International Journal of Cryptology Research 9(1): 19 - 51 (2019)

# Design and Analysis of Rabin-$p$ Key Encapsulation Mechanism for CyberSecurity Malaysia MySEAL Initiative

**Muhammad Asyraf Asbullah**[*1,3], **Muhammad Rezal Kamel Ariffin**[1,2], and **Zahari Mahad**[1]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia*
[2]*Department of Mathematics, Faculty of Science, Universiti Putra Malaysia*
[3]*Centre of Foundation Studies for Agricultural Science, Universiti Putra Malaysia*

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

## ABSTRACT

The modular square root problem has a special property of the having computational equivalent to a well-known hard mathematical problem namely integer factorization problem. The proposed Rabin-$p$ Key Encapsulation Mechanism is built upon the said problem as its source of security, aiming for efficient and practical modular square root-based cryptosystem of which accompanied with the following properties; 1) improves the performance without plaintext padding mechanisms or sending extra bits during encryption and decryption processes, 2) the plaintext is uniquely decrypted without decryption failure, 3) improve decryption efficiency by using only one modular exponentiation, 4) a decryption key using only a single prime number, 5) sufficiently large plaintext space, 6) appropriate plaintext-ciphertext expansion ratio, 7) implementable on software and hardware with ease, and 8) achieves IND-CPA security.

**Keywords:** Rabin-$p$ cryptosystem, key encapsulation, AKBA MySEAL, CyberSecurity Malaysia

# 1 INTRODUCTION

## 1.1 Background

The Rabin encryption scheme Rabin (1979) is one of an existing workable asymmetric cryptosystem that comes with nice cryptographic properties. For instance, it has low-cost encryption of which the Rabin encryption is relatively fast to encrypt compared to the widely commercialized RSA cryptosystem Rivest et al. (1978), and it has been proven to be as difficult as the integer factorization problem. On the other hand, the decryption of Rabin's scheme produces four possible answers, which only one is correct. This four-to-one decryption setting of the Rabin decryption could lead to a decryption failure scenario since no indicator for selecting the correct plaintext is given.

Theoretically speaking, it is such a waste to abandon a cryptosystem that possesses nice features such as the Rabin cryptosystem. Hence attempts were made by numerous researchers with the objective to turn the Rabin cryptosystem to be as practical and implementable as the RSA cryptosystem. Broadly speaking, all the previous attempts made seem to employ one or more additional features in order to obtain a unique decryption result, but at the same time may have a small probability for decryption failure. One of the ways to accomplish this is through manipulation of some mathematical objects such as the role of the Jacobi symbol or the Dedekind's sums theorem. Also, it can be done by designing an encryption function with a special message structure. Yet, at the same time all the designs lose the computational advantage of the original Rabin's encryption over the RSA cryptosystem.

In order to engage this problem and to overcome all the previous drawbacks of Rabin's original design and its variants, we propose the Rabin-$p$ Key Encapsulation Mechanism, provided with theoretical analysis, perfomance mea-

surement and robust implementation. We revisit the Rabin cryptosystem and then aspire to furnish a new design aiming for efficient, secure and practical Rabin-like cryptosystem. In our design, we use the modulus $N = p^2q$ and we restrict the plaintext to be less than $p^2$. Hence, to decrypt correctly, it suffices to apply an efficient algorithm that solves the square root of quadratic congruence modulo $p$ instead of modulo $N = p^2q$.

## 1.2 Design Rationale

In designing the Rabin-$p$ Key Encapsulation Mechanism, the following are the main criteria that were taken into consideration:

1. improves the performance without plaintext padding mechanisms or sending extra bits during encryption and decryption processes
2. the plaintext is uniquely decrypted without decryption failure
3. improve decryption efficiency by using only one modular exponentiation
4. a decryption key using only a single prime number
5. sufficiently large plaintext space
6. appropriate plaintext-ciphertext expansion ratio
7. implementable on software and hardware with ease
8. achieves IND-CPA security.

## 1.3 Design Principle

The design principle to overcome the drawbacks of the original Rabin cryptosystem and all its variants are outlined as follows. Firstly, we put the condition on the modulus to be used is of the type $N = p^2q$. We note that such modulus $N = p^2q$ is claimed to be no easier than factoring the conventional modulus of $N = pq$ Castagnos et al. (2009). We then impose restriction on

the plaintext $m$ and ciphertext $c$ space as $m \in \mathbb{Z}_{p^2}$ and $c \in \mathbb{Z}_{p^2q}$, respectively. From the plaintext-ciphertext expansion, such restriction leads to a system that is not a length-preserving for the message.

Let $m$ and $c$ be the plaintext and ciphertext and $c(m)$ be the function of $c$ taking $m$ as its input. Say, for instance, the plaintext spaces and the ciphertext spaces in the RSA cryptosystem are the same. Thus we denote the mapping for the RSA cryptosystem as $c(m) : \mathbb{Z}_{pq} \longrightarrow \mathbb{Z}_{pq}$. Note that this situation could be an advantage for the RSA scheme since RSA encryption has no message expansion. This is, however, not true for all cryptosystems.

The size of a message $m$ is determined by the size of its plaintext space. Suppose we put a restriction on the size of such $m$. If the intended plaintext $m$ is merely the secret key needed for the use of a symmetric cryptosystem, then such key is indeed a short message. For example, the plaintext-ciphertext mapping for Okamoto-Uchiyama cryptosystem Okamoto and Uchiyama (1998) is $c(m) : \mathbb{Z}_p \longrightarrow \mathbb{Z}_{p^2}$, Pailier cryptosystem Paillier (1999) and the cryptosystem proposed by Galindo et al. (2002) is $c(m) : \mathbb{Z}_{pq} \longrightarrow \mathbb{Z}_{(pq)^2}$, Rabin-Boneh Boneh (2001) mapping is $c(m) : \mathbb{Z}_{\frac{pq}{2}} \longrightarrow \mathbb{Z}_{pq}$ and the Rabin variant introduced by Schmidt-Samoa (2006) is $c(m) : \mathbb{Z}_{pq} \longrightarrow \mathbb{Z}_{p^2q}$.

Therefore, we note that the issue of losing the ability to encrypt a relatively longer $m$ is insignificant. Hence, we reason that, even imposing restrictions on the plaintext space or to set a prefix message size would not be a hindrance for designing a considerable efficient cryptosystem.

## 2 RABIN-$P$ CRYPTOSYSTEM: THE DESIGN

In this section, we provide the details of the proposed cryptosystem namely Rabin-$p$ Cryptosystem. Rabin-$p$ is named after the Rabin cryptosystem with the additional $p$ symbolizing that the proposed scheme only uses a single prime $p$ as the decryption key. This section is structured as follows. We first describe

the Rabin-$p$ key generation, encryption and decryption procedures. We then provide the explanation of the Rabin-$p$ decryption process.

## 2.1 System Parameters

The key generation algorithm of the Rabin-$p$ cryptosystem (Algorithm 2.1) produces two random and distinct primes $p$ and $q$ of the same length such that $p \equiv 3 \pmod 4$ and $q \equiv 3 \pmod 4$.

## 2.2 Rabin-$p$ Key Generation Algorithm

The key generation algorithm then produces an integer $N$ as a product $N = p^2q$, which is denoted as the public key. The private key is the prime $p$.

**Algorithm 2.1** Rabin-$p$ Key Generation Algorithm

**Input:** The size $k$ of the security parameter
**Output:** The public key $N = p^2q$ and the private key $p$

1: Choose two random and distinct primes $p$ and $q$ such that $2^k < p, q < 2^{k+1}$ satisfy $p, q \equiv 3 \pmod 4$
2: Compute $N = p^2q$
3: Return the public key $N$ and the private key $p$

## 2.3 Rabin-$p$ Encryption Algorithm

To encrypt a plaintext, the Rabin-$p$ encryption algorithm with the public key $N$ does the following.

**Remark 2.1.** *The encryption algorithm (Algorithm 2.2) takes the plaintext $m < 2^{2k-1}$ and compute $c \equiv m^2 \pmod N$. We observe that the plaintext $m$ is restricted to the range of $m < 2^{2k-1} = \frac{2^{2k}}{2} < \frac{p^2}{2} < p^2$. The output is the ciphertext $c$.*

**Algorithm 2.2** Rabin-$p$ Encryption Algorithm

**Input:** The public key $N$
**Output:** A ciphertext $c$
1: Choose plaintext $0 < m < 2^{2k-1}$ such that $\gcd(m, N) = 1$
2: Compute $c \equiv m^2 \pmod{N}$
3: Return the ciphertext $c$

## 2.4 Rabin-$p$ Decryption Algorithm

To decrypt a ciphertext, the Rabin-$p$ decryption algorithm with the private key $p$ does the following.

**Algorithm 2.3** Rabin-$p$ Decryption Algorithm

**Input:** A ciphertext $c$ and the private key $p$
**Output:** The plaintext $m$
1: Compute $w \equiv c \pmod{p}$
2: Compute $m_p \equiv w^{\frac{p+1}{4}} \pmod{p}$
3: Compute $i = \frac{c - {m_p}^2}{p}$
4: Compute $j \equiv \frac{i}{2m_p} \pmod{p}$
5: Compute $m_1 = m_p + jp$
6: If $m_1 < 2^{2k-1}$, then return $m = m_1$. Else, return $m = p^2 - m_1$

**Remark 2.2.** *We observe that the decryption algorithm needs only a single prime number as its key. Hence, only one modular exponentiation is taking place during the decryption process. Such computational advantage would positively affect the overall operations.*

**Remark 2.3.** *We reason that since our proposed scheme does not need to carry out any CRT computation, thus the Novak's attack is not applicable on the Rabin-p cryptosystem (i.e. resilient against Novak's attack).*

## 2.5 Proof of Correctness for Rabin-$p$ Decryption

This section explain why the Rabin-$p$ decryption procedure works.

**Lemma 2.1.** *(Kumanduri and Romero, 1998). Let $p$ be a prime number such that $p \equiv 3 \pmod 4$ and $c$ an integer such that $\gcd(c, p) = 1$. The congruence $c \equiv m^2 \pmod p$ has either no solutions or exactly two solutions. If $m_1$ is a solution, then $-m_1 \pmod p$ is the other solution.*

**Lemma 2.2.** *(Kumanduri and Romero, 1998). Let $p$ be a prime number such that $p \equiv 3 \pmod 4$ and $c$ an integer such that $\gcd(c, p) = 1$. The congruence $c \equiv m^2 \pmod{p^2}$ has exactly two solutions if $c \equiv m^2 \pmod p$ has exactly two solutions.*

**Lemma 2.3.** *(Asbullah and Ariffin, 2016). Consider Lemma 2.2. Let $c \equiv m^2 \pmod{p^2}$. Then $m_1 = m_p + jp$ is a solution to $c \equiv m^2 \pmod{p^2}$ where $m_p \equiv c^{\frac{p+1}{4}} \pmod p$, $j \equiv \frac{i}{2m_p} \pmod p$ such that $i = \frac{c - {m_p}^2}{p}$. Furthermore $m_2 \equiv -m_1 \pmod{p^2}$ is the other solution.*

**Lemma 2.4.** *(Asbullah and Ariffin, 2016) Consider Lemma 2.3. If $m_1$ and $m_2$ are the two distinct integers solution for $c \equiv m^2 \pmod{p^2}$, then $m_1 + m_2 = p^2$.*

**Lemma 2.5.** *(Asbullah and Ariffin, 2016) Let $m_1$ and $m_2$ be integers such that $m_1 + m_2 = p^2$ with $p^2$ is an odd integer. Then either $m_1$ or $m_2$ is less than $\frac{p^2}{2}$.*

**Theorem 2.1.** *Let $c \equiv m^2 \pmod N$ be the Rabin-p ciphertext. Then Algorithm 2.3 is **correct**.*

**Proof.** Suppose $c \equiv m^2 \pmod N$ be the Rabin-$p$ ciphertext where $N = p^2q$, thus we have $c - m^2 \equiv 0 \pmod N$. Since $p^2 \mid N$, then $p^2 \mid c - m^2$. Algorithm 2.2 show that $m < p^2$, therefore it is sufficient just solving for $c \equiv m^2 \pmod{p^2}$ which is efficiently solved using Lemma 2.3. In addition, according to Lemma 2.2, there are exactly two distinct solution $m_1$ and $m_2$ satisfies $c \equiv m^2 \pmod{p^2}$. From Lemma 2.4 we have $m_1 + m_2 = p^2$. We now show that the Algorithm 2.3 only produce a unique solution for $m < 2^{2k-1}$. Observe that the upper bound for $m < \frac{p^2}{2}$. Consider Lemma 2.5, then we have either $m_1$ or $m_2$ is less than $\frac{p^2}{2}$ such that $m_1 + m_2 = p^2$ satisfy $m < 2^{2k-1}$. Finally, we conclude that only one of $m_1$ or $m_2$ are less than $\frac{p^2}{2}$ and will be outputted by Algorithm 2.3 as the unique $m < 2^{2k-1}$. □

# 3 RABIN-$P$ CRYPTOSYSTEM: THE ANALYSIS

This chapter discusses the hard problem that becomes the source of security for the Rabin-$p$ cryptosystem. In the following sections, we show that the problem of solving the Rabin-$p$ ciphertext is reduced to factoring $N = p^2q$. Hence, in conclusion, it proves that breaking the Rabin-$p$ cryptosystem is indeed equivalent to factoring $N = p^2q$. We then extend our security analysis by discussing some possible cryptanalysis, for instance; the continued fraction's attack, the Coppersmith's theorems and the Novak's attack.

## 3.1 Reduction to Factoring $N = p^2q$

In this section, we show that if there exists an algorithm that can decrypt message $m$ from any random Rabin-$p$ ciphertext, then such algorithm also be able to factor $N = p^2q$. We observe the following.

**Theorem 3.1.** *Let $N = p^2q$, $m < 2^{2k-1}$ and $2^{2k-1} < \hat{m} < p^2$ such that $m + \hat{m} = p^2$. Then $\gcd(m + \hat{m}, N) = p^2$.*

**Proof.** Suppose $2^k < p < 2^{k+1}$, then $2^{2k} < p^2 < 2^{2k+2}$, and $2^{2k-1} < \frac{p^2}{2} < 2^{2k+1}$. Suppose $m < 2^{2k-1}$, then from Lemma 2.5 there exists another integer $\hat{m} > 2^{2k-1}$ such that $m + \hat{m} = p^2$. Thus this implies $p^2 - \hat{m} = m < 2^{2k-1}$. Now, we determine the range of the $\hat{m}$ such that $p^2 - \hat{m} < 2^{2k-1}$. Then we obtain the lower bound for $\hat{m}$, of which

$$\begin{aligned} \hat{m} &> p^2 - 2^{2k-1} \\ &> 2^{2k} - 2^{2k-1} \\ &> 2^{2k-1} \end{aligned}$$

and upper bounded by $\hat{m} < p^2$. Take the $\gcd(m + \hat{m}, N)$, then we obtain $p^2$. Hence $q = \frac{N}{p^2}$. □

**Remark 3.1.** *Theorem 3.1 implies that if there exists someone or an algorithm that can decrypt the message $m$ from the Rabin-$p$'s ciphertext, then that someone must also be able to factor $N = p^2q$.*

### 3.1.1 Algorithm for Factoring $N = p^2q$

Note that the Algorithm 2.3 will output only the integer $m < 2^{2k-1}$. Hence, if we generate an integer $\hat{m}$ such that $2^{2k-1} < \hat{m} < 2^{2k}$, then we can build a factoring algorithm for $N$, according to Theorem 3.1 and the Algorithm 2.3. The factoring algorithm is defined as follows.

**Algorithm 3.1** Algorithm for Factoring $N = p^2q$

**Input:** A ciphertext $c$ and the modulus $N$
**Output:** The prime factors $p$,$q$

1: Choose an integer $2^{2k-1} < \hat{m} < 2^{2k}$
2: Compute $\hat{c} \equiv \hat{m}^2 \pmod{N}$
3: Ask the decryption of $\hat{c}$ from Algorithm 2.3
4: Algorithm 2.3 output $m < 2^{2k-1}$, else reject
5: Compute $\gcd(\hat{m} + m, N)$
6: If $\gcd(\hat{m} + m, N) = 1$, then reject
7: If $\gcd(\hat{m} + m, N) \neq 1$, then return $p^2$
8: Compute $\frac{N}{p^2} = q$
9: Return the prime factors $p, q$

## 3.2 Computational Equivalent

If a new cryptosystem is designed, we are expected to provide a comparison of the relative difficulty of breaking the scheme into the solving any existing hard problems. Now, we show that breaking the Rabin-$p$ cryptosystem is indeed reducible to factoring the modulus $N = p^2q$. Furthermore, the converse of such statement is also true.

**Lemma 3.1.** *Breaking the Rabin-p cryptosystem is reducible to factoring* $N = p^2q$.

**Proof.** Suppose there exists an algorithm $\mathcal{A}_1$ with the ability to factor the modulus $N = p^2q$, then we obtain the primes $p$ and $q$. Thus, we can solve the Rabin-$p$'s ciphertext $c \equiv m^2 \pmod{N}$ directly by using the Algorithm 2.3. □

**Lemma 3.2.** *Factoring $N = p^2q$ is reducible to breaking the Rabin-$p$ cryptosystem.*

**Proof.** Conversely, suppose there exists an algorithm $\mathcal{A}_2$ that breaks the Rabin-$p$ cryptosystem. Then such algorithm is able to find the message $m$ from the ciphertext $c \equiv m^2 \pmod N$. By using the same approach as Theorem 3.1, hence $\mathcal{A}_2$ can proceed to compute $\hat{m}$. Finally, with the help of Algorithm 3.1, $\mathcal{A}_2$ can easily factor the modulus $N = p^2q$. □

**Theorem 3.2.** *Breaking the Rabin-$p$ cryptosystem is equivalent to factoring the modulus $N = p^2q$.*

**Proof.** This assertion is a consequence from Lemma 3.1 and Lemma 3.2. □

## 3.3 Analysis via Continued Fraction's Method

We begin with the definition of the continued fractions, which serves as a very useful mathematical tool and has been applied in many cryptanalytic works.

**Definition 3.1** (Continued Fractions)**.** *Hardy and Wright (1965). The continued fraction of a real number $R \in \mathbb{R}$ is an expression of the form*

$$R = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}} \tag{1}$$

*where $a_0 \in \mathbb{Z}$ and $a_i \in \mathbb{N} - \{0\}$ for $i \geq 1$. The numbers $a_0, a_1, a_2, \ldots$ are called the partial quotients. The equation (1) can be denoted as $R = [a_0, a_1, a_2, \ldots]$ and are called the convergents of the continued fraction expansion of $R$. If $R$ is a rational number then the continued fraction expansion of $R$ is finite.*

Following this definition is an important theorem of the continued fraction which be used widely throughout this proposal. This theorem simply says, the

unknown integers $x$ and $y$ can be recovered from the list of continued fraction expansion of a rational number $R$ satisfying the given inequality.

**Theorem 3.3** (Legendre's Theorem)**.** *Hardy and Wright (1965) Let $R$ is a rational number. Let $x, y \in \mathbb{Z}, y \neq 0$ and $\gcd(x, y) = 1$. Suppose*

$$\left| R - \frac{x}{y} \right| < \frac{1}{2y^2}$$

*Then $\frac{x}{y}$ is a convergent of the continued fraction expansion of $R$.*

We outline the analysis by continued fraction's method as follows. Suppose $c$ and $N$ are the parameters from the Rabin-$p$ cryptosystem. Since we have the ciphertext $c \pmod N$, thus $c < N$. Therefore $c$ can be written as $c = a + bpq$ or $c = a' + b'p^2$ for some integer $a, a', b, b'$.

**Theorem 3.4.** *(Asbullah and Ariffin, 2017) Let $c = a + bpq$ for some positive integer $a$ and $b$. If $a < \frac{q}{2}$ and $b < p$, then $\frac{b}{p}$ is a convergent of the continued fraction expansion of $\frac{c}{N}$.*

**Theorem 3.5.** *(Asbullah and Ariffin, 2017) Let $c = a' + b'p^2$ for some positive integer $a'$ and $b'$. If $a' < \frac{p^2}{2q}$ and $b' < q$, then $\frac{b'}{q}$ is a convergent of the continued fraction expansion of $\frac{c}{N}$.*

## 3.4 Analysis via Coppersmith's Method

Coppersmith (1997) invented a significantly powerful method for finding small roots of modular polynomial equations. This method has found many different applications in the area of cryptography and vastly useful tool for cryptanalysis Galbraith (2012). We now reproduce Coppersmith's theorems for the benefit of the reader.

**Theorem 3.6.** *Coppersmith (1997) Let $N$ be an integer of unknown factorization. Let $f_N(x)$ be a univariate, a monic polynomial of degree $\delta$. Then we can find all solutions $x_0$ for the equation $f_N(x) \equiv 0 \pmod N$ with $|x_0| < N^{1/\delta}$ in polynomial time.*

**Theorem 3.7.** *May (2003) Let $N$ be an integer of unknown factorization, which has a divisor $b > N^{\beta}$ . Furthermore, let $f_b(x)$ be a univariate, a monic polynomial of degree $\delta$.Then we can find all solutions $x_0$ for the equation $f_b(x) \equiv 0 \pmod{b}$ with $|x_0| < \frac{1}{2}N^{\beta^2/\delta}$ in polynomial time.*

We now analyze the Rabin-$p$ cryptosystem based on the Theorem 3.6 and Theorem 3.7 and obtain the following results. Suppose $c, m$ and $N$ are the parameters from the Rabin-$p$ cryptosystem.

**Theorem 3.8.** *(Asbullah and Ariffin, 2017) Let $c \equiv m^2 \pmod{N}$ and $N = p^2q$. If $m < 2^{3k/2}$ then $m$ can be found in polynomial time.*

**Proof.** Suppose $c \equiv m^2 \pmod{N}$ and $N = p^2q$. Consider the univariate, monic polynomial $f_N(x) \equiv x^2 - c \equiv 0 \pmod{N}$. By applying Theorem 3.6 we set $\delta = 2$. Hence the root $x_0 = m$ can be recovered if $m < N^{1/\delta} = N^{1/2} \approx 2^{3k/2}$. □

**Theorem 3.9.** *(Asbullah and Ariffin, 2017) Let $c \equiv m^2 \pmod{p^2}$ such that $p^2$ is an unknown factor for $N$. If $m < 2^{2k/3}$ then $m$ can be found in polynomial time.*

**Proof.** Suppose $c \equiv m^2 \pmod{p^2}$ such that $p^2$ is an unknown factor for $N$. Consider $f_{p^2}(x) \equiv x^2 - c \equiv 0 \pmod{p^2}$ with $p^2 \approx N^{2/3} \approx 2^{2k}$. We can find a solution $x_0 = m$ if $m < \frac{1}{2}N^{\beta^2/\delta} < N^{(2/3)^2/2} = N^{2/9} \approx 2^{2k/3}$. □

**Remark 3.2.** *Therefore in order to avoid both attacks, we would set $m > 2^{3k/2}$ in the Rabin-$p$ encryption algorithm.*

## 3.5 Resistant to Novak's Attack

In general, the decryption algorithm of a Rabin-like cryptosystem consists of two parts. The first part is for the modular exponentiation operation of which in order to obtain the message in the form of $m$ modulo $p$ and $m$ modulo $q$ from its corresponding ciphertext $c$. The second part then would be the recombination process using the Chinese Remainder Theorem (CRT) algorithm to

recover the proper message $m$. Most side channel attacks deal with the first part. For instance, the work by Kocher (1996), Schindler (2000) and Brumley and Boneh (2005) which uses the timing attack approach or the result in Messerges et al. (1999) enables side channel attack using the power analysis approach.

Alternatively, Novak (2002) proposed a very efficient side channel attack upon the CRT computation (i.e. the second part of the Rabin-like decryption). We observe that all variants of the Rabin-like cryptosystem (except Rabin-Williams scheme) involves a process that hardly depends on the CRT or Garner's algorithm (i.e. the process to recover all the modulo square roots). Therefore, Novak's attack is indeed applicable for such computation, of which can result in the insecurity of the cryptosystems Okeya and Takagi (2006).

**Remark 3.3.** *We reason that since our proposed scheme does not need to carry out any CRT computation, thus the Novak's attack is not applicable on the Rabin-p cryptosystem (i.e. resilient against Novak's attack).*

### 3.6 Resistant to Chosen Ciphertext Attack

Notice that the factoring algorithm mentioned by the Algorithm 3.1 could provide a way to launch a chosen ciphertext attack upon the proposed scheme in polynomial time, hence resulting in the system totally insecure in this sense. Therefore, to provide security against this kind of attack, we could consider implementing as a Key Encapsulation Mechanism (KEM) following the KEM framework for Rabin cryptosystem as proposed in Dent (2003). We will discuss this issue further in details in Section 5.

## 4 COMPARATIVE ANALYSIS

This chapter gives comparison of the basic scheme of Rabin-$p$ cryptosystem and other existing implementable, standardized public key encryption (basic) schemes that are based on the intractibility of the integer factorization problem;

namely the HIME(R), Rabin-SAEP+ and RSA-OAEP.

### 4.1 Security Level and Key Lengths

For the primes $p, q$ of the Rabin-$p$ cryptosystem should be chosen to be intractable to factor the modulus of $N = p^2q$. We choose the NIST Recommendations (2016) Giry (2017) for factoring modulus which present the appropriate key length for user's desired level of protection, as follows. Note that for good protection against quantum computers, the modulus size of 15360-bit is sufficent, unless Shor's algorithm applies Giry (2017).

| Date | Security Level | Modulus Size (bits) | Prime Size (bits) |
|---|---|---|---|
| 2016 - 2030 (& beyond) | 128 | 3072 | 1024 |
| 2016 - 2030 (& beyond) | 192 | 7680 | 2560 |
| 2016 - 2030 (& beyond) | 256 | 15360 | 5120 |

**Table 1:** Recommendation modulus length for Rabin-$p$ cryptosystem

We suppose that the bit-length $k$ of the modulus $N = p^2q$ for Rabin-$p$ and HIME(R) and the bit-length $K$ of the modulus $N = pq$ for Rabin-SAEP+ and RSA-OAEP have been selected so that the security level of these moduli against integer factorization attacks is the same. The bit-length of the prime factors of a Rabin-$p$ or HIME(R) $k$-bits modulus is denoted by $t$ (so $t = \frac{k}{3}$), while the bit-length of the prime factors of an RSA-OAEP or Rabin-SAEP+ $K$-bits modulus is denoted by $T$ (so $T = \frac{K}{2}$). Hence we have the comparative tables as follows.

| Algorithm | Modulus length | Public key | Private key |
|---|---|---|---|
| Rabin-$p$ | $N = p^2q$ | $N$ | $p$ |
| HIME(R)Hitachi (2002) | $N = p^2q$ | $N$ | $p, q$ |
| Rabin-SAEP+Shoup (2002) | $N = PQ$ | $N$ | $P, Q$ |
| RSA-OAEPBellare and Rogaway (1995) | $N = PQ$ | $N, e$ | $P, Q, d_P, d_Q$ |

**Table 2:** Key bit length vs HIME(R), Rabin-SAEP+ and RSA-OAEP

| Algorithm | Modulus length | Public key | Private key |
|---|---|---|---|
| Rabin-$p$ | 3072 | 3072 | 1024 |
| | 7680 | 7680 | 2560 |
| | 15360 | 15360 | 5120 |
| HIME(R)Hitachi (2002) | 3072 | 3072 | 2048 |
| | 7680 | 7680 | 5120 |
| | 15360 | 15360 | 10240 |
| Rabin-SAEP+Shoup (2002) | 3072 | 3072 | 3072 |
| | 7680 | 7680 | 7680 |
| | 15360 | 15360 | 15360 |
| RSA-OAEPBellare and Rogaway (1995) | 3072 | 3072∼6144 | 6144 |
| | 7680 | 7680∼15360 | 15360 |
| | 15360 | 15360∼30720 | 30720 |

**Table 3:** Modulus, Public key(s) and Private key(s) of Rabin-$p$, HIME(R), Rabin-SAEP+ and RSA-OAEP

## 4.2 Performance Efficiency

In this section, we compare the speed of Rabin-$p$ when compared to HIME(R), Rabin-SAEP+ and RSA-OAEP through its most fundamental complexity order (i.e. basic textbook operation speed without any enhancement). As a note, any enhancement for the benchmark algorithms will result also in Rabin-$p$ cryptosystem using the enhanced operation mechanism.

### 4.2.1 Encryption

The computational steps that dominate the execution time of the encryption process for the Rabin-$p$, HIME(R), Rabin-SAEP+ and RSA-OAEP are:

1. Rabin-$p$: $m^2 \pmod N$. That is, a modular squaring operation with a $k$-bit modulus.
2. HIME(R): $m^2 \pmod N$. That is, a modular squaring operation with a

$k$-bit modulus.

3. Rabin-SAEP+: $m^2 \pmod N$. That is, a modular squaring operation with a $K$-bit modulus.
4. RSA-OAEP: $m^e \pmod N$. That is, a modular exponentitation operation with a $K$-bit modulus.

### 4.2.2 Decryption

The computational steps that dominate the execution time of the decryption process for the Rabin-$p$, HIME(R), Rabin-SAEP+ and RSA-OAEP are:

1. Rabin-$p$: $c^{\frac{p+1}{4}} \pmod p$. That is, one modular exponentiations with $t$-bit modulus.
2. HIME(R): $c^{\frac{p+1}{4}} \pmod p$ and $c^{\frac{q+1}{4}} \pmod q$. That is, two modular exponentiations with $t$-bit moduli.
3. Rabin-SAEP+:$c^{\frac{P+1}{4}} \pmod P$ and $c^{\frac{Q+1}{4}} \pmod Q$. That is, two modular exponentiations with $T$-bit moduli.
4. RSA-OAEP: $c^{d_P} \pmod P$ and $c^{d_Q} \pmod Q$. That is, two modular exponentiations with $T$-bit moduli.

## 4.3 Complexity Comparison

| Algorithm | Encryption Complexity | Decryption Complexity |
|---|---|---|
| Rabin-$p$ | $O(n^2)$ | $O(n^3)$ |
| HIME(R) | $O(n^2)$ | $O(n^3)$ |
| Rabin-SAEP+ | $O(n^2)$ | $O(n^3)$ |
| RSA-OAEP | $O(n^3)$ | $O(n^3)$ |

**Table 4:** Performance efficiency between the Rabin-$p$, HIME(R), Rabin-SAEP+ and RSA-OAEP

## 4.4 Plaintext to Ciphertext Ratio

Message expansion is another angle where comparison can be made. This area is closely related to bandwith overhead. The larger the expansion the more bandwith is utilized. We provide a table for comparison against the HIME(R), Rabin-SAEP+ and RSA-OAEP. Plaintext to ciphertext ratio is denoted as $m : c$.

| Algorithm | $m : c$ |
|---|---|
| Rabin-$p$ | $2 : 3$ |
| HIME(R)Hitachi (2002) | $\sim 3 : 4$ |
| Rabin-SAEP+Shoup (2002) | $1 : 4$ |
| RSA-OAEPBellare and Rogaway (1995) | $\sim 3 : 4$ |

**Table 5:** Plaintext to Ciphertext Ratio vs HIME(R), Rabin-SAEP+ and RSA-OAEP

## 4.5 Conclusion

The ability of Rabin-$p$ cryptosystem to have the following characteristics:

1. Key length comparable to currently deployed public key encryptions algorithms;
2. Fast performance during encryption and decryption;
3. Fair message expansion rate;
4. Does not have decryption failure,

makes Rabin-$p$ cryptosystem a possible candidate for a secure national encryption scheme. Moreover, with the beneficial features that the Rabin-$p$ has, the possibility of seamless deployment within current public key infrastructure cannot be ruled out. Additionally, for good protection against quantum computers, the modulus size of 15360-bit is sufficent, unless Shor's algorithm applies.

# 5 RABIN-$P$ KEY ENCAPSULATION MECHANISM: THE PROPOSAL

The security of a modern public key cryptosystem is usually viewed from their mathematical hard problem and its security model. In this section, we propose the design for Rabin-$p$ cryptosystem in the setting of Key Encapsulation Mechanism (KEM) following the KEM framework for Rabin cryptosystem as proposed in Dent (2003).

## 5.1 Preliminaries

In order to facilitate fundamental flow of knowledge, we lay down some definitions. We begin with important definitions concerning with the material of related cryptographic hard problems. Secondly, we outline our security model.

### 5.1.1 Related Cryptographic Hard Problem

**Definition 5.1** (Cryptographic Hard Problem)**.** *(Menezes et al., 1997). A cryptographic hard problem is defined as a concrete mathematical object which is easily to compute in one direction, but very hard to invert.*

**Definition 5.2** (Negligible Function)**.** *(Katz and Lindell, 2008). A function $\epsilon$ is negligible if for every polynomial with integer coefficients $f(\cdot)$ there exists an $N > 0$ such that for all integers $n > N$ it holds that $\epsilon(n) < \frac{1}{f(n)}$.*

Let $\mathcal{A}$ be a probabilistic polynomial time algorithm and a probability denoted as $\Pr$. Then we have the following definitions.

**Definition 5.3** (Integer Factorization Problem)**.** *(Hoffstein et al., 2008). Let $N$ be a positive integer. Then, the integer factorization problem (IFP) is defined as the problem to find the prime factorization of $N$ such that, $N = p_1^{r_1} p_2^{r_2} p_3^{r_3} \dots p_s^{r_s}$ where $p_i$ are distinct primes and $r_i \geq 1$. For our case, the problem is to find the prime factors $p$ and $q$ from $N = p^2 q$.*

**Definition 5.4** (IFP Hard Problem)**.** *(Katz and Lindell, 2008). Let the IFP is defined as in Definition 5.3 with the particular modulus such that $N = p^2q$. Suppose $[\mathcal{A}_{(IFP)} = 1]$ is an event such that $\mathcal{A}$ is successfully factor $p$ and $q$ given $N = p^2q$, otherwise $[\mathcal{A}_{(IFP)} = 0]$. We say that IFP (i.e. factoring $N = p^2q$) is hard if for all probabilistic polynomial time algorithm $\mathcal{A}$ there exists a negligible function $\epsilon$ such that*

$$\Pr[\mathcal{A}_{(IFP)} = 1] \leq \epsilon$$

**Definition 5.5** (Rabin-$p$ Hard Problem)**.** *Let the Rabin-$p$ cryptosystem is as defined as in Section 2. Suppose $[\mathcal{A}_{(Rabin-p)} = 1]$ is an event such that $\mathcal{A}$ successfully invert the Rabin-$p$ cryptosystem and obtained the correct message $m$, otherwise $[\mathcal{A}_{(Rabin-p)} = 0]$. As proven in Theorem 3.2 that breaking Rabin-$p$ cryptosystem is equivalent to factoring the modulus $N = p^2q$, thus $\Pr[\mathcal{A}_{(Rabin-p)} = 1] = \Pr[\mathcal{A}_{(IFP)} = 1]$. We say that breaking the Rabin-$p$ cryptosystem is hard relative to IFP (i.e. Definition 5.4) if for all probabilistic polynomial time algorithm $\mathcal{A}$ there exists a negligible function $\epsilon$ such that*

$$\Pr[\mathcal{A}_{(Rabin-p)} = 1] \leq \epsilon$$

### 5.1.2 Security Goals and Attack Models

The security of public key cryptosystem is usually categorized from the point of view of their goals and attack models. The currently known standard goals of public key cryptosystems are defined as follows.

**Definition 5.6** (Indistinguishability)**.** *(Goldwasser and Micali, 1984). Indistinguishability (IND) refers to the situation of given a ciphertext of one of the two plaintexts (i.e. both plaintexts known to the adversary), and then any adversary cannot distinguish which one is encrypted. This notion is rather artificial, but in considering provable security of a public key cryptosystem it is usually convenient to employ this notion as the goal of the system.*

On the other hand, the currently known standard attack models upon a public key cryptosystem are as follows.

**Remark 5.1** (Chosen Plaintext Attacks (CPA))**.** . *In this model, an adversary has access to an encryption oracle. That is, such adversary can choose a set of plaintexts and obtain the corresponding ciphertexts.*

**Remark 5.2** (Non-adaptive Chosen Ciphertext Attacks (CCA1))**.** . *In this model, an adversary has, in addition to the ability to the CPA adversary, access to a decryption oracle before obtains a challenge ciphertext. That is, the adversary can choose a set of ciphertexts and obtain the corresponding plaintexts during this period (Naor and Yung, 1990).*

**Remark 5.3** (Adaptive Chosen Ciphertext Attacks (CCA2))**.** . *In this model, an adversary has, in addition to the ability of the CCA1 adversary, access to a decryption oracle even after obtaining the challenge ciphertext. However, this kind of adversary is prohibited from asking the oracle to decrypt the challenge ciphertext itself (Rackoff and Simon, 1992).*

Several security notions can be constructed by combining these goals and attack models, and, of course, there are relations between some of these notions. In fact, the following facts on such relations have been known so far Watanabe et al. (2002). First, regarding the attack models, the power of the adversaries gets stronger in the order CPA, CCA1, and CCA2, so does the strength of the security notions. It is largely agreed upon that security against CCA2 is one of the most important attributes of any public key cryptosystem (Müller, 2001).

Secondly, in proposing a public key cryptosystem, it is conventional to claim that the public key cryptosystem has the strongest security by showing that it is secure in the sense of indistinguishability against chosen ciphertext attacks (IND-CCA2). For instance see Bellare and Rogaway (1995), and Cramer and Shoup (2003). Hence, formalizing and proving for any designated public key cryptosystem resilient to such stronger attack model is very important.

### 5.1.3 Deterministic Encryption

We will start by considering deterministic encryption schemes.

**Definition 5.1** (A Deterministic Encryption Scheme Dent (2003))**.** *A deterministic encryption scheme is a triple* $(G, E, D)$ *where:*

1. *an encryption algorithm, $E$, which takes as input a message $m \in \mathcal{M}$ and the public key $pk$ and outputs a ciphertext $C \in \mathcal{C}$,*

2. *the decryption algorithm, $D$, which takes as input a ciphertext $C \in \mathcal{C}$ and and the secret-key $sk$ and outputs either a message $m \in \mathcal{M}$ or the error symbol $\perp$.*

The weakest notion of security for a deterministic encryption scheme is one-way security.

**Definition 5.2.** *A deterministic encryption scheme $(G, E, D)$ is said to be one-way if the probability that a polynomial time attacker $A$ can invert a randomly generated ciphertext $C = E(m, pk)$ (where $m$ is chosen at random from $\mathcal{M}$ is negligible as a function of $k$. Such a cryptosystem is often said to be secure in the OW-CPA model.*

### 5.1.4 Key Encapsulation Mechanism - KEM

Technically, to use the public key systems in sending long messages is not practical. Instead, they are frequently applied to exchange, symmetric keys, which are comparatively short (Abe et al., 2008). The symmetric key is then employed to encrypt the longer messages. The public key cryptosystem is somehow relatively slow compared to its symmetric counterpart; thus it is not suited for encrypting large bulk of information.

Essentially, Dent (2003) gives a generic construction method to allow an algorithm designer to construct a KEM from almost any cryptographic problem. As a result, we propose a Rabin-$p$ KEM, that is as secure as factoring, in the random oracle model. Firstly, we recall the definition of the random oracle model as follows.

**Definition 5.3** (Random Oracle Model (Katz and Lindell, 2008))**.** *A random oracle is a function $H(\cdot) : \{0,1\}^n \longrightarrow \{0,1\}^n$ that maps an input value to a true random output value.*

In the random oracle model (ROM), one assumes that some hash function is replaced by a random function accessible to the public. This means that the adversary cannot calculate the result of the hash function itself, instead he must query the random oracle. This also means that anyone, including the adversary has access to the random oracle Coron et al. (2008).

**Definition 5.4** (Key Encapsulation Mechanism (Dent, 2003))**.** *A KEM is a triple of algorithms:*

1. *a key generation algorithm, $KEM.Gen$, which takes as input a security parameter $1^k$ and outputs a public/secret key-pair $(pk, sk)$,*
2. *an encapsulation algorithm, $KEM.Encap$, that takes as input a public key $pk$ and outputs an encapsulated key-pair $(K, C)$ (i.e. $C$ is sometimes said to be an encapsulation of the key $K$),*
3. *a decapsulation algorithm, $KEM.Decap$, that takes as input an encapsulation of a key $C$ and a secret-key $sk$, and outputs a key $K$.*

We choose to approach provable security from an asymptotic point of view and suggest that a scheme is secure if the probability of breaking that scheme is negligible as a function of the security parameter.

## 5.2 The Proposal for Rabin-$p$ KEM

### 5.2.1 The Security of Rabin-$p$ Encryption

Clearly, Rabin-$p$ does not achieve IND-CPA because Rabin-$p$ encryption algorithm as shown in Chapter 2 is deterministic. Next we discuss the onewayness (OW) and unbreakability (UB) of Rabin-$p$.

As described and discussed in Section 2, the onewayness for Rabin-$p$ scheme or the Rabin-$p$ decryption problem is: Given public key $N$ and ciphertext $c$,

find $m$ such that $E(N, m) \equiv m^2 \pmod{N} \equiv c$. Section 3 have proven that under CPA the **Rabin-$p$ decryption problem** is reduced to the integer factorization problem (IFP). The proof includes an algorithm (See Section 3.2) which chooses and encrypts a message which is larger than $p^2$ and queries it to the OW adversary. The adversary then returns a message less than $p^2$. Utilizing the Euclidean algorithm on the two distinct messages enable the factoring of the public key $N$. Let this algorithm (i.e. Algorithm 3.1) be named Rabin-$p$ factoring algorithm. By the proofs of Theorem 3.2 and by Definition 5.2, hence the Rabin-$p$ encryption achieves OW-CPA assuming that integer factorization is hard.

Furthermore, from the public key of Rabin-$p$, which is in the form of $N = p^2q$ where $p$ and $q$ are $k$-bit primes and $p, q \equiv 3 \pmod{4}$. The private key is the prime $p$. Hence the **Rabin-$p$ private key problem** can be stated as: Given the public key, $N$, find the private key, a $k$-bit prime $p$ such that $p^2$ divides $N$. As such, the Rabin-$p$ private key problem is exactly the integer factorization problem under CPA and this is correctly proven in the previous section. Hence, Rabin-$p$ is UB-CPA, assuming integer factorization is hard.

## 5.3 Generic construction of secure KEM

Dent (2003) propose a simpler construction for designing a KEM based on a deterministic encryption scheme with weak security assumptions.In other words, a secure KEM is build from a deterministic encryption scheme that is secure in the OW-CPA model. The following Algorithm 5.1, Algorithm 5.2 and Algorithm 5.3, gives a construction of a KEM based on a deterministic asymmetric encryption scheme $(G, E, D)$. The scheme makes use of a key derivation function $KDF$ and a hash function $Hash$. These functions will be modelled as random oracles and so care must be taken that their outputs are suitably independent.

**Algorithm 5.1** Key Generation of a KEM derived from an OW-CPA secure, deterministic encryption scheme

1: Key-generation is given by $G$, i.e. $KEM.Gen = G$

**Algorithm 5.2** Encapsulation of a KEM derived from an OW-CPA secure, deterministic encryption scheme

1: Generate a suitably large bit-string $x \in \mathcal{M}$.
2: Set $C_1 := \mathcal{E}(x, pk)$
3: Set $C_2 := Hash(x)$
4: Set $C := (C_1, C_2)$
5: Set $K := KDF(x)$
6: Output $(K, C)$

**Algorithm 5.3** Decapsulation of a KEM derived from an OW-CPA secure, deterministic encryption scheme

1: Parse $C$ as $(C_1, C_2)$.
2: Set $x := \mathcal{D}(C_1, sk)$. If $x = \perp$ then output $\perp$ and halt.
3: Check that $C_2 = Hash(x)$. If not, output $\perp$ and halt.
4: Set $K := KDF(x)$
5: Output $K$

This construction also has the advantage that the decryption algorithm need not return a unique solution but need only return a small subset of the message space that includes the original message, as, with high probability, the original message will be the only message in the subset that hashes to give the correct value of $C_2$. We will make heavy use of this fact in the specifIcation of Rabin-$p$ KEM.

**Theorem 5.1** (Dent (2003))**.** *Suppose that (G, E, D) is a deterministic encryption algorithm that is secure in the OW-CPA model. Then the KEM derived from (G, E, D) in Table 4 is, in the random oracle model, IND-CCA2 secure.*

**Proof.** Appendix B of Theorem 4 in Dent (2003) □

## 5.4 The Design of Secure Rabin-$p$ KEM

In this work, we will view the Rabin-$p$ as a KEM-DEM framework, and study only the KEM component. Security analysis for Rabin-$p$ KEM instead of a

hybrid scheme is more elegant because the KEM-DEM framework has specified the required security level for KEM relating directly to security of Rabin-$p$ scheme. This section presents the security of Rabin-$p$ as a KEM, following the KEM framework for Rabin as proposed in Dent (2003).

Now we are ready to present our KEM design for the Rabin-$p$ cryptosystem. The same procedure is retained for the key generation as described in Algorithm 2.1 and output the public key $N = p^2q$ and the private key $p$. We begin with the key generation algorithm as follows.

**Algorithm 5.4** Rabin-$p$ KEM Key Generation

**Input:** The size $k$ of the security parameter.
**Output:** The public key $N$ and the 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: Return the public key $N$ and the secret key $p$.

**Algorithm 5.5** Rabin-$p$ KEM Encapsulation Algorithm

**Input:** The public key $N$.
**Output:** A ciphertext tuple $(K, C)$.

1: Choose a random integer $2^{3k/2} < x < 2^{2k-1}$.
2: Compute $C_1 \equiv x^2 \pmod N$.
3: Compute $C_2 = Hash(x)$.
4: Set $C := (C_1, C_2)$
5: Set $K := KDF(x)$
6: Output $(K, C)$.

## 5.5 Security Proof for Rabin-$p$ KEM

We proposing a new KEM whose security is equivalent to factoring, that is the Rabin-$p$ KEM. The Rabin-$p$ KEM construction will be based on the generic construction given in Section 5.3 and the Rabin-$p$ encryption from Chapter 2. The algorithms of Rabin-$p$ KEM is described by Algorithm 5.4, Algorithm 5.5

**Algorithm 5.6** Rabin-$p$ KEM Decapsulation Algorithm

**Input:** A ciphertext $C$ and the 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 $x_p \equiv w^{\frac{p+1}{4}} \pmod{p}$.
4: Compute $i = \frac{c - x_p^2}{p}$.
5: Compute $j \equiv \frac{i}{2x_p} \pmod{p}$.
6: Compute $x_1 = x_p + jp$.
7: If $x_1 < 2^{2k-1}$, then return $x = x_1$. Else set $x = p^2 - x_1$.
8: Check that $C_2 = Hash(x)$. If not, output $\perp$ and halt.
9: Let $x$ be the unique square root of $C_1$ modulo $N$ for which $Hash(x) = C_2$.
10: Set $K := KDF(x)$
11: Output $K$.

and Algorithm 5.6, respectively. The provable security proof of the proposed Rabin-$p$ KEM can be summed up in the following theorem.

**Theorem 5.2.** *Providing the factoring problem is hard, Rabin-p KEM is IND-CPA secure in the random oracle model.*

**Proof.** It is proven in Theorem 3.2 that the Rabin-$p$ function is one-way providing that the factoring assumption is hard. Therefore, given that the factoring problem is intractable, by Theorem 5.1 the proposed Rabin-$p$ KEM is IND-CPA secure in the random oracle model. □

**Remark 5.4.** *Observe that, the Rabin-p cryptosystem falls prey to the integer factorization based-encryption security incompatibility in the same way as Rabin cryptosystem (Rabin, 1979). This incompatibility is first found by Williams (1980) in Rabin cryptosystem and was formally stated and proven in Paillier and Villar (2006). A simplified statement of the security incompatibility is: If an encryption scheme OW-CPA implies integer factorization problem, then the scheme is totally broken under CCA. Therefore, particularly in our case, it is necessary to reduce the security claims of Theorem 5.1 which originally proved for IND-CCA2 security to only achieve IND-CPA secure.*

# 6 SUGGESTED IMPLEMENTATION PRACTICES

## 6.1 Key Generation Procedure

A practical key generation methodology for factoring based cryptosystems are already established and well-developed. In implementing the Rabin-$p$ key generation procedure properly, we suggested the implementers to utilize the key generation mechanism provided in Giry (2017) and Shoup (2006) satisfying the condition within Section 2.2 in Section 2.

## 6.2 Rabin-$p$ Encryption Procedure

Section 2(in Section 2.3) and Section 3 (in Section 3.3, Section 3.4) lists out strict conditions for variables within Rabin-$p$ encryption procedures. These conditions have to be satisfied in order for Rabin-$p$ security properties to be realized.

## 6.3 Rabin-$p$ Decryption Procedure

For implementers that wish to optimize the decryption procedure, we suggest the the implementers to follow the mechanism as described in Section 2(in Section 2.4) and in Section 3.

The next subsection discuss the computational running time for both the encryption and decryption process for Rabin-$p$ cryptosystem.

## 6.4 Encryption Computational Running Time

The Rabin-$p$ encryption process involves a squaring and a modular reduction process. Its total running time is $O(14k^2 + 4k)$.

## 6.5 Decryption Computational Running Time

The Rabin-$p$ decryption process involves 1 modular exponentiation, 2 modulo reduction, 1 division over the integers, 1 modular inverse and 1 addition process. Its total running time is $O(3k^3 + 142k^2 + 154k + 3)$.

## 6.6 Empirical Performance Data

These experiments were conducted using Microsoft Visual Studio 2010 on ASUS Model G551J, Windows 8.1 with Intel(R) Core(TM) i7-4710HQ CPU 2.50GHz and 4.00GB RAM.

### 6.6.1 Rabin-$p$ Encryption

Table 6 shows the computational time of Rabin-$p$ encryption algorithm when executing on specific numbers of data.

| Number of data encrypted | Time (ms) |
|---|---|
| 100 | 13 |
| 500 | 76 |
| 1000 | 138 |
| 5000 | 717 |
| 10000 | 1430 |

**Table 6:** Rabin-$p$ encryption algorithm execution time.

### 6.6.2 Rabin-$p$ Decryption

Table 7 shows the computational time of Rabin-$p$ decryption algorithm when executing on specific numbers of data.

| Number of data decrypted | Time (ms) |
|---|---|
| 100 | 21 |
| 500 | 83 |
| 1000 | 156 |
| 5000 | 842 |
| 10000 | 1538 |

**Table 7:** Rabin-$p$ decryption algorithm execution time.

# 7 INTELLECTUAL PROPERTY STATEMENT FOR THE SUBMISSION OF RABIN-$P$ KEY ENCAPSULATION MECHANISM TO THE MYSEAL PROJECT

I do hereby declare that the cryptosystems that I have submitted, known as Rabin-$p$ Key Encapsulation Mechanism, are partially in publications before this proposal submission, as follows;

1. M A Asbullah & M R K Ariffin. Design of Rabin-like Cryptosystem Without Decryption Failure (2016). Malaysian Journal of Mathematical Science, 10(S), 1-18.
2. M A Asbullah, M R K Ariffin & Z Mahad (2016). Analysis on the Rabin-$p$ cryptosystem. AIP Conference Proceedings 1787, 080012.
3. M A Asbullah & M R K Ariffin (2016). Provably Secure Rabin-$p$ Cryptosystem in Hybrid Setting. AIP Conference Proceedings 1739, 020001.
4. M A Asbullah & M R K Ariffin. Algebraic Analysis of a Rabin-Like Cryptosystem and Its Countermeasures (2017). Indian Journal of Science and Technology, 10(1), 1-5.
5. M A Asbullah, Z Mahad & M R K Ariffin, Efficient Programming Deployment Strategy for Rabin-p Cryptosystem in C/C++, MyIPO Copyright Filing No. LY2018004528, 27 September 2018.

6. M A Asbullah, Z Mahad & M R K Ariffin, Efficient Programming Deployment Strategy for Rabin-p Cryptosystem in Java, MyIPO Copyright Filing No. LY2018004528, 27 September 2018.

Finally, I will undertake to update the MySEAL project when necessary.

# REFERENCES

Abe, M., Gennaro, R., and Kurosawa, K. (2008). Tag-KEM/DEM: A New Framework for Hybrid Encryption. *Journal Of Cryptology*, 21(1):97–130.

Asbullah, M. A. and Ariffin, M. R. K. (2016). Design of Rabin-like Cryptosystem without Decryption Failure. *Malaysian Journal of Mathematical Sciences*, 10(S):1–18.

Asbullah, M. A. and Ariffin, M. R. K. (2017). Algebraic Analysis of a Rabin-Like Cryptosystem and Its Countermeasures. *Indian Journal of Science and Technology*, 10(1):1–5.

Bellare, M. and Rogaway, P. (1995). Optimal Asymmetric Encryption. In *Advances In Cryptology - EUROCRYPT'94*, pages 92–111. Springer.

Boneh, D. (2001). Simplified OAEP For The RSA And Rabin Functions. In *Advances In Cryptology-Crypto 2001*, pages 275–291. Springer.

Brumley, D. and Boneh, D. (2005). Remote Timing Attacks Are Practical. *Computer Networks*, 48(5):701–716.

Castagnos, G., Joux, A., Laguillaumie, F., and Nguyen, P. Q. (2009). Factoring $pq^2$ With Quadratic Forms: Nice Cryptanalyses. In *Advances In Cryptology - ASIACRYPT 2009*, pages 469–486. Springer.

Coppersmith, D. (1997). Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities. *Journal Of Cryptology*, 10(4):233–260.

Coron, J.-S., Patarin, J., and Seurin, Y. (2008). The Random Oracle Model and the Ideal Cipher Model are Equivalent. In *Advances In Cryptology–Crypto 2008*, pages 1–20. Springer.

Cramer, R. and Shoup, V. (2003). Design and Analysis of Practical Public-Key Encryption Schemes Secure Against Adaptive Chosen Ciphertext Attack. *SIAM Journal On Computing*, 33(1):167–226.

Dent, A. (2003). A designers guide to kems. cryptography and coding, lncs 2898: 133–151.

Galbraith, S. D. (2012). *Mathematics Of Public Key Cryptography*. Cambridge University Press.

Galindo, D., Martỳn, S., Morillo, P., and Villar, J. L. (2002). A Practical Public Key Cryptosystem from Paillier and Rabin Schemes. In *Public Key Cryptography - PKC 2003*, pages 279–291. Springer.

Giry, D. (2017). NIST Recommendations on Key Length (2016). *https://www.keylength.com/en/4/*.

Goldwasser, S. and Micali, S. (1984). Probabilistic Encryption. *Journal Of Computer And System Sciences*, 28(2):270–299.

Hardy, G. and Wright, E. (1965). *An Introduction to the Theory of Numbers*. Oxford University Press, London.

Hitachi (2002). HIME(R) Public-Key Cryptosystem. *http://www.hitachi.com/rd/yrl/crypto/hime/*.

Hoffstein, J., Pipher, J., and Silverman, J. H. (2008). *An Introduction To Mathematical Cryptography*. Springer.

Katz, J. and Lindell, Y. (2008). *Introduction To Modern Cryptography: Principles And Protocols* . Chapman And Hall/ CRC Press.

Kocher, P. C. (1996). Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In *Advances In Cryptology - Crypto'96*, pages 104–113. Springer.

Kumanduri, R. and Romero, C. (1998). *Number theory with Computer Applications*. Prentice Hall New Jersey.

May, A. (2003). *New RSA Vulnerabilities Using Lattice Reduction Methods*. PhD thesis, University Of Paderborn.

Menezes, A., Oorschot, P., and Vanstone, S. (1997). *Handbook Of Applied Cryptography*. CRC Press.

Messerges, T. S., Dabbish, E. A., and Sloan, R. H. (1999). Power Analysis Attacks of Modular Exponentiation in Smartcards. In *Cryptographic Hardware And Embedded Systems - CHES'99*, pages 144–157. Springer.

Müller, S. (2001). On the Security of Williams Based Public Key Encryption Scheme. In *Public Key Cryptography*, pages 1–18. Springer.

Naor, M. and Yung, M. (1990). Public-Key Cryptosystems Provably Secure Against Chosen Ciphertext Attacks. In *Proceedings Of The Twenty-Second Annual ACM Symposium On Theory Of Computing*, pages 427–437. ACM.

Novak, R. (2002). SPA-Based Adaptive Chosen-Ciphertext Attack on RSA Implementation. In *Public Key Cryptography*, pages 252–262. Springer.

Okamoto, T. and Uchiyama, S. (1998). A New Public-Key Cryptosystem as Secure as Factoring. In *Advances In Cryptology - EUROCRYPT'98*, pages 308–318. Springer.

Okeya, K. and Takagi, T. (2006). Security Analysis of CRT-Based Cryptosystems. *International Journal Of Information Security*, 5(3):177–185.

Paillier, P. (1999). Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In *Advances In Cryptology - EUROCRYPT'99*, pages 223–238. Springer.

Paillier, P. and Villar, J. L. (2006). Trading one-wayness against chosen-ciphertext security in factoring-based encryption. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 252–266. Springer.

Rabin, M. O. (1979). Digitalized Signatures and Public-Key Functions as Intractable as Factorization. *MIT Technical Report*, MIT/LCS/TR-212.

Rackoff, C. and Simon, D. R. (1992). Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. In *Advances In Cryptology - Crypto'91*, pages 433–444. Springer.

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.

Schindler, W. (2000). A Timing Attack Against RSA with the Chinese Remainder Theorem. In *Cryptographic Hardware And Embedded Systems - CHES 2000*, pages 109–124. Springer.

Schmidt-Samoa, K. (2006). A New Rabin-Type Trapdoor Permutation Equivalent To Factoring. *Electronic Notes In Theoretical Computer Science*, 157(3):79–94.

Shoup, V. (2002). Oaep reconsidered. *Journal of Cryptology*, 15(4):223–249.

Shoup, V. (2006). ISO 18033-2: A Standard for Public-Key Encryption. *http://www.shoup.net/iso/*.

Watanabe, Y., Shikata, J., and Imai, H. (2002). Equivalence Between Semantic Security and Indistinguishability Against Chosen Ciphertext Attacks. In *Public Key Cryptography - PKC 2003*, pages 71–84. Springer.

Williams, H. (1980). A Modification of the RSA Public-Key Encryption Procedure. *IEEE Transactions On Information Theory*, 26(6):726–729.