relationship between the cipher input and its corresponding output. The results show that the output from the permutation method is not linear with the input data. This is a required criterion in designing a lightweight block cipher to prevent successful cryptanalysis attacks.

There are a few factors that influenced the avalanche effect and correlation coefficient. For the table-based permutation, components of the analyzed tables are uniformly distributed in which no component is repeated. The table components ensure no similar output is produced throughout the permutation process. Positions of the components also give impact to the generated output.

Selection of the Feistel-based permutation type contributes to the results of the analysis. The type of data being processed during the permutation process determined the effectiveness of the permutation method for lightweight block cipher implementation.

In formulation-based permutation, the method is dependent on the mathematical equation to determine the output of the function. Similar to the other two methods, the formulation-based permutation aims to generate good cipher bits distribution. Hence, proper analysis of the formulation is required to ensure that the mathematical equation can provide security to the function.

Another finding obtained from the results is the effect of the data and block sizes of the cipher which can be represented by the distribution ratio. For table-based permutation and formulation-based permutation, the correlation coefficient is decreased when the distribution ratios are increased. However, the correlation coefficient of Feistel-based permutation increased linearly with the distribution ratio because it does not distribute its block in a random manner but splits it into multiple blocks according to the dedicated position. Therefore, the more blocks are split, the better the correlation coefficient result.

Lastly, it can be concluded that there is no clear-cut winner between the three competing permutation methods since their strength cannot be distinguished from one another. However, all of the permutation methods achieved good avalanche effect and correlation coefficient results that can improve the diffusion property of lightweight block ciphers.

5 CONCLUSION

This paper provided an analysis of permutation functions implemented in existing algorithms. From the analysis conducted on three permutation methods, it is observed that there are factors that influenced their cryptographic strength that include the data and block sizes of the cipher. Apart from that, components of the table-based permutation, types of Feistel-based permutation, and mathematical equation of the formulation-based permutation play an important role in determining the performance of the permutation methods. Permutation functions offer benefits for lightweight block cipher adoption due to their excellent characteristics, especially for software and hardware implementations. Although the strength of each permutation function is not similar, every function has its advantages in increasing the security of lightweight algorithms.

REFERENCES

Lai, X. and Massey, J. L. (1991). A proposal for a new block encryption standard. *In Workshop on the Theory and Application of Cryptographic Techniques*, 389–404. Springer, Berlin, Heidelberg.

Schneier, B. (1993). Description of a new variable-length key, 64-bit block cipher (Blowfish). *In International Workshop on Fast Software Encryption*, 191–204. Springer, Berlin, Heidelberg.

Wheeler, D. J. and Needham, R. M. (1994). TEA, a tiny encryption algorithm. *In International Workshop on Fast Software Encryption*, 363–366. Springer, Berlin, Heidelberg.

Feng, J. and Li, L. (2022). SCENERY: A lightweight block cipher based on Feistel structure. *Frontiers of Computer Science*, 16: 1–10.

Nayak, M. K. and Swain, P. K. (2022). ESIT: An Enhanced Lightweight Algorithm for Secure Internet of Things. *In Lecture Notes in Networks and Systems*, 107–116. Springer, Singapore.

Girija, M., Manickam, P., and Ramaswami, M. (2020). PriPresent: An embedded prime lightweight block cipher for smart devices. *Peer-to-Peer Networking and Applications*, 14: 1–11.

Liu, B. T., Li, L., Wu, R. X., Xie, M. M., and Li, Q. P. (2019). Loong: A family of involutional lightweight block cipher based on SPN structure. *IEEE Access*, 7: 136023–136035.

Singh, P., Acharya, B., and Chaurasiya, R. K. (2019). A comparative survey on lightweight block ciphers for resource constrained applications. *International Journal of High Performance Systems Architecture*, 8: 250–270.

Sehrawat, D., Gill, N. S., and Devi, M. (2019). Comparative analysis of lightweight block ciphers in IoT-enabled smart environment. *In 6th International Conference on Signal Processing and Integrated Networks*, 915–920. IEEE.

Zakaria, A. A., Azni, A. H., Ridzuan, F., Zakaria, N. H., and Daud, M. (2020). Modifications of Key Schedule Algorithm on RECTANGLE Block Cipher. *In International Conference on Advances in Cyber Security*, 194-206.

Turan, M. S., McKay, K. A., Çalık, Ç., Chang, D., and Bassham, L. (2021). Status report on the first round of the NIST lightweight cryptography standardization process. *NIST Internal or Interagency Report (NISTIR)*, 8369.

Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J. B., Seurin, Y., and Vikkelsoe, C. (2007). PRESENT: An ultra-lightweight block cipher. *In International Workshop on Cryptographic Hardware and Embedded Systems*, 450–466. Springer-Verlag.

Bansod, G. (2016). A new ultra lightweight encryption design for security at node level. *International Journal of Security and its Applications*, 10: 111–128.

Jithendra, K. B. and Kassim, S. T. (2020). ACT: An ultra-light weight block cipher for internet of things. *International Journal of Computing and Digital Systems*, 9: 921–929.

Li, L., Liu, B., Zhou, Y., and Zou, Y. (2018). SFN: A new lightweight block cipher. *Microprocessors and Microsystems*, 60: 138–150.

Standaert, F. X., Piret, G., Rouvroy, G., Quisquater, J. J., and Legat, J. D. (2004). ICEBERG: An involutional cipher efficient for block encryption in reconfigurable hardware. *In International Workshop on Fast Software Encryption*, 279–298. Springer, Berlin, Heidelberg.

Das, S. (2014). Halka: A lightweight, software friendly block cipher using ultra-lightweight 8-bit S-box. *IACR Cryptology ePrint Archive*, 1–16.

Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., and Regazzoni, F. (2015). Midori: A block cipher for low energy. *In International Conference on the Theory and Application of Cryptology and Information Security*, 411–436. Springer, Berlin, Heidelberg.

Suzaki, T., Minematsu, K., Morioka, S., and Kobayashi, E. (2011). Twine: A lightweight, versatile block cipher. *In ECRYPT Workshop on Lightweight Cryptography*, 146–169. Springer, Berlin, Heidelberg.

Ramadan, R. A., Aboshosha, B. W., Yadav, K., Alseadoon, I. M., Kashout, M. J., and Elhoseny, M. (2021). LBC-IoT: Lightweight block cipher for IoT constraint devices. *Computers, Materials and Continua*, 67: 3563–3579.

Izadi, M., Sadeghiyan, B., Sadeghian, S. S., and Khanooki, H. A. (2009). MIBS: A new lightweight block cipher. *In International Conference on Cryptology and Network Security*, 334–348. Springer, Berlin, Heidelberg.

Kolay, S. and Mukhopadhyay, D. (2014). Khudra: A new lightweight block cipher for FPGAs. *In International Conference on Security, Privacy, and Applied Cryptography Engineering*, 126–145. Springer, Cham.

Liu, X., Zhang, W. Y., Liu, X. Z., and Liu, F. (2014). Eight-sided fortress: A lightweight block cipher. *Journal of China Universities of Posts and Telecommunications*, 21: 104–128.

Yeoh, W. Z., Teh, J. S., and Sazali, M. I. S. B. M. (2020). μ2: A Lightweight block cipher. *In Computational Science and TechnologyE*, 281–290. Springer, Singapore.

Jha, P., Zorkta, H. Y., Allawi, D., and Al-Nakkar, M. R. (2020). Improved lightweight encryption algorithm (ILEA). *In International Conference for Emerging Technology*, 1–4. IEEE.

Biswas, A., Majumdar, A., Nath, S., Dutta, A., and Baishnab, K. L. (2020). LRBC: A lightweight block cipher design for resource constrained IoT devices. *Journal of Ambient Intelligence and Humanized Computing*, 1–15.

Zakaria, A. A., Azni, A. H., Ridzuan, F., Zakaria, N. H., and Daud, M. (2020). Extended RECTANGLE algorithm using 3D bit rotation to propose a new lightweight block cipher for IoT. *IEEE Access*, 8: 198646–198658.

Astuti, N. R. D. P., Arfiani, I., and Aribowo, E. (2019). Analysis of the security level of modified CBC algorithm cryptography using avalanche effect. *In IOP Conference Series: Materials Science and Engineering*, 1–8.

Imdad, M., Ramli, S. N., and Mahdin, H. (2022). An enhanced key schedule algorithm of PRESENT-128 block cipher for random and non-random secret keys. *Symmetry*, 14: 1–22.

Kanjo, E., Kuss, D. J., and Ang, C.S. (2017). NotiMind: Utilizing responses to smart phone notifications as affective sensors. *IEEE Access*, 5: 22023–22035.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

Authentication Methods that use Haptic and Audio : A Review

Yvonne Hwei-Syn Kam[*1] and **Ji-Jian Chin**[2]

[1]*Faculty of Engineering, Multimedia University, Selangor, Malaysia*
[2]*Faculty of Computing & Informatics, Multimedia University, Selangor, Malaysia*

E-mail: hskam@mmu.edu.my
[*]*Corresponding author*

ABSTRACT

Authentication is an important step for many applications eg. banking, communications etc. Identity theft can happen because the systems that we use are vulnerable to observation attacks both by a person (shoulder surfing) and also video recordings. Thus authentication methods that rely on the visual channel alone, which are displayed onscreen are more susceptible to shoulder surfing. In this paper, a review of haptic and audio authentication methods is presented where we describe and analyse various methods that utilise non-visual channels, namely haptic and audio, for authentication. We categorise the methods into the categories of recall, recognition and challenge-response. We focus on the security aspects of the methods, comparing the susceptibility of the methods to various attacks. The review indicates that most haptic and audio methods are less susceptible to observation attacks and intersection attacks.[1]

Keywords: Haptic, audio, vibration, authentication, review

[1]A preliminary version of this work was presented at the International Conference on Software Engineering, Knowledge Engineering (SEKEIE 2015) but not published. This is an extended version with updated analysis of recent works.

Yvonne Hwei-Syn Kam & Ji-Jian Chin

1 INTRODUCTION

1.1 Motivation

In the year 2020 in the United Kingdom, card identity theft in ATM machine fraud accounted for £29.7 million losses (Finance (2021)). Graphical passwords which do not reveal the PIN have been touted as a possible alternative (Moncur and Leplâtre (2007)) to entering PINs. However, shoulder surfing is a large threat as interactions are still visible. Since login procedures are fixed, some graphical passwords may able to be attacked after a number of observations by reverse engineering the user's response.

A solution proposed is using haptic or audio channel for authentication, either standalone or in addition to the visual channel, because haptic or audio channels are less easily observed or perceived by attackers. The methods are categorised as Haptic-based or audio-based because of the channel used to send or receive the challenge (Binbeshr et al. (2020)). However, such methods are not immune to attacks either. Past methods have been broken because there are information leakages which may leave them susceptible to observation, intersection and side channel attacks.

This work is a survey of the various haptic and audio methods in literature. We break down and categorise them accordingly in Section 2. Section 3 looks at the security attacks against haptic and audio methods. Section 4 summarises them in a table. Finally, we provide concluding remarks in Section 5.

2 REVIEW OF PAST WORKS

Observation attacks are those which the attacker obtains information by looking at the password entry of a valid user. This can be achieved by means of physically looking over the user's shoulder, also known as shoulder surfing or recording the user's login sessions for analysis, which is a stronger attack.

Often, graphical password systems try to avoid this by obfuscating infor-

mation through not selecting passwords directly, or asking the user to perform a mental task. However this increases the mental strain on the user.

Thus one of the solutions is to use other channels of information rather than just visual. This leads to haptic and audio based authentication systems. These are often hybrid methods with haptic or audio feedback together with a visual display. There are also a few which do not have the visual element altogether.

In this paper, we survey the authentication methods that incorporate the nonvisual channels namely haptic and audio. Haptic and audio methods generally fall into either one of three categories which are recall, recognition and challenge-response.

2.1 Recall

Recall refers to a task where information is extracted from memory when requested (Renaud et al., 2013). Examples of this kind of system are traditional password authentication and PINs, or where users are tasked to reproduce a secret drawing (Biddle et al., 2012).

An example of a secret drawing system is Draw-a-secret (DAS) (Jermyn et al., 1999) which was the an early recall-based password system. This was a purely visual based system where users drew their password using a mouse or stylus. The main weakness of recall based systems is that they are susceptible to observation attacks, which is an attack based on looking at (shoulder surfing) or recording the entire login of the user.

Various improvements were made to the DAS methods. Perković et al. (2010) proposed penups in addition to putting a background picture and increasing neighbouring connectivity to make it harder for shoulder surfing to occur.

Then, haptic input was added into a DAS-like scheme (Orozco et al., 2006), (Malek et al., 2006) where pressure information was the additional input. This was for the purpose of obfuscating input from shoulder surfers. However,

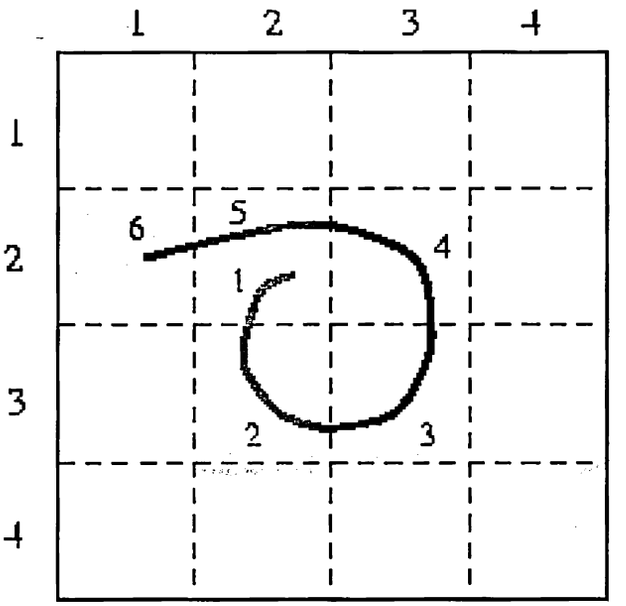


Figure 1: Draw-a-secret (Jermyn et al., 1999).

from the authors' experiment, most users did not choose to use this feature thus forfeiting the purported resistance to observation attacks. As the drawing is observable, drawing-based recall methods are not secure against observation attacks. Also under the recall methods category, are counting techniques, which count pulses up to each PIN digit. Examples are SpinLock, ColorLock and TimeLock (Bianchi et al., 2012). In SpinLock, users have to turn the dial in a particular direction until they hear the number of haptic or audio cues that correspond to their PIN digit whereas in ColorLock and TimeLock, cues are counted while pressing a button. In these schemes, users reported that the randomly distributed timing of the haptic/audio challenges made it difficult to predict when to stop spinning or holding the button. As such, the most common error was overshooting or undershooting the target by one.

2.2 Recognition

Recognition schemes are where users have to memorise a set of items and then recognise those same items when authenticating later. These include im-

ages, tactons or earcons. Tactons and earcons are structured tactile and audio messages respectively (Brewster and King, 2005). The haptic and audio methods under this category include Tactile authentication system (Kuber and Yu, 2010). The Tactile authentication scheme featured the use of a VT player mouse which can generate different tactons under the fingers of the user.

Bianchi et al. (2010c) came up with Secure Haptic Keypad and also the Haptic Wheel which used purpose built hardware to give tactile challenges to be recognised by the user. The haptic wheel is in the form of a dial. Subsequently, Phone Lock by Bianchi et al. (2010a) (similar design to Haptic Wheel) was implemented on a mobile phone software interface with tactile and audio challenges.

The time taken for login was longer for these schemes compared to graphical password schemes because the users had to take more time to detect and recognise the PIN digits presented via audio or haptic cues, which is slower compared to detecting a visual display of numbers and images.

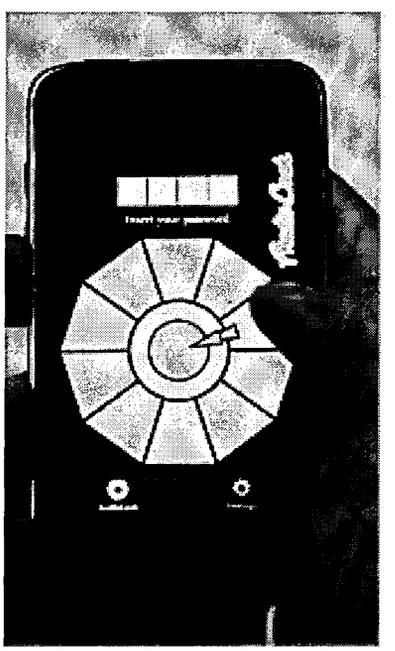


Figure 2: Phone Lock (Bianchi et al., 2011).

2.3 Challenge-response

Challenge based haptic and audios are those that present challenges via the haptic or audio channel to direct the user in interacting with the authentication system. Challenges are often combined with simple recognition based methods such as password and PIN entry systems. An example of a challenge is haptic or audio output from the system to guide the user to enter a particular input. For example, in Vibrapass by De Luca et al. (2009), vibration output is used to indicate to the user the positions within the PIN to insert "lies" or extraneous input. Other haptic-based methods include (Chakraborty et al. (2016); Kabir et al. (2020);Kuribara et al. (2014)).

Undercover by Sasamoto et al. (2008) is a challenge-response scheme which used graphical recognition together with haptic or audio challenges. The system issued challenges which have a visible component and a hidden component, called visible and hidden challenges respectively, to prompt the user to use a particular button layout to enter a digit. In an alternative Undercover design by Hasegawa et al. (2009), from the hidden challenges issued, the user responds to find the path from the challenge number to their PIN digit. Perković et al. (2011) enhanced the design of Undercover to solve side channel leakages contributed by the display layout in the previous two versions.

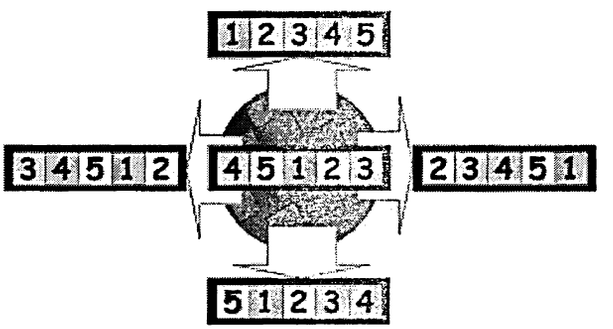


Figure 3: Undercover Sasamoto et al. (2008).

The SpinPad method (Rajarajan et al. (2018)) is an audio-based method. In each round, an audio challenge which is a spoken alphabet letter is conveyed via earphones to the user. The user aligns an onscreen digit to that challenge. Audio-PES by Dan and Ku (2017) uses the phone receiver instead of requiring

earphones as the challenge is purely in audio form without any visible challenge.

3 ATTACKS

In this section, we attempt to analyse the susceptibility of haptic and audio methods to various attacks. As haptic and audio methods were mostly developed with the aim to reduce susceptibility to visual observation (shoulder surfing) attack, we chose to analyse the susceptibility of these methods against observation attack and a closely related attack: intersection attack for which multiple observations are utilised. Susceptibility to brute force attack was also included to compare the password spaces.

3.1 Brute force Attack

Brute force attack (or exhaustive-search attack) tries every element in a search space, whether from graphical or text passwords in order to find the right combination. To prevent brute force attack, it is necessary to have a large enough password space. A 4-digit PIN has a password space of 10000. Tactile authentication system (Kuber and Yu, 2010) has a smaller password space of 6561.

Passwords that are drawings theoretically have unlimited password space. However in practice users limit their number of strokes because of the high memory cost. For example, in the Haptic-based graphical password (Malek et al. (2006)), users did not use the same line more than twice and did not tend to cross a point more than three times.

The larger the number of PIN digits, the higher the password space. For example, the 8-digit PINs in the case of (Ku and Xu (2019)) produces a password space of 100,000,000. However, that also increases the memory burden correspondingly.

3.2 Observation Attack

As mentioned, an observation attack is an attack based on observing (shoulder surfing) or recording the login sessions of the user. In this respect, recall based systems which involve drawing patterns are not secure against observation. With a pressure element (Orozco et al. (2006)), there is an increased resistance. However, even with one observation, the basic drawing without pressure elements is revealed.

In methods that have a graphical password element, such as ColorLock (Bianchi et al., 2012), the password space will be reduced after one observation of the button colors pressed. However, for Undercover, even though there is a graphical password, the images are not selected directly. The use of hidden challenges is able to resist the discovery of the password.

In general, because haptic and audio methods use non-visual elements which are not shown on screen, they are more secure against observation attack. To mitigate observation or recording of non-visual elements i.e. vibrations and audio, some methods have attempted to mitigate this with low noise vibration capability of phones. Ambient noise can also obscure the vibrations. Earphones are usually the method to obscure audio eavesdropping. A caveat is, this needs the user to have this additional hardware available which could be inconvenient.

Under the column, "After 1 observation" in Table 1, the effect of one observation of the entire login action and/or screen display on the various methods is shown. No security indicates that the attack is successful to reveal the credential as in the case of the standard PIN and also the DAS. Otherwise, some information is revealed, as in the case of drawing-based methods (Malek et al. (2006), Mat Kiah and Por (2010)).

In ColorLock (Bianchi et al. (2012)), a part of the secret (colour) is revealed, thus after one observation, the password space is reduced to 625. In the case of HapticLock (Dhandapani et al. (2021)), the authentication mobile app cycles through digits 0-9 and indicates the current digit via Morse-code vibration feedback. The user has to keep track of the cycling numbers to select their PIN digit. There are two ways to configure the start digit (zero or ran-

dom). In the "starting from zero" configuration, the first digit is always zero. Thus the PIN digit being entered can be deduced by counting the previous swipes. The "random digit configuration" starts at a random digit and is thus resistant to counting.

In the Vibrapass method (De Luca et al. (2009)), the PIN is inserted with "lies". The user enters the PIN digits interspersed with the lie digits. Thus 1 observation reveals the PIN digits plus one or more lies.

VpointsPES (Ku and Xu (2019)) has the user aligning their PIN digits with the letter in the challenge. Thus after one observation of the Low SA resistance & High efficiency configuration, the password space is reduced.

3.3 Intersection Attack

In a challenge-response authentication, the response varies according to the challenge presented to the user. When a response reveals some information about the secret then the secret can be reconstructed after observing a number of logins. Intersection attacks involve pooling leaked information gathered from observing or recording several logins for schemes in which the authentication response varies across login instances (Biddle et al., 2012).

Since more observations of different login instances progressively reveal more information about the secret, we use this criteria to ascertain whether an authentication method is susceptible. Intersection attack is possible if there are observable responses that vary across login instances. In audio and haptic-based authentication, the response varies according to the challenge. Nevertheless, most audio and haptic methods are resistant to observation attacks and by extension, intersection attacks. Some however, leak some information across logins and are susceptible. The metric shows a concise evaluation of whether a method was susceptible to intersection attacks.

For methods where the observable response does not vary in a quantifiable manner, then this attack is considered not applicable. For example, for drawing-based methods (Jermyn et al. (1999); Mat Kiah and Por (2010);Malek et al. (2006)), one observation reveals the basic drawing and subsequent obser-

vations do not yield responses where the difference is noticeable.

Perković et al. (2011) showed that the passwords of Undercover (Sasamoto et al. (2008) and Alternative Undercover (Hasegawa et al. (2009)) could be exposed with high probability, with only O(10) observed login sessions. They showed that certain combinations of digits could be eliminated by observing the input pattern of the user.

Vibrapass (De Luca et al., 2009) has this weakness where more than one observation will reduce the password space. This is because intersection of more than one observation of Vibrapass(De Luca et al. (2009)) can identify which digits are true and which digits are lies. At the smallest lie overhead, 2 recordings can lead to breaking the four-digit PIN.

In the case of VibraInput (Kuribara et al. (2014), in a 4-digit PIN entry, the attacker can obtain one digit with a probability of 0.213 and the 4-digit PIN with a probability of 0.002.

The candidate PIN set of VpointsPES (Ku and Xu, 2019) reduces with every observation via intersection analysis. Thus the PIN candidates can be progressively reduced through the analysis of multiple logins. After one observation, the probability of obtaining the password increases to 10^{-2}.

4 SUMMARY OF METHODS SUSCEPTIBILITY TO ATTACKS

Table 1 summarizes the susceptibility of different methods to the security attacks discussed in the preceding section.

5 CONCLUSION

The various haptic and audio methods in literature have been explored and analysed. A comparison of these haptic and audio methods has thus been pre-

Category	Method	Susceptibility to Attacks		
		Brute force (Probability)	After 1 Observation	Intersection
Recall	Standard PIN with keypad (4-digit)	$1/10^4 = 1/10000$	No security	Yes
	DAS (Jermyn et al., 1999)	Theoretically unlimited	No security	NA
	Background Pass-Go (Mat Kiah and Por, 2010)	Theoretically unlimited	Basic drawing revealed	NA
	Haptic-based graphical password (Malek et al., 2006)	Theoretically unlimited	Basic drawing revealed	NA
	Spin Lock Bianchi et al. (2012)	$1/10^4 = 1/10000$	No	No
	ColorLock Bianchi et al. (2012)	$1/20^4 = 1/160000$	1/625	No
	TimeLock Bianchi et al. (2012)	$1/5^4 * 4! = 1/15000$	1/625	No
	Audio-PES (Dan and Ku, 2017)	$1/10^6 = 1/1000000$ (6 digit PIN)	No	No
	(Kabir et al., 2020)	$1/(5 * 10)4 = 1/6250000$	$1/5^4 = 1/625$	No
Recognition	Tactile authentication system (Kuber and Yu, 2010)	$1/9^4 = 1/6561$	No	No
	Secure Haptic Keypad (Bianchi et al., 2010b)	$1/3^9 = 1/19863$	No	No
	Haptic Wheel (Bianchi et al., 2010c)	$1/5^6 = 1/15625$, $1/3^9 = 1/19683$	No	No
	Phone Lock (Bianchi et al., 2010a)	$1/10^4 = 1/10000$	No	No
	HapticLock (Dhandapani et al., 2021) -starting at zero -random digit	$1/10^4 = 1/10000$	Yes No	No No
Challenge-response	Undercover (Sasamoto et al., 2008)	$1/({}^7C_2 * 4^5) = 1/20480$	No	Yes*
	Vibrapass (De Luca et al., 2009)	$1/10^4 = 1/10000$ (4 digit PIN)	Yes	Yes
	Alternative Undercover (Hasegawa et al., 2009)	$1/(10x10)^4 = 1/10^8$	No	Yes
	Enhanced Undercover (Perković et al., 2011)	$1/({}^7C_2 * 4^5) = 1/20480$	No	No
	VibraInput (wheel type) (Kuribara et al., 2014)	$1/10^4 = 1/10000$ (4 digit PIN)	No	Yes
	VDLS (Chakraborty et al., 2016)	$1/9^4 = 1/6561$	No	No
	SpinPad (Rajarajan et al., 2018)	$1/10^4 = 1/10000$ (4 digit PIN)	No	No
	VpointsPES (Ku and Xu, 2019) (Low SA resistance & High efficiency)	$1/10^8 = 1/100000000$ (8-digit PIN)	$1/10^2$	Yes

*If the images are randomized (Perković et al., 2011)

Table 1: Susceptibility to Various Attacks.

sented. Generally, where preventing observation attacks is paramount, haptic and audio methods have advantages over purely graphical methods. From the review presented, most haptic and audio methods are less susceptible to observation attacks and intersection attacks. Even so, other attacks such as side channel attacks may still be possible. It is thus of importance to design the haptic and audio schemes carefully.

ACKNOWLEDGMENTS

This research was supported by a research grant from the Multimedia University IR Fund [grant number MMUI/210071-IR Fund].

REFERENCES

Bianchi, A., Oakley, I., Kostakos, V., and Kwon, D. S. (2010a). The phone lock: audio and haptic shoulder-surfing resistant PIN entry methods for mobile devices. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*, TEI '11, pages 197–200, New York, NY, USA. Association for Computing Machinery.

Bianchi, A., Oakley, I., and Kwon, D. S. (2010b). The secure haptic keypad: a tactile password system. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1089–1092, New York, NY, USA. Association for Computing Machinery.

Bianchi, A., Oakley, I., and Kwon, D. S. (2012). Counting clicks and beeps: Exploring numerosity based haptic and audio PIN entry. *Interacting with computers*, 24(5):409–422. Publisher: Oxford University Press Oxford, UK.

Bianchi, A., Oakley, I., Lee, J. K., and Kwon, D. S. (2010c). The haptic wheel: design & evaluation of a tactile password system. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 3625–3630, New York, NY, USA. Association for Computing Machinery.

Biddle, R., Chiasson, S., and Van Oorschot, P. (2012). Graphical passwords: Learning from the first twelve years. *ACM Computing Surveys*, 44(4):19:1–19:41.

Binbeshr, F., Kiah, M., Por, L., and Zaidan, A. (2020). A systematic review of pin-entry methods resistant to shoulder-surfing attacks. *Computers & Security*, 101.

Brewster, S. and King, A. (2005). An Investigation into the Use of Tactons to Present Progress Information. In Costabile, M. F. and Paternò, F., editors, *Human-Computer Interaction - INTERACT 2005*, Lecture Notes in Computer Science, pages 6–17, Berlin, Heidelberg. Springer.

Chakraborty, N., Anand, S. V., Randhawa, G. S., and Mondal, S. (2016). On designing leakage-resilient vibration based authentication techniques. In *2016 IEEE Trustcom/BigDataSE/ISPA*, pages 1875–1881. IEEE.

Dan, Y.-X. and Ku, W.-C. (2017). A simple observation attacks resistant PIN-entry scheme employing audios. In *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, pages 1410–1413. ISSN: 2472-8489.

De Luca, A., von Zezschwitz, E., and Hußmann, H. (2009). Vibrapass: secure authentication based on shared lies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 913–916, New York, NY, USA. Association for Computing Machinery.

Dhandapani, G., Ferguson, J., and Freeman, E. (2021). HapticLock: Eyes-Free Authentication for Mobile Devices. In *Proceedings of the 2021 International Conference on Multimodal Interaction*, ICMI '21, pages 195–202, New York, NY, USA. Association for Computing Machinery.

Finance, U. (2021). FRAUD - THE FACTS 2021 THE DEFINITIVE OVERVIEW OF PAYMENT INDUSTRY FRAUD. Technical report.

Hasegawa, M., Christin, N., and Hayashi, E. (2009). New directions in multisensory authentication. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, SOUPS '09, page 1, New York, NY, USA. Association for Computing Machinery.

Jermyn, I., Mayer, A., Monrose, F., Reiter, M. K., and Rubin, A. D. (1999). The design and analysis of graphical passwords. In *Proceedings of the 8th conference on USENIX Security Symposium - Volume 8*, SSYM'99, page 1, USA. USENIX Association.

Kabir, M. M., Hasan, N., Tahmid, M. K. H., Ovi, T. A., and Rozario, V. S. (2020). Enhancing Smartphone Lock Security using Vibration Enabled Randomly Positioned Numbers. In *Proceedings of the International Conference on Computing Advancements*, ICCA 2020, pages 1–7, New York, NY, USA. Association for Computing Machinery.

Ku, W.-C. and Xu, H.-J. (2019). Efficient shoulder surfing resistant pin authentication scheme based on localized tactile feedback. In *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pages 151–156. IEEE.

Kuber, R. and Yu, W. (2010). Feasibility study of tactile-based authentication. *International Journal of Human-Computer Studies*, 68(3):158–181.

Kuribara, T., Shizuki, B., and Tanaka, J. (2014). Vibrainput: two-step PIN entry system based on vibration and visual information. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, pages 2473–2478, New York, NY, USA. Association for Computing Machinery.

Malek, B., Orozco, M., and El Saddik, A. (2006). Novel shoulder-surfing resistant haptic-based graphical password. In *Proc. EuroHaptics*, volume 6, pages 1–6.

Mat Kiah, M. L. and Por, Y. (2010). Shoulder Surfing Resistance Using PENUP Event And Neighboring Connectivity Manipulation. *Malaysian Journal of Computer Science*, 23:121–140.

Moncur, W. and Leplâtre, G. (2007). Pictures at the ATM: exploring the usability of multiple graphical passwords. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 887–894, New York, NY, USA. Association for Computing Machinery.

Orozco, M., Malek, B., Eid, M., and El Saddik, A. (2006). Haptic-based sensible graphical password. In *Proceedings of Virtual Concept*, volume 56, pages 1–4. Citeseer. Issue: 7.

Perković, T., Čagalj, M., and Saxena, N. (2010). Shoulder-surfing safe login in a partially observable attacker model. In *International Conference on Financial Cryptography and Data Security*, pages 351–358. Springer.

Perković, T., Li, S., Mumtaz, A., Khayam, S. A., Javed, Y., and Čagalj, M. (2011). Breaking undercover: exploiting design flaws and nonuniform human behavior. In *Proceedings of the Seventh Symposium on Usable Privacy and Security*, SOUPS '11, pages 1–15, New York, NY, USA. Association for Computing Machinery.

Rajarajan, S., Kalita, R., Gayatri, T., and Priyadarsini, P. (2018). SpinPad: A Secured PIN Number Based User authentication Scheme. In *2018 International Conference on Recent Trends in Advance Computing (ICRTAC)*, pages 53–59.

Renaud, K., Mayer, P., Volkamer, M., and Maguire, J. (2013). Are graphical authentication mechanisms as strong as passwords? In *2013 Federated Conference on Computer Science and Information Systems*, pages 837–844.

Sasamoto, H., Christin, N., and Hayashi, E. (2008). Undercover: authentication usable in front of prying eyes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 183–192.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

AH_{QTR}: A New NTRU Variant based on Quaternion Algebra

Hassan Rashed Yassein[*1], **Amna Hamed Reshan**[2], and **Nadia M.G. Al-Saidi**[3]

[1,2]*Department of Mathematics, College of Education, University of Al-Qadisiyah, Diwaniya, Iraq*
[3]*Department of Applied Science, University of Technology, Baghdad, Iraq*

E-mail: hassan.yaseen@qu.edu.iq
[*]*Corresponding author*

ABSTRACT

NTRU cryptosystem is classified as secure and efficient. However, changing the algebraic structure of the NTRU leads to a new variant cryptosystem. QTRU is one such variant new design based on quaternion algebra. A new mathematical structure is proposed in this paper using the same algebra, which is an alternative to QTRU. It is called AH_{QTR}. The new mathematical construction contributes to making this method more secure compared to some other methods.

Keywords: NTRU, QTRU, AH_{QTR}, quaternion algebra.

1 INTRODUCTION

In 1996, Hoffstein et al. proposed the NTRU public-key cryptosystem based on $Z[x]/(x^N - 1)$ a truncated polynomial ring (Hoffstein et al., 1998). Several researchers have improved the performance of NTRU by developing its

algebraic structure or changing $Z[x]/(x^N - 1)$. In 2002, Gaborit et al. (2002) by replacing the base ring $Z[x]/(x^N - 1)$ of NTRU with a polynomial ring over the field $F_2[x]$ proposed an analogous NTRU cryptosystem called CTRU. In 2005, Coglianese et al. proposed NTRU-like called the MaTRU by replacing $Z[x]/(x^N - 1)$ with all square matrices ring and polynomial entries (Coglianese and Goi, 2005). In 2009, Malekian et al. proposed a multidimensional public-key QTRU depending on quaternion algebra (Malekian et al., 2009). Also, they presented NTRU-like called OTRU cryptosystem based on multidimensional octonion algebra (Malekian and Zakerolhosseini, 2010). In 2013, Jarvis et al. introduced a new structure depending on an Eisenstein ring $Z[w]$, called ETRU (Jarvis and Nevins, 2015).

In 2015, Alsaidi et al. (2015) used commutative quaternion algebra to propose the CQTRU cryptosystem. Then, Yassein et al. improved NTRU by introducing many new variants called HXDTRU depending on hexadecenoic algebra, BITRU depending on binary algebra, BCTRU depending on bi-cartesian algebra, and QOB_{TRU} on carternion algebra (Al-Saidi and Yassein, 2017, Alsaidi and Yassein, 2016, Yassein and Al-Saidi, 2019, 2016, 2017, 2018, Yassein et al., 2022). As well as, Yassein et al. (2020) proposed a new NTRU-like cryptosystem, called NTRTE, with a new structure based on a commutative quaternion algebra. In 2021, Yassein et al. (2021) improved the QTRU public key by using a new mathematical structure. In the same year, Abo-Alsood and Yassein (Abo-Alsood and Yassein, 2021a,b,c, 2022) introduced NTRU-like designs called TOTRU, BOTRU, QOTRU, and QuiTRU depending on Octonion, Bi-octonion, Qu-octonion, and HH-Real algebra, respectively. In the same year, Shahhadi and Yassein (Shahhadi and Yassein, 2021a,b,c, 2022) introduced NTR_{TRN}, NTRS, and NTR_{SH} depending on the tripternion algebra with different mathematical structures for each of them; also, they introduced TrQtNTR depending on the qua-tripternion.

In this work, based on quaternion algebra with a new mathematical structure, we proposed a public-key cryptosystem called AH_{QTR}. Its mathematical structure is demonstrated with its main three faces. Furthermore, a comparison between AH_{QTR}, NTRU, and QTRU in terms of speed and security is accomplished.

2 PRELIMINARIES

Consider the rings $\mathfrak{D} = Z[x]/\left(x^N - 1\right), \mathfrak{D}_p = Z_p[x]/\left(x^N - 1\right)$ and $\mathfrak{D}_q = Z_q[x]/\left(x^N - 1\right)$, and algebras

$$\begin{aligned}
\mathcal{B} &= \{f_0 + f_1 i + f_2 j + f_3 k \mid f_0, f_1, f_2, f_3 \in \mathfrak{D}\}, \\
\mathcal{B}_p &= \{f_0 + f_1 i + f_2 j + f_3 k \mid f_0, f_1, f_2, f_3 \in \mathfrak{D}_p\}, \\
\mathcal{B}_q &= \{f_0 + f_1 i + f_2 j + f_3 k \mid f_0, f_1, f_2, f_3 \in \mathfrak{D}_q\},
\end{aligned}$$

such that, $i^2 = j^2 = k^2 = -1$ and $ij = -ji = k$. As in the NTRU, the QTRU cryptosystem depends on the exact condition of generic parameters N, p, and q. The subsets $\mathcal{L}_{\mathrm{F}}, \mathcal{L}_{\mathrm{G}}, \mathcal{L}_{\delta}$, and $\mathcal{L}_M$ are defined as follows:

$$\begin{aligned}
\mathcal{L}_{\mathrm{F}} &= \left\{ f_0 + f_1 i + f_2 j + f_3 k \in \mathcal{B} \mid f_\alpha \text{ has } \mathcal{L}_{\left(d_{f\alpha}, d_{f\alpha}-1\right)} \right\}, \\
\mathcal{L}_{\mathrm{G}} &= \left\{ g_0 + g_1 i + g_2 j + g_3 k \in \mathcal{B} \mid g_\alpha \text{ has } \mathcal{L}_{\left(d_{g_\alpha}, d_{g_\alpha}\right)} \right\}, \\
\mathcal{L}_{\delta} &= \left\{ \delta_0 + \delta_1 i + \delta_2 j + \delta_3 k \in \mathcal{B} \mid \delta_\alpha \text{ has } \mathcal{L}_{\left(d_{\delta_\alpha}, d_{\delta_\alpha}\right)} \right\}, \\
\mathcal{L}_{\mathrm{M}} &= \{m_0 + m_1 i + m_2 j + m_3 k \in \mathcal{B} | m_\alpha \text{ has coefficients belong to } (-p/2, p/2)\}
\end{aligned}$$

where $\mathcal{L}_{(d_a, d_b)} = \{f \in \mathfrak{D} \mid f \text{ has } d_a \text{ coefficients equal to } 1, d_b \text{ coefficients equal to } -1, \text{the rest equal to } 0\}$.
The QTRU cryptosystem goes through the following phases:

I. Key Generate: For the key generating purpose, we randomly choose $F \in L_F$ and $G \in L_G$ such that F invertible in $\mathcal{B}_p$ and $\mathcal{B}_q$ denote by F_q^{-1}. The key is calculated as follows:

$$H = F_q^{-1} * G \mod q$$

II. Encryption: For the encrypting purpose of the original message $M \in \mathcal{L}_{\mathrm{M}}$, we randomly choose $\delta \in \mathcal{L}_{\delta}$ and then the ciphertext is calculated as follows:

$$E = p\,(H * \delta) + M \mod q$$

III. Decryption: To retrieve the original message from the ciphertext, we perform the following operations:

$$F * E \pmod q = F * (pH * \delta + M) \bmod q$$

$$= (pG * \delta + F * M) \bmod q$$

$$F * E \pmod p = (pG * \delta + F * M) \bmod p$$

$$= (F * M) \bmod p$$

Take, $W = (F * M) \bmod p$

$F_p^{-1} * W = M \bmod p$.

3 THE PROPOSED AH_{QTR}

As in the QTRU cryptosystem, the subsets $\mathcal{L}_F$, $\mathcal{L}_G$, $\mathcal{L}_\delta$, $\mathcal{L}_M$ are used in AH_{QTR} in addition to the subsets $\mathcal{L}_S$ and $\mathcal{L}_R \subset \mathcal{B}$, which are defined as follows:

$$\mathcal{L}_S = \left\{ s_0 + s_1 i + s_2 j + s_3 k \in \mathcal{B} \mid s_\alpha \text{ has } \mathcal{L}_{(d_{s\alpha}, d_{s\alpha})} \right\}$$
$$\mathcal{L}_R = \left\{ r_0 + r_1 i + r_2 j + r_3 k \in \mathcal{B} \mid r_\alpha \text{ has } \mathcal{L}_{(d_{r\alpha}, d_{r\alpha})} \right\}$$

The phases of AH_{QTR} are described as follows.

I. Key Generation: To generate a key, we randomly choose $F \in \mathcal{L}_F, G \in \mathcal{L}_G$, and $S \in \mathcal{L}_S$ such that F invertible in $\mathcal{B}_p$ and $\mathcal{B}_q$ denoted by F_p^{-1} and F_q^{-1} respectively. The key is calculated as follows:

$$H = F_p^{-1} * G * S(\bmod q).$$

II. Encryption: To encrypt the original message $M \in \mathcal{L}_\mathrm{M}$, we randomly choose $\delta \in \mathcal{L}_\delta$ and $R \in \mathcal{L}_R$ and then the ciphertext is calculated as follows:

$$E = p(H * \delta + R) + M \bmod q.$$

III. Decryption: To retrieve the original message from the ciphertext, we perform the following steps:

$$\begin{aligned} F * E(\mathrm{mod} q) &= F * (p(H * \delta + R) + M) \bmod q \\ &= pF * H * \delta + pF * R + F * M \bmod q \\ &= pF * F_q^{-1} * G * S * \delta + pF * R + F * M \bmod q \\ &= pG * S * \delta + pF * R + F * M \bmod q \\ F * E(\mathrm{mod} p) &= pG * S * \delta + pF * R + F * M \bmod p \\ = F * M \bmod p & \end{aligned}$$

Take, $T = (F * M) \bmod p$

$$F_p^{-1} * T = M \bmod p$$

4 COMPARISON BETWEEN AH$_{QTR}$, NTRU, AND QTRU

4.1 Mathematical Analysis

A comparison of the mathematical calculations addition and convolution (Conv.) multiplication of the key generation, encryption, and decryption between AH$_{QTR}$, NTRU, and QTRU are shown in Table 1. We conclude that the computation time of AH$_{QTR}$ is more than NTRU and QTRU.

4.2 Security Space

This kind of analysis is used to determine how many times the private key has been attempted to be recovered. It works by repeatedly transmitting the same message using the same public key. In AH$_{QTR}$, the attacker can recover a significant portion of the message M if the sender sends the same message M several times using the same public keys H, R and varying blinding levels of δ.

Table 1: Mathematical calculations

	AH_{QTR}	NTRU	QTRU
Key Generate	64 Conv. multiplications	one Conv. multiplications	16 Conv. multiplications
Encryption	eight addition, 16 Conv. multiplications	one addition, one Conv. multiplications	four addition, 16 Conv. multiplications
Decryption	eight addition, 96 Conv. multiplications	one addition, two Conv. multiplications	four addition, 32 Conv. multiplications

Table 2: Message and key security spaces for AH_{QTR}, NTRU, and QTRU

	Message Security Space	**Key Security Space**
AH_{QTR}	$(\frac{N!}{(d_r!)^2(N-2d_r)!})^4 (\frac{N!}{(d_\delta!)^2(N-2d_\delta)!})^4$	$(\frac{N!}{(d_g!)^2(N-2d_g)!})^4 (\frac{N!}{(d_s!)^2(N-2d_s)!})^4$
$QTRU$	$(\frac{N!}{(d_r!)^2(N-2d_r)!})^4$	$(\frac{N!}{(d_g!)^2(N-2d_g)!})^4$
$NTRU$	$(\frac{N!}{(d_r!)^2(N-2d_r)!})$	$(\frac{N!}{(d_g!)^2(N-2d_g)!})$

The comparison of the message security space and the key security space for AH_{QTR}, NTRU, and QTRU according to the generic parameters is shown in Table 2.

With the same number of coefficients, the security level of the message of AH_{QTR} is double times the security level of QTRU and eight times that of NTRU. The security level of the key of AH_{QTR} is double the security level of QTRU and eight times that of NTRU.

5 CONCLUSION

The QTRU system is an improvement to NTRU, whereas, AH_{QTR} is an improvement to the QTRU. When choosing $S = 1$ and $R = 1$, it converts to QTRU. AH_{QTR} provides a high level of security based on its complex mathematical construction. It increases polynomials for private keys and the message, which gives it an advantage in many applications that require a long time for the attackers to try all possibilities, including bank transfers and electronic exams, and other applications that require a long time to maintain confidentiality of data. The defect of high computation can be overcome by choosing lower values of N.

REFERENCES

Abo-Alsood, H. and Yassein, H. (2021a). QOTRU: A New Design of NTRU Public Key Encryption Via Qu-Octonion Subalgebra. In *Journal of Physics: Conference Series*, pages 1–7. IOP Publishing.

Abo-Alsood, H. and Yassein, H. (2021b). Quitru: design secure variant of ntruencrypt via a new multi-dimensional algebra. presented at *Virtual Symposium on Multidisciplinary Science, Malaysia.*

Abo-Alsood, H. H. and Yassein, H. R. (2021c). Design of an Alternative NTRU Encryption with High Secure and Efficient. *Computer Science*, 16(4):1469–1477.

Abo-Alsood, H. H. and Yassein, H. R. (2022). Analogue to NTRU public key cryptosystem by multi-dimensional algebra with high security. In *AIP Conference Proceedings*, pages 1–6. AIP Publishing LLC.

Al-Saidi, N. M. and Yassein, H. R. (2017). A new alternative to NTRU cryptosystem based on highly dimensional algebra with dense lattice structure. *Malaysian Journal of Mathematical Sciences*, 11:29–43.

Alsaidi, N., Saed, M., Sadiq, A., and Majeed, A. A. (2015). An improved NTRU cryptosystem via commutative quaternions algebra. In *Proceedings*

of the international conference on security and management (SAM), page 198. The Steering Committee of The World Congress in Computer Science, Computer ....

Alsaidi, N. M. and Yassein, H. R. (2016). BITRU: binary version of the NTRU public key cryptosystem via binary algebra. *International Journal of Advanced Computer Science and Applications*, 7(11).

Coglianese, M. and Goi, B.-M. (2005). MaTRU: A new NTRU-based cryptosystem. In *International conference on cryptology in India*, pages 232–243. Springer Verlag Berlin Heidelberg.

Gaborit, P., Ohler, J., and Solé, P. (2002). *CTRU, a polynomial analogue of NTRU*. PhD thesis, INRIA Rapport de recherché N.

Hoffstein, J., Pipher, J., and Silverman, J. H. (1998). Ntru: A ring-based public key cryptosystem. In *International algorithmic number theory symposium*, pages 267–288. Springer.

Jarvis, K. and Nevins, M. (2015). ETRU: NTRU over the Eisenstein integers. *Designs, Codes and Cryptography*, 74(1):219–242.

Malekian, E. and Zakerolhosseini, A. (2010). OTRU: A non-associative and high speed public key cryptosystem. In *2010 15th CSI international symposium on computer architecture and digital systems*, pages 83–90. IEEE.

Malekian, E., Zakerolhosseini, A., and Mashatan, A. (2009). QTRU: a lattice attack resistant version of NTRU PKCS based on quaternion algebra. *The ISC Int'l Journal of Information Security*, 3(1):29–42.

Shahhadi, S. H. and Yassein, H. R. (2021a). A New Design of NTRUEncrypt-analog Cryptosystem with High Security and Performance Level via Tripternion Algebra. *Computer Science*, 16(4):1515–1522.

Shahhadi, S. H. and Yassein, H. R. (2021b). A newly algebra for designing secure public-key like ntruencrypt. presented at *Virtual Symposium on Multidisciplinary Science, Malaysia*.

Shahhadi, S. H. and Yassein, H. R. (2021c). NTR_{SH}: A New Secure Variant of NTRUEncrypt Based on Tripternion Algebra. In *Journal of Physics: Conference Series*, pages 1–6. IOP Publishing.

Shahhadi, S. H. and Yassein, H. R. (2022). An innovative tripternion algebra for designing NTRU-like cryptosystem with high security. In *AIP Conference Proceedings*, pages 1–6. AIP Publishing LLC.

Yassein, H. and Al-Saidi, N. (2019). An innovative bi-cartesian algebra for designing of highly performed NTRU like cryptosystem. *Malaysian Journal of Mathematical Sciences*, 13(S):77–91.

Yassein, H., Al-Saidi, N., and Almosawi, A. (2020). A multi-dimensional algebra for designing an improved NTRU cryptosystem. *Eurasian journal of mathematical and computer applications*, 8(4):97–107.

Yassein, H. R., Abidalzahra, A. A., and Al-Saidi, N. M. (2021). A new design of NTRU encryption with high security and performance level. In *AIP Conference Proceedings*, page 080005. AIP Publishing LLC.

Yassein, H. R. and Al-Saidi, N. M. (2016). HXDTRU Cryptosystem Based on Hexadecnion Algebra. In *Proceeding of the 5th International Cryptology and Information Security Conference, Kota Kinabalu, Malaysia.*

Yassein, H. R. and Al-Saidi, N. M. (2017). A comparative performance analysis of NTRU and its variant cryptosystems. In *2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT)*, pages 115–120. IEEE.

Yassein, H. R. and Al-Saidi, N. M. (2018). BCTRU: A New Secure NTRU Crypt Public Key System Based on a Newly Multidimensional Algebra. In *Proceeding of the 6th International Cryptology and Information Security Conference*, pages 1–11.

Yassein, H. R., Al-Saidi, N. M., and Farhan, A. K. (2022). A new NTRU cryptosystem outperforms three highly secured NTRU-analog systems through an innovational algebraic structure. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(2):523–542.

Proceedings of the 8th International Cryptology and Information Security Conference 2022 (CRYPTOLOGY2022)

The Cubic Pell Digital Algorithm CP256-1299

Nur Azman Abu[*1] and **Abderrahmane Nitaj**[2]

[1]*Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia*
[2]*Normandie Univ, UNICAEN, CNRS, LMNO, 14000 Caen, France*

E-mail: nura@utem.edu.my
[*]*Corresponding author*

ABSTRACT

Elliptic curve cryptography (ECC) is increasingly used in various industrial applications for providing privacy and authenticity. ECC is based on the arithmetic of elliptic curves over finite fields and provides cryptosystems with increased security and smaller key sizes. An elliptic curve is not an ellipse. Geometrically, an elliptic curve is more hyperbolic in nature which makes it unpredictable. There are encouraging developments of ECC design which aims at having more efficient and secure curves.

In this paper, we propose another 256-bit ECC variant based on a cubic Pell curve, called CP256-1299. The idea of using a cubic Pell curve has been used to design a cryptosystem which is a variant of RSA. The proposed scheme shall be compared to three popular curves Ed25519, Secp256k1 and Secp256r1. This cubic Pell curve cryptosystem has been designed to ride on a larger order at about the same computing requirement among popular elliptic curve cryptosystems. This CP256-1299 is a part of Digital Ringgit proposal to the central bank of Malaysia, Bank Negara Malaysia (BNM).

Keywords: Elliptic curve cryptography; Elliptic curve; Cubic Pell equation; Cryptosystem.

1 INTRODUCTION

In the last three decades, ECC Koblitz (1987), Miller (1986) has been used in a vast variety of applications. The security of ECC is based on the Elliptic Discrete Logarithm Problem (ECDLP) which provides more security than most of the systems based on factorization such RSA. ECC is used for key exchange and digital signatures Corp (1998), and is the main ingredient in the domain name system (DNS), the blockchain Nakamoto and Bitcoin (2008), and various mobile applications (see Hankerson et al. (2006) for more applications). ECC is known to provide equivalent cryptographic security with smaller key sizes. A 256-bit ECC is sufficient to achieve 128-bit security level compared to RSA Rivest et al. (1978) which requires at least 2048-bit modulus. ECC can carry a smaller key size while providing the same level of security given by an RSA cryptosystem. For instance, a 256-bit ECC should provide security equivalence to a 3072-bit RSA. There are several popular elliptic curves. They are known by their nicknames, such as curves Ed25519, Secp256k1 and Secp256r1. A comparison between Koblitz Secp256k1 and Secp256r1 has been done in Houria et al. (2019). Typically, Secp256k1 is a Koblitz curve which is defined in a characteristic 2 finite field, while Secp256r1 is a prime field curve. The "k" in Sepc256k1 stands for Koblitz and the "r" in Sepc256r1 stands for random. Secp256k1 has hardly been used prior to an advent of Bitcoin Nakamoto and Bitcoin (2008). It is gaining popularity due to its many properties. The curve Secp256k1 has been generated by Certicom Corp (1998) while Secp256r1 has been designed by NIST. Even though they are part of standard curves, Certicom is known to have extensive patent on most of the elliptic curve algorithmic properties.

Development on central bank digital currencies (CBDCs) are ongoing across the globe. Without CBDCs, private digital money would become increasingly dominated by non-financial corporations. This CP256-1299 is a part of Digital Ringgit proposal to the central bank of Malaysia, Bank Negara Malaysia (BNM).

In this paper, we propose another 256-bit digital signature algorithm, more or less related to ECC. As an immediate application, our new cryptosystem could be used as a technical support for the Malaysia Central Bank Digital

Currencies (CBDC) of digital ringgit.

While ECC is based on the algebraic properties of groups of points of elliptic curves, our scheme is based on a group of points of a different kind of curves, raised from the solutions of a cubic Pell equation Barbeau (2003). More precisely, let p be a prime number, and $\mathbb{F}_p$ be the finite field with p elements. An element $c \in \mathbb{F}_p$ is a cubic-residue modulo p if the equation $x^3 \equiv c \pmod{p}$ has at least one solution in $\mathbb{F}_p$. If this equation has no solution in $\mathbb{F}_p$, then c is called a cubic non-residue modulo p.

Let c be a cubic non-residue modulo p. We consider the set

$$\mathcal{C} = \left\{(x, y, z) \in \mathbb{F}_p^3 : x^3 + cy^3 + c^2z^3 - 3cxyz \equiv 1 \pmod{p}\right\}$$

Then $\mathcal{C}$ is a finite group with the following law and properties.

- The sum of two solutions (x_1, y_1, z_1) and (x_2, y_2, z_2) of the Pell equation is defined by $(x_1, y_1, z_1) \oplus (x_2, y_2, z_2) = (x_3, y_3, z_3)$ where

$$\begin{aligned} x_3 &= cy_1z_2 + cy_2z_1 + x_1x_2, \\ y_3 &= cz_1z_2 + x_1y_2 + x_2y_1, \\ z_3 &= x_1z_2 + x_2z_1 + y_1y_2. \end{aligned}$$

- The neutral element is $(1, 0, 0)$.
- The inverse of (x, y, z) is $\left(x^2 - cyz, cz^2 - xy, y^2 - xz\right)$.
- The order of $\mathcal{C}$ is $p^2 + p + 1$.

With the former operations, it is possible to define a scalar multiplication of a point $P = (x, y, z)$ by a scalar $n \geq 1$ as follows

$$P_n = (x_n, y_n, z_n) = nP = P \oplus P \oplus \cdots \oplus P \ (n \text{ times}).$$

This scalar multiplication enables one to define the Cubic Pell Discrete Logarithm Problem (CPDLP).

Definition 1.1 (CPDLP). *Given a prime number p, a non-cubic integer c modulo p, a point $P = (x, y, z)$, and a point $P_n = (x_n, y_n, z_n)$ on the curve with the cubic Pell equation $x^3 + cy^3 + c^2z^3 - 3cxyz \equiv 1 \pmod{p}$, find n, if any, such that $P_n = nP$.*

The PCDLP is untractable in the group $\mathcal{C}$, and will serve to guarantee the security of our new digital signature algorithm. Moreover, our scheme will use a hash function such as SHA2 or SHA3 families.

It is possible to use affine coordinates, and $\mathcal{C}$ can be transformed into a related group $\mathcal{B}$ with a slightly different group law by the following rules.

1. If $x \neq 0$, $z = 0$, $y = 0$, then (x, y, z) can be divided by x and transformed to $(1, 0, 0)$.
2. If $z = 0$ and $y \neq 0$, then (x, y, z) can be divide by y and transformed to $(x, 1, 0)$.
3. If $z \neq 0$, then (x, y, z) can be divide by z and transformed to $(x, y, 1)$.

The order of the group $\mathcal{B}$ is $p^2 + p + 1$. We point out that the idea of using the group $\mathcal{B}$ with the induced law was used in Murru and Saettone (2017) to build a variant of the RSA cryptosystem.

The rest of this paper is organized as follows. In Section 2, we present some preliminaries on the the arithmetic of the cubic Pell equation, and the Cubic Pell Discrete Logarithm Problem. In Section 3, we present our new digital signature scheme based on the Cubic Pell equation. In Section 4, we present a practical comparison between our new scheme and existing ones. We conclude the paper in Section 5.

2 PRELIMINARIES

In 2008, Murru and Saettone Murru and Saettone (2017) proposed a cubic Pell RSA variant. It is intended to be more secure than RSA in broadcast

applications. Working on a cubic field related to a cubic Pell equation, a group field can be constructed to ride on a much larger periodic cycle.

2.1 Background overview on cubic Pell equations

In 1659, John Pell and Johann Rahn have written an algebra text on finding infinitely many positive integer solutions to the quadratic equation $u^2 - dv^2 = 1$. In 1909, Axel Thue has shown that a cubic equation $u^3 - dv^3 = 1$ has finitely many integer solutions. A sequence of solution points (u_n, v_n) modulo a prime number can be generated without bound as n increases without bound. This basic cubic equation is birationally equivalent to an elliptic curves of the form $y^2 = x^3 - D$ (see Cunningham et al. (2006)).

Let p be a prime number, and c a cubic non-residue in $\mathbb{F}_p^*$. We then have

$$\mathbb{F}_p^* = \left\{g, g^2, \ldots, g^{p-1} \equiv 1\right\} \pmod{p},$$

where g is a primitive root of $\mathbb{F}_p^*$. Suppose that $p \equiv 2 \pmod 3$. Then $\gcd(p-1, 3) = 1$, and, by Bezout identity, $(p-1)u + 3v = 1$ for some integers u and v. Then, since $g^{p-1} \equiv 1 \pmod p$, we get

$$g = g^{(p-1)u+3v} = g^{(p-1)u} g^{3v} = (g^v)^3 .$$

Hence g is a cubic residue, and so is every element of $\mathbb{F}_p^*$. Next, suppose that $p \equiv 1 \pmod 3$. For such moduli, the set of cubes is in the form

$$E_3 = \left\{g^3, g^6, \cdots, \left(g^3\right)^{\frac{p-1}{3}} \equiv 1\right\} \pmod{p},$$

where g is a primitive root of $\mathbb{F}_p^*$. For each $a \in E_3$, the equation $t^3 \equiv a \pmod p$ has three solutions, and no solution if $a \notin E_3$. As a consequence, an integer $c \in \mathbb{F}_p^*$ is a cubic residue if and only if $c^{\frac{p-1}{3}} \equiv 1 \pmod p$.

A cubic Pell equation in the finite field $\mathbb{F}_p$ is given by the equation

$$x^3 + cy^3 + c^2 z^3 - 3cxyz \equiv 1 \pmod{p}.$$

A group of points (x, y, z) can be formed by points which satisfy the cubic Pell equation modulo p. The set C of all solutions of the cubic Pell equation form a finite group with a law $\oplus$ and the following properties

- The sum of two solutions (x_1, y_1, z_1) and (x_2, y_2, z_2) of the Pell equation is defined by $(x_1, y_1, z_1) \oplus (x_2, y_2, z_2) = (x_3, y_3, z_3)$ where

$$\begin{aligned} x_3 &= cy_1z_2 + cy_2z_1 + x_1x_2, \\ y_3 &= cz_1z_2 + x_1y_2 + x_2y_1, \\ z_3 &= x_1z_2 + x_2z_1 + y_1y_2. \end{aligned}$$

- The neutral element is $(1, 0, 0)$.
- The inverse of (x, y, z) is $\left(x^2 - cyz, cz^2 - xy, y^2 - xz\right)$.

The following result gives an explicit formula for the order of the group $\mathcal{C}$ Dutto and Murru (2022).

Lemma 2.1. *Let p be a prime number such that $p \equiv 1 \pmod 3$. Let c be a cubic non-residue in $\mathbb{F}_p^*$. Then*

$$\begin{aligned} &\#\left\{(x, y, z) \in \mathbb{Z}/p\mathbb{Z}^3 \ : \ x^3 + cy^3 + c^2z^3 - 3cxyz \equiv 1 \pmod p\right\} \\ &= p^2 + p + 1. \end{aligned}$$

2.2 The Cubic Pell Discrete Logarithm Problem (CPDLP)

The law $\oplus$ in $\mathcal{C}$ can be used to define a scalar multiplication of a point $P \in \mathcal{C}$ by an integer n. This is processed by adding P to itself n times

$$nP = P \oplus P \oplus \ldots \oplus P.$$

The set of multiples of P is denoted $\langle P \rangle$. To compute nP, various algorithms can be used such as Square-and-Multiply Exponentiation Algorithm Hankerson et al. (2006). The scalar multiplication gives rise to the Cubic Pell Discrete Logarithm Problem (CPDLP) as follows.

Definition 2.1 (CPDLP). *Let P and Q be two points of $\mathcal{C}$. The Cubic Pell Discrete Logarithm Problem (CPDLP) is to find an integer n such that $Q = nP$ if any.*

In general, the elliptic curve discrete logarithm problem (ECDLP) is an intractable problem and is a fundamental building block for elliptic curve cryptography. There are many known generic algorithms devoted to solve the DLP An elliptic curve E over a finite field $\mathbb{F}_p$ has an order of the form $\#E = p + 1 - t_p$, where, according to Hasse's Theorem, $0 \leq |t_p| \leq 2\sqrt{p}$. As a consequence, the generic algorithms such as Baby-Step Giant-Step Shanks (1973), Pollard rho Pollard (1974, 1978), and Pohling-Hellman Pohlig and Hellman (1978) have running time $\mathcal{O}\left(\sqrt{p}\right)$.

The same generic algorithms can obviously be applied to solve the CPDLP in the group $\mathcal{C}$. Since the order of $\mathcal{C}$ is $N = p^2 + p + 1$ for $p \equiv 1 \pmod 3$, then the running time is $\mathcal{O}(p)$, which is much larger than the running times for elliptic curves.

The security of our new system is based on the difficulty of solving the CPDLP. To ovoid attacks, it is necessary that the order $p^2 + p + 1$ is sufficiently large, and is divisible by a sufficiently large prime factor. Note that some elliptic curves are weak since the ECDLP can be easily solved. A typical example is the family of anomalous elliptic curves satisfying $\#E = p$. For anomalous elliptic curves, the ECDLP can be solved in time $\mathcal{O}\left(\log(p)\right)$ (see Satoh et al. (1998), Smart (1999)). Another known example is the family of elliptic curves that are weak for MOV attack Menezes et al. (1993).

2.3 A numerical example for the cubic Pell equation

There are 3 cases in periodic cycles which satisfy a cubic Pell equation

$$E_p : x^3 + cy^3 + c^2z^3 - 3cxyz = 1 \pmod p.$$

Let p be a prime, a periodic cycle is mostly determined by a prime modulo p. There are three cases, namely $\#E_p \in \left\{(p-1)^2, p^2 - 1, p^2 + p + 1\right\}$. If c is a cubic non-residue of $\mathbb{F}_p^*$, then, by Theorem 2.1 $\#E_p = p^2 + p + 1$.

In order to gain a similar gain in general ECC, the third case will be chosen. We will take a small instance on each case on parameter $c = 7$. From an identity $e = P_0(x_0, y_0, z_0) = (1, 0, 0)$ and a base point $P_1(x_1, y_1, z_1) = (4, 2, 1)$,

we can compute $P_n(x_n, y_n, z_n)$ via a balanced point projection algorithm Abu and Abd Ghafar (2015). An instance is given in Table 1.

i	b_i	P_i	x_i	y_i	z_i
24	1	1	4	2	1
23	0	2	44	23	12
22	0	4	1689	3032	1585
21	0	8	3252	312	2436
20	0	16	3192	3653	2741
19	0	32	1039	2704	2231
18	0	64	2791	4088	1068
17	0	128	764	3972	1155
16	1	257	3554	3514	3806
15	1	515	1159	936	1178
14	1	1031	3451	993	1173
13	1	2063	3158	3674	2285
12	1	4127	2060	1117	2715
11	0	8254	3921	3245	1825
10	0	16508	2314	1094	3013
9	0	33016	3207	1736	187
8	0	66032	1320	239	3450
7	1	132065	1212	1526	1614
6	1	264131	2409	2110	1312
5	1	528263	1392	468	1482
4	1	1056527	3554	3259	1422
3	0	2113054	2294	4013	995
2	0	4226108	68	1615	3232
1	0	8452216	3008	168	1526
0	1	16904433	1	0	0

Table 1: Take $p = 4111$ and $n = p^2 + p + 1 = 16904433 = 1000000011111000011110001$, then a projection point is on the left $n \otimes (4, 2, 1) = (x_n, y_n, z_n)$ goes back to an identity $(1, 0, 0)$.

3 A NEW CRYPTOSYSTEM DESIGN BASED ON THE CUBIC PELL EQUATION

In this section, we present our new system for digital signatures based on the cubic Pell equation.

3.1 Generating an efficient strong 256-bit prime

A choice of prime p will determine ECC algebraic efficiency in computing modulo p due to its friendly form to CPU word processing. Since the target here is 256-bit ECC, a prime modulus is preferably chosen as to a power of 256 for efficient modular reduction. The best candidate is in a tight form $2^{256} - k$ for some small integer k. An integer $k = 1299$ is chosen as the smallest positive integer for which $2^{256} - k$ satisfies strong prime criteria. It should be noted that $k = 1299 = 2^{10} + 2^8 + 2^4 + 2^1 + 1 < 216$. This prime is also well protected on the left and the right with

$$\begin{aligned} p &= 2^{256} - 1299 \\ &= 115792089237316195423570985008687907853269984665640564039457584007913129638637, \\ p - 1 &= 2^2 \cdot 3 \cdot 79 \cdot 116121953 \cdot 4944856253 \cdot 21271717174740349246610256385913263328140464110678736952 3, \\ p + 1 &= 2 \cdot 107 \cdot 3113857811 \cdot 173766614273427837579699871663288827015055936164654544261019401447. \end{aligned}$$

On the left, its largest prime factor on $p - 1$ is a 188-bit prime while on the right its largest prime factor on $p + 1$ is a 217-bit prime.

3.2 Selection of a system parameter and a base point

Take a sequence starting from an identity point $P_0(x_0, y_0, z_0) = (1, 0, 0)$ and an initial base point $P_1(x_1, y_1, z_1)$ which satisfies a cubic Pell equation. The smallest sample point $(x, y, z) \in \mathbb{F}_p^3$ on small parameters $c = 2, \ldots, 7$ with $p = 2^{256} - 1299$ are listed in Table 2.

c	x	y	z
2	1	1	1
3	4	3	2
5	41	24	14
6	109	60	33
7	4	2	1

Table 2: The smallest point is an ideal base point on a cubic Pell equation.

Since there is an issue on a random base point of Secp256r1, a basepoint shall be prescribed from a fundamental solution of a cubic Pell equation and project it to a $(p+1)$ point. For instance, let $c = 7$ then a fundamental solution is $(4, 2, 1)$. Take a base point as

$$(p+1) \otimes (4, 2, 1) = (2, 101572757264380952535097412522523772699491042803557044710886595566980934141402, 0).$$

In this ECC, such a base point can be generated since a cubic Pell equation gives a luxury of riding on a larger periodic order of $p^2 + p + 1$. A random secret key in this cryptosystem is within $(2, p - 2)$. In a digital signature scheme, a projection is done twice. First, it is done by a private key. Second, it is done by a secret session key. A double projection in this scheme will not go over bound of the periodic order $p^2 + p + 1$.

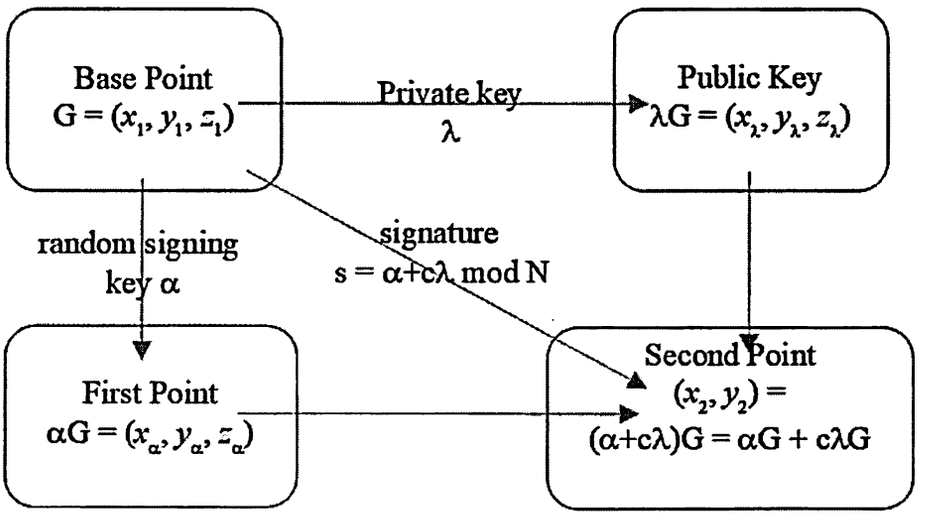


Figure 1: Point projection in a basic digital signing and verification.

3.3 Selection on digital signature and verification scheme

An Elliptic Curve Digital Signature Algorithm (ECDSA) is based on the Digital Signature Algorithm (DSA). Secp256r1 curve is the most popular elliptic curve as part of NIST standards (FIPS 186-4). In an open public ledger, for example, Bitcoin can process 7,000 transactions per second. A modern payment needs to process about 100,000 transactions per second using Secp256r1. However, digital signing and verification using EdDSA on Ed25519 is faster and more secure than ECDSA on Secp256r1.

A digital signature scheme with anonymity and spontaneity are typically referred to as a ring of signatures. In the context of digital ringgit, they will ultimately allow for unforgeable, signer-ambiguous transactions that leave currency flows largely untraceable.

System parameters in this cryptosystem initial proposal are as follows:

i. A prime modulus $p = 2^{256} - 1299$,

ii. An identity point $P_0(x_0, y_0, z_0) = (1, 0, 0)$,

iii. A parameter $c = 7$,

iv. A periodic order $\#E(\mathbb{F}_p) = p^2 + p + 1$,

v. A base point $G = P_1(x_1, y_1, z_1) = (p+1) \otimes (4, 2, 1)$,

vi. A public key $\lambda G = P_\lambda(x_\lambda, y_\lambda, z_\lambda) = \lambda \otimes (x_1, y_1, z_1)$.

An output pair $(\alpha G, s)$ is expected to be a digital signature on a message m from an owner of public key λG. It should be noted that α is a random 512-bit session number in a traditional digital signature algorithm. A digital signature here consists of an EC point αG, a signature scalar s and a public key λG. They are compactly represented in 32+32+32 bytes. They will be visualized as three emblems in this project.

Digital Signature

Let $G = P_1(x_1, y_1, z_1)$ be a base point generator and λ be a private key. Then take precomputed $\lambda G = P_\lambda(x_\lambda, y_\lambda, z_\lambda)$ as a public key. Computing a multiple λG of point G is considered as a one-way function. Given both base points G and λG, it is intractable to extract λ from them.

i. Generate random scalar 512-bit α and compute $\alpha G = P_\alpha(x_\alpha, y_\alpha, z_\alpha)$.

ii. Compute $\sigma = \text{SHA2}(m)$.

iii. Calculate a signature scalar $s \equiv \alpha + \sigma\gamma \pmod{p^2 + p + 1}$.

iv. Output a signature pair $(\alpha G, s)$ of message m.

An output pair $(\alpha G, s)$ is expected to be a digital signature on a message m from an owner of public key λG. It should be noted that α is a random session number in a traditional digital signature algorithm. A digital signature here consists of an EC point αG, a signature scalar s and a public key K. They are represented in six 32-bytes. They will be visualized as six emblems in future proposal.

Signature Verification

From a signature pair $(\alpha G, s)$, public key λG and a message m,

i. Compute $\sigma' = \text{SHA2}(m)$.

ii. Compute $Q = sG = s \otimes (x_1, y_1, z_1)$.

iii. Compute $\sigma'\lambda G = \sigma' \otimes (x_\lambda, y_\lambda, z_\lambda) = (x_{\lambda\sigma'}, y_{\lambda\sigma'}, z_{\lambda\sigma'})$.

iv. Compute $Q' = \alpha G + \sigma'\lambda G = (x_\alpha, y_\alpha, z_\alpha) + (x_{\lambda\sigma'}, y_{\lambda\sigma'}, z_{\lambda\sigma'})$.

v. Check on validation whether $Q = Q'$.

Referring to Figure 1, there are two paths to compute and project from a base point G to a second point $(\alpha+\sigma\lambda)G$. First, given a signature scalar $s = \alpha+\sigma\lambda$ and system parameter base point G, the second point can be computed directly via a point multiplication sG.

Second, given a first point αG as part of a signature, take a public key γG and message m, then a scalar c can be independently computed as $c' = \text{SHA2}(m)$. Next, $c'\lambda G$ will be projected from a public key λG via a point multiplication. Thus, αG and $c'\lambda G$ will be added together to form $\alpha G + \sigma'\lambda G = (\alpha + \sigma'\lambda)G$.

In a case of both first and second paths will give the same answer, then the pair $(\alpha G, s)$ is considered a valid signature on a message m from an owner of public key λG who must have used a private key λ in computing $s = \alpha + \sigma\lambda$ to digitally sign it.

An Example on Digital Signature

Let us take a sample 256-bit private key from the next prime of a 256-bit fraction of an exponential number e,

λ =831713535784724095196510241312745119742990801481105920105552158153065082921 89.

Then take precomputed public key,

$$\begin{aligned}\lambda G =& P_\lambda(x_\lambda, y_\lambda, z_\lambda)\\ =&(80652997912631420190387978371245749184911663089942002269536582833308066000638,\\ &14781155146764325842105941936507012120419574796497643063788349941245912181899 9,\\ &10331671645360432302242651223816484639870195054949637384751936796284443968853 1).\end{aligned}$$

Take a 512-bit random session from a 512-bit fraction of a popular number π,

$$\begin{aligned}\alpha =&189844710362284492072474648994184972281789985171207447242400075693851369205545798938826247774701606337367572235687533276603126818975945170305282718558031 1.\end{aligned}$$

Compute the first projection point

$$\begin{aligned}\alpha G =& P_\alpha(x_\alpha, y_\alpha, z_\alpha)\\ =&(78193603571961798132195496358874452069777436011327297247055985179143770400755,\\ &58933705053663476286578595285430900140613957738341869185239379173447439819297,\\ &10423227864793148517672585908193216725183045172061930955599009086644824319290 2).\end{aligned}$$

Take a simple message $m =$ "abc". Compute

$$\begin{aligned}\sigma =&\text{SHA2}(m)\\ =&84342368487090800366523834928142263660104883695016514377462985829716817089965.\end{aligned}$$

Calculate a signature scalar

$$\begin{aligned}s =&\alpha + \sigma\gamma\\ =&89133160547084829443831285993992937680553851107239543247139107928154977476339926326145468844234627773692251985971712078754002664864475062712658881053636 96.\end{aligned}$$

An Example on Signature Verification

From a signature pair $(\alpha G, s)$, public key λG and a message m. Compute

$$\begin{aligned}\sigma' =&\text{SHA2}(m)\\ =&84342368487090800366523834928142263660104883695016514377462985829716817089965.\end{aligned}$$

Compute

$$\begin{aligned}Q = sG =&s \otimes (x_1, y_1, z_1)\\ =&(67000248439631917674742898485920369160223444648448\\ &26868288579478329162858796 2,\\ &84027019278154678998474883614044676081906680375763\\ &48717824558406315685936921 5,\\ &42636317058827573429703393632571482481359474039619\\ &54794084418389595220314442 7).\end{aligned}$$

Compute

$$\begin{aligned}\sigma'\lambda G =&\sigma' \otimes (x_\lambda, y_\lambda, z_\lambda)\\ =&(x_{\lambda\sigma'}, y_{\lambda\sigma'}, z_{\lambda\sigma'})\\ =&(10718871988627107781032047484950894973099234000916\\ &13111815137240261919501177 52,\\ &78189457200157595846947313250786321533443769992089\\ &12517714139612602468831101 8,\\ &72444716359751134594684675486446796463466878218442\\ &77356126872019093300970587 9).\end{aligned}$$

Compute an addition point

$$\begin{aligned}Q' =&\alpha G + \sigma'\lambda G\\ =&(x_\alpha, y_\alpha, z_\alpha) + (x_{\lambda\sigma'}, y_{\lambda\sigma'}, z_{\lambda\sigma'})\\ =&(67000248439631917674742898485920369160223444648448268682885794783291628587962,\\ &840270192781546789984748836140446760819066803757634871782455840631568593692 15,\\ &426363170588275734297033936325714824813594740396195479408441838959522031444 27).\end{aligned}$$

Both point Q' and point Q are indeed equal. They are moving towards the same second point. Thus, this signature has been verified.

4 COMPARISON

Table 3 presents a usage comparison of elliptic curves, prime modulus and adoption system among popular ECC.

ECC	Curve	p	Adoption
Ed25519	$-x^2+y^2 = 1-\frac{121665}{121666}x^2y^2$	$2^{255}-2^4-2^1-1$	Monero
Secp256k1	$y^2 = x^3+7$	$2^{256}-2^{32}-2^9-2^8-2^7-2^6-2^4-1$	Bitcoin Ethereum
Secp256r1	$y^2 = x^3-3x+b$	$2^{256}-2^{224}+2^{192}+2^{96}-1$	Hyperledger Fabric
CP256-1299	$x^3+cy^3+c^2z^3-3cxyz = 1$	$2^{256}-2^{10}-2^8-2^4-2^1-1$	Digital Ringgit

Table 3: Comparison of ECC usage of elliptic curves

The security of the systems Ed25519, Secp256k1, and Secp256r1 is based on the ECDLP, while the security of CP256-1299 is based on the CPDLP. With the generic algorithms, the run time to solve ECDLP is $O\left(\sqrt{p}\right)$ while it is $O\left(p\right)$ to solve CPDLP. As a consequence, CP256-1299 is more secure than Ed25519, Secp256k1, and Secp256r1.

5 CONCLUSION

An elliptic curve cryptosystem has been recognised to provide a compact support with smaller key sizes among popular public-key cryptosystems. In this paper, an elliptic curve over a 256-bit prime field has been proposed. This new ECC has incorporated recent development on an ECC design which aims at having an efficient secure curve. This paper proposes another 256-bit elliptic curve cryptosystem (ECC) variant based on a cubic Pell curve. This cubic Pell cryptosystem has been designed to ride on a larger order at about the same computing requirement in an elliptic curve cryptosystem. This new elliptic curve cryptosystem called CP256-1299 has been compared to 3 popular curves Ed25519, Secp256k1 and Secp256r1. We aimed new cryptosystem to be used as a technical support for the Malaysia Central Bank Digital Currencies (CBDC) of digital ringgit.

REFERENCES

Abu, N. A. and Abd Ghafar, A. H. (2015). A Secure Cryptographic Algorithm against Side Channel Attacks. 5(2):45–55.

Barbeau, E. (2003). *The cubic analogue of Pell's equation. In: Pell's Equation. Problem Books in Mathematics.* Springer.

Corp, C. (1998). The elliptic curve crypto system for smart cards,: Certicom white paper. Technical Report `https://www.certicom.com/content/certicom/en/ecc.html`, Certicom Corp.

Cunningham, J. A., Ho, N., Lostritto, K., Middleton, J. A., and Thomas, N. T.

(2006). On Large Rational Solutions of Cubic Thue Equations: What Thue Did to Pell. *Rose-Hulman Undergraduate Mathematics Journal*, 7(2):6.

Dutto, S. and Murru, N. (2022). On the cubic Pell equation over finite fields. *arXiv preprint arXiv:2203.05290* `https://arxiv.org/abs/2203.05290`.

Hankerson, D., Menezes, A. J., and Vanstone, S. (2006). *Guide to elliptic curve cryptography*. Springer Science & Business Media.

Houria, A., Abdelkader, B. M., and Abderezzak, G. (2019). A comparison between the secp256r1 and the koblitz secp256k1 bitcoin curves. *Indonesian Journal of Electrical Engineering and Computer Science*, 13(3):910–918.

Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209.

Menezes, A. J., Okamoto, T., and Vanstone, S. A. (1993). Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Transactions on information Theory*, 39(5):1639–1646.

Miller, V. S. (1986). Advances in cryptology—crypto'85 proceedings. *Use of elliptic curves in cryptography*, pages 417–426.

Murru, N. and Saettone, F. M. (2017). A novel rsa-like cryptosystem based on a generalization of the rédei rational functions. In *International Conference on Number-Theoretic Methods in Cryptology*, pages 91–103. Springer.

Nakamoto, S. and Bitcoin, A. (2008). A peer-to-peer electronic cash system. *Bitcoin.–URL:* `https://nakamotoinstitute.org/static/docs/bitcoin.pdf`, 4:2.

Pohlig, S. and Hellman, M. (1978). An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance (corresp.). *IEEE Transactions on information Theory*, 24(1):106–110.

Pollard, J. M. (1974). Theorems on factorization and primality testing. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 76, pages 521–528. Cambridge University Press.

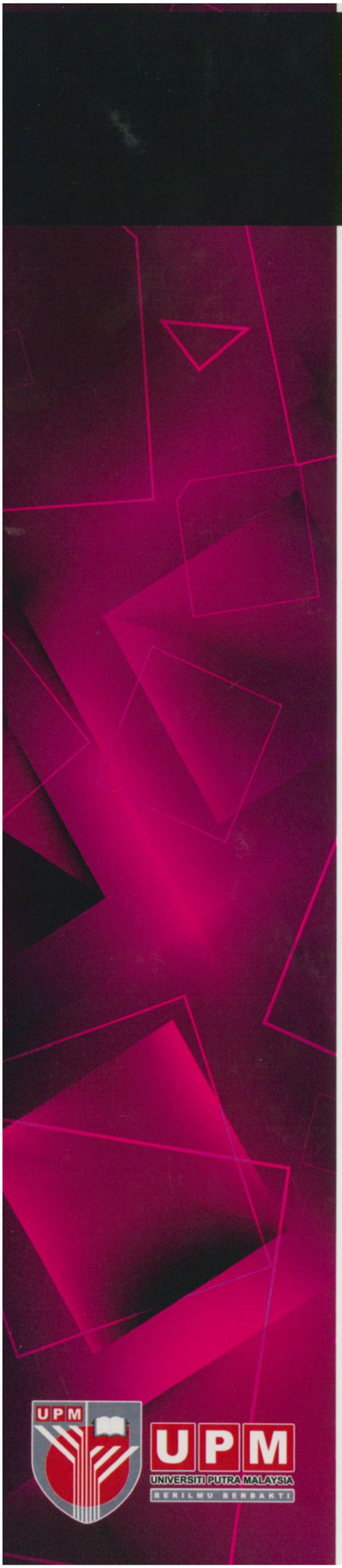
Pollard, J. M. (1978). Monte Carlo methods for index computation $(\mathrm{mod}\ p)$. *Mathematics of computation*, 32(143):918–924.

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.

Satoh, T., Araki, K., et al. (1998). Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. *Rikkyo Daigaku sugaku zasshi*, 47(1):81–92.

Shanks, D. (1973). Five number-theoretic algorithms. In *Proceedings of the Second Manitoba Conference on Numerical Mathematics (Winnipeg), 1973*, pages 51–70.

Smart, N. P. (1999). The discrete logarithm problem on elliptic curves of trace one. *Journal of cryptology*, 12(3):193–196.



Institute for Mathematical Research
Universiti Putra Malaysia,
43400 UPM, Serdang, Selangor,
MALAYSIA.
Website: http://www.inspem.upm.edu.my

ISSN 2716-6783