Application of Shor's Algorithm in Finding RSA Decryption Key

Nur Alam Hasabie - 13517096
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13517096@std.stei.itb.ac.id

***Abstract*—RSA encryption is an widely applied asymmetric encryption method. The strength of RSA lies in the large integer prime factorization problem, which is solved fastest by general number field sieve in super-polynomial time. Quantum computation provides a quantum algorithm, called Shor's algorithm, that is able to solve the problem in exponential time complexity, thus putting RSA key at risk. However, implementation of quantum algorithm and quantum algorithm faces some challenges, while post-quantum key RSA has been proposed to be improvable.**

***Keywords*— cryptography, quantum computing, RSA key, Shor's algorithm.**

I. Introduction

The secrecy of information has been a concern for humans in a long time. The oldest record of cryptography can be traced to ancient Egyptians in around 1900 B.C., and used extensively by civilizations for various purposes.[1]

However old it is, the study of cryptography had only been extensively studied in these past few decades, mainly due to advances in Mathematics and demands to encrypt sensitive data in the ever-growing Internet. One encryption method which is widely used is RSA. The power of RSA encryption lies in the fact that to find two prime factors of a certain number, usually in hundreds or even thousand of digits, is a really tedious task, if not impossible, even for modern computers.[2]

In 1994, Peter Shor from MIT proposed an algorithm which could factorize big numbers efficiently. Shor's Algorithm employs some fundamental properties of quantum computing (superposition to be exact) to solve the problem. The algorithm, therefore, cannot be applied to classical computer. However, until 2014, the biggest number to be factorized with quantum computing is 56,513, a number much smaller than numbers used for RSA encryption.[3] Furthermore, quantum computing is area which still require further studies and researches , therefore RSA encryption will still be useful for quite a long time.

This publication will briefly discuss about RSA encryption, basic quantum computing terminologies, history of Shor's Algorithm and the algorithm itself, and also extending the topic to some challenges in implementing such algorithm and improvement of RSA encryption post-quantum.

II. Theories

A. Number Theory : Modular Arithmetic

It is important to review the basic of number theorem to understand RSA encryption and few important steps in Shor's Algorithm. The first notation of integer congruence appeared in Gauss' *Disquisitiones Arithmeticae.* Let m be an integer, For a, b $\in \mathbb{Z}$, a ≡ b mod m and it can be stated that "a is congruent to b mod m" if m | (a-b) ((a-b) is divisible by m).

Due to its definition , there are several theorems describing modular arithmetic (note that only the most fundamental theorems will be written, otherwise is omitted) :

1. Let m $\in \mathbb{Z}$ be a nonzero integer. For each a $\in \mathbb{Z}$, there is a unique r with a ≡ r mod m and $0 \leq r < |m|$.
2. If a ≡ b mod m and c ≡ d mod m, then a + c ≡ b + d mod m and ac ≡ bd mod m.
3. If a ≡ b mod m and b ≡ c mod m then a ≡ c mod m. [4]

B. RSA Encryption

RSA Encryption is a method of encrypting message introduced by Ron Rivest, Adi Shamir and Leonard Adleman in 1978, first published in the *Communications of the Association for Computing Machinery*.

The steps of encryption is as the following [2] :

1. Choose a number n = pq, where p and q both are primes. the value n is public key, therefore can be published.
2. Find a number e, where gcd(e,φ(n)) = 1.(Note : gcd(x,y) is the greatest common divisor of x and y). φ(n) is the Euler totient function (number of positive integer less than n which are relatively prime to n , with 1 is relatively prime to all number). This number is also a public key, therefore it can also be published
3. Find a number d, such as ed = 1 mod (φ(n)). In other word, d is the multiplicative modular inverse of n. such number d exists if and only if e and φ(n) are relatively prime.

Let there be a message M . To get the encrypted message C from the initial message M, then $C \equiv M^e \bmod n$. To get the original message M from the encrypted message C, then $M \equiv C^d \bmod n$. The decryption is ensured by Euler's Theorem, extension of Fermat's Little Theorem to general modulus :

$$\text{For } m \geq 2 \text{ in } Z^+ \text{ and any } a \in Z \text{ such that } (a, m) = 1,\quad a^{\phi(m)} \equiv 1 \bmod m \tag{1}$$

The power of this encryption lies in the fact that to find the appropriate value of d , the value of φ(n) must be determined first. According to the property of φ(n), φ(n) = (p-1)(q-1) for prime p and q. Therefore, one should find the values of both p and q to be able to decrypt the message.

The simplest method to find the value of p and q is by brute force, which is by trying all numbers below the square root of n.(such algorithm is called as trial division). However, there are already several algorithms found for faster integer factorization, such as Pollard's rho algorithm, general number field sieve etc. The best algorithm with asymptotic behavior known is general number field sieve, with time complexity of $\exp[c(\ln n)^{1/3} (\ln(\ln n))^{2/3}]$ in Big-O notation.

To illustrate the difficulty of factoring certain integer n ,it is stated in a paper published by a team of international researchers in IACR (International Association for Cryptologic Research) in 2010 that it is possible to factorize 768-bit RSA key, but such factorization requires instructions of order 2^{67} to be carried on. Such large number of operations can be operated for 1500 years in a standard desktop (single core 2.2GHz AMD Opteron with 2GB RAM). [5] The study then suggested that keys with digits fewer than 768 are not to used again. However, RSA-1024 and RSA-2048 is the standard widely used in the current years, meaning that it will require more operations to break the encryption.

RSA is commonly applied, with some implementation of includes RSA is Amazon Elastic Compute Cloud (Amazon EC2) key pair and OpenSSH(Secure Shell) protocol.

B . Quantum Computing

A simple definition, a quantum computing is a computational method that employs quantum properties (such as entanglement, superposition). A quantum computer , therefore, is a machine operating based on quantum computing properties. However, the term of quantum computing is more similar to analog computing instead of digital computing, as quantum computing allows continuum value of qubits (as it will be explained below). One of the goal of quantum computing is to use quantum properties which have no equivalents in classical computing , therefore quantum computing has some advantages over classical computing.

There are few underlying terminologies and properties in quantum computing :

1. Qubits and Some of Its Properties

Qubit, or a quantum bit, can be seen as a counterpart of classical's computing bit. However, qubit differs bit by the fact that it allows for continuum value; all state which are spanned by its unit vectors are also qubits. Qubit can represents all quantum mechanical system modeled by a two dimensional complex vector space .

The convention of the orthonormal is that |0> and |1> correspond to the states |↑> and |→>, respectively. (The notation follows Dirac's notation). With this convention, then |0> and |1> can be directly compared to classical computer bits' 0 and 1. However, as stated above, qubit is a continuous value. Therefore, a qubit can be represented as superposition of $\alpha|0> + \beta|1>$ and $|\alpha| + |\beta| = 1$. [6]

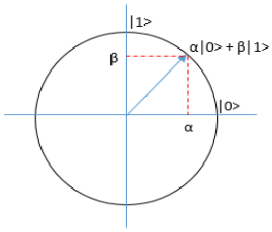

Fig. 1. Graphical representation of qubit. Notice that a qubit can be defined as a linear combination of |0> and |1>, and both α and β are complex numbers. Source : https://www.ijser.org/paper/A-Survey-The-Next-Generation-Of-High-Quantum-Performance.html, accessed 22:35 UTC+7 December 8 2018.

However, there are some peculiar properties about measurement in qubit , derived from various experiments and axioms in quantum mechanics : Let there be a device to measure a quantum state with preferred base of {|u>,|v>}. Measurement of a quantum state will change the state into one of the preferred base. If the quantum state is defined as $|x> = \alpha|u> + \beta|v>$, then the probability of the state measured as |u> is $|\alpha|^2$, and the probability of being measured as |v> is $|\beta|^2$. Measurements of the state after the first measurement will always result in the changed state with probability 1. Consequently, it is impossible to determine the original state before measurement. Other consequence is that although it is possible to store infinitely many states in a single qubit, but there can only be one bit worth of data extracted from qubit (since the measurement device has two orthonormal basis).

Another peculiar property of qubit which has no equivalent in classical bit is the principle of quantum parallelism. This property is one of the property that gives quantum computation advantage over classical computation,.

A single qubit Hadamard transformation H is applied to |0> to generate superposition state $(1/\sqrt{2})(|0>+|1>)$. If applied to all n qubit individually, all in state |0>, then a superposition of all 2^n standard basis vector will be generated. The superposition will be as the following :

$$\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x> \tag{2}$$

2^n can be written as a certain N.

A linear transformation U_f such that $U_f = |x,y> \rightarrow |x, y \oplus f(x)>$ acts on a superposition of input values $\Sigma a_x|x>$ as the following :

$$U_f = \sum_x |x,0> \rightarrow \sum_x |x, f(x)> \tag{3}$$

The effect of applying U_f to the superposition values obtained by Eq.2 is defined by Eq.4.

$$\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|\, x,0> \rightarrow \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|\, x, f(x)> \quad (4)$$

Eq. 4 means that one application of U_f contains all the 2^n values of f(x) entangled by their input value x. The ability of working simultaneously with 2^n values is named as quantum parallelism. However, the principle would be pretty meaningless without further transformation. Measuring in standard basis will return a random state |x,f(x)>, and the measured state will be projected into the final state (recall about measurement of qubits).

C . Challenges in Implementation

This sub-chapter is necessary to give a brief explanation of requirements in implementing a machine that employs quantum computation.

DiVincenzo has proposed five physical requirements to implement quantum computation[7] :

1. A scalable system physical system with well characterized qubits.
2. The ability of initialize the state of qubits to a simple fiducial (assumed as a fixed basis of comparison) basis, such as |00..>
3. Long relevant coherence duration, much longer than gate operation time. However, it is practically impossible to truly isolate a quantum computer from its environment. Environment is described as a subsystem which is not under control ; no measurements or gates can be applied.

Decoherence occurs when the information of the computational subsystem is lost to the environment. Decoherence acts as the main principal for the emergence of classical behavior. However, as stated above, environment is a state which is outside the computational control, therefore the problems becomes serious. To solve the problem, there are several quantum states error models and correction.

4. An "universal" set of quantum gates.
5. A qubit-specific measurement capability. An ideal measurement will be independent to state of nearby qubits and without changing the state of the rest of the quantum computer.

III. SHOR'S ALGORITHM

A. Motivation for Shor's Algorithm

The first motivation comes to a fact well-known to mathematician in 1970s' , that one can easily solve the problem of integer factorization if another hard problem, the order (or period) finding problem, can be solved. The problem will be briefly discussed later.

Some of the earlier ideas of a quantum machine came from Feynman (1982) , Benioff (1982) and Deutsch (1985). Benioff demonstrated that a quantum machine can model Turing machine, means that the quantum machine at least is as powerful as Turing machine. Deustch expanded the notion by demonstrating the quantum equivalent of Turing machine, and showed that quantum machine can simulate some problems beyond the scope of Turing machine (for example is generating a truly random number, instead of a pseudo random number). Studies circling around quantum computing grows, with the study of quantum circuit and gates. A paper by Bernstein and Vazirani in 1993 demonstrates that some problems which are to be solved in super-polynomial time classically can be solved in polynomial time by quantum algorithm. Simon showed an example of such algorithm in 1993, which became the inspiration for Shor to develop other quantum algorithm. As in 1994, in his paper "Polynomial-Time Algorithms For Prime Factorization and Discrete Algorithms on a Quantum Computer" , Shor presented two quantum algorithms which perform in a much less complexity of time when compared to its classical counterpart. [8]

This paper will only briefly explains the prime factorization algorithm , as the prime factorization problem is the key of breaking RSA encryption.

B. Outline of Shor's Algorithm

Shor algorithm reduces the problem into few basic steps. The algorithm is as the following :

1. Choose a random number m. By Euclidean algorithm, find gcd(m,n). If it is equal to 1 , then continue to next step, otherwise then the factor of n can be directly determined.
2. Let there a function $f(a) = x^a \bmod n$ where x is co-prime to m. From number theory, it can be concluded that f(a) is periodic , which means there exists r such as f(a) = f(a+r). Because $m^0 \equiv 1 \bmod n$, therefore $m^r \equiv 1 \bmod n$.
3. If r is an even integer , then proceed to step 4. Otherwise proceed to step 1.
4. Because r is even and m is co-prime to n, then the equation $m^r \equiv 1 \bmod n$ can be written as $(m^{1/2}+1)(m^{1/2}-1) = kn$. If $m^{1/2}+1 \equiv 0 \bmod n$, return to step 1.Otherwise, go to step 5.Determine d = gcd $(m^{1/2}+1, n)$, and d is the non-trivial solution.

Step 1, 3,4,5 are steps which can be solved by classical computer. Step 2 reduces the factorization problem into period-finding problem, which can be solved far more efficiently using quantum computation.

.B. Quantum Computation in Period-Finding Problem

As mentioned above, Shor's Algorithm reduces the factorization problem into period finding problem.

The period finding problem can be solved by quantum computer efficiently by employing parallelism and Quantum Fourier Transform (a quantum variant of Fourier Transform, analogue to Discrete Fourier Transform). The algorithm is as the following [9] :

1. Determine a number q, such that q is a power of 2 and $n^2 \le q \le 2n^2$.
2. Create two quantum registers , (register 1 and 2 respectively), so the quantum state can be defined as |reg1, reg2>. The first register should be put in the superposition of states representing number a mod q. The state of the quantum machine will be :

$$\frac{1}{q^{1/2}}\sum_{a=0}^{q-1}|\, a>|\, 0>$$

(5)

3. Initialize the second register with the superposition of all states $x^a \bmod n$ in Eq.(5) .The current state will be as the following :

$$\frac{1}{q^{1/2}}\sum_{a=0}^{q-1}|\, a\ > |\, x^a \bmod n >$$

(6)

4. Perform Quantum Fourier Transform A_q to the first register in Eq. (6). The state will be :

$$\frac{1}{q}\sum_{c=0}^{q-1}\sum_{a=0}^{q-1}\exp(2\pi iac/q)\ \ |\, c >|\, x^a \bmod n >$$

(7)

Then , do measurement to the machine in state represented by Eq.(7). The probability of a certain machine state of $|c\rangle|x^k \bmod n\rangle$ will be :

$$\left|\frac{1}{q}\sum_{a:x^a\equiv x^k}\exp(2\pi iac/q)\right|^2$$

(8)

That is, the sum when $x^a \equiv x^k$ for $0 \leq a < q$. Also, it can assumed that $k < r$ (as defined, r is the period). In other word , $a \equiv k \pmod r$, and therefore $a = br + k$.

The probability of a state $|c\rangle|x^k \bmod n\rangle$ will be at least $1/3r^2$ if exists $d \in \mathbb{Z}$ such as :

$$\frac{-r}{2} \leq rc - dq \leq \frac{r}{2}$$

(9)

Dividing by rq and rearranging the equation in Eq.(5), it is obtained that :

$$\left|\frac{c}{q} - \frac{d}{r}\right| \leq \frac{1}{2q}$$

(10)

The fraction d/r can be obtained with continued fraction expansion, with the algorithm being executable by classical computer. This part of the algorithm is considered to be the post-processing part.

C. Example of Shor's Algorithm

To visualize the algorithm in a more concrete sense and to demonstrate some steps, consider a problem of factoring 21[10] :

1. Take m = 6. Because gcd(6,21) = 3, then it can be directly determined that 21 = 3.7 .
2. Take other m, for example 11. Because gcd(11,21) = 1, continue to next step.
3. Next step is to find the appropriate q. Since n = 21, then $n^2 = 441$ and $2n^2 = 882$, therefore q = 512.
4. Initialize the first register with the superposition of all states a mod q :

$$\frac{1}{\sqrt{512}}\sum_{a=0}^{511}|\, a >|\, 0 >$$

(11)

5. Initialize the second register to be the superposition of all states $m^a \bmod n$:

$$\frac{1}{\sqrt{512}}\sum_{a=0}^{511}|\, a >|\, 11^a \bmod 21 >$$

(12)

To visualize it easier , Eq(12). can be written as :

$$\frac{1}{\sqrt{512}}(|\, 0 >|\, 1 > + |\, 1 >|\, 11 > + |\, 2 >|\, 16 > + |\, 3 >|\, 8 > \ldots)$$

(13)

The periodicity can be viewed from the expansion :

a	0	1	2	3	4	5	6	7
$11^a \bmod 21$	1	11	16	8	4	2	1	11

Table 1.Expansion of the first few terms. By the definition of period defined previously, thus it can be easily concluded that the period r is 6. However, r cannot be determined easily for larger numbers. Source : https://qudev.phys.ethz.ch/content/QSIT15/Shors%20Algorithm.pdf

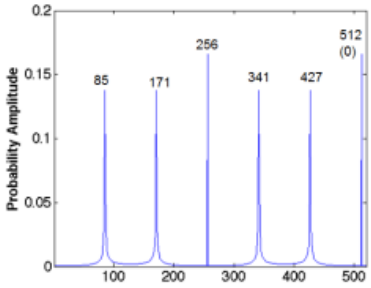

Figure 2 . The probabilistic distribution of c. It can be observed that applying the transformation enormously increases the probability of some values of c to be observed. Also, the "peaks" are located in a periodic manner, namely c= 512d/6 , d ∈ ℤ. Source : https://qudev.phys.ethz.ch/content/QSIT15/Shors%20Algorithm.pdf

6. Apply Quantum Fourier Transform to the first register :

$$\frac{1}{512}\sum_{c=0}^{511}\sum_{a=0}^{511}\exp(2\pi iac/512)\ \ |\, c >|\, 11^a \bmod 21 >$$

(14)

Probability of a certain state $|c\rangle|ma \bmod n\rangle$ is :

(15)

$$\left|\frac{1}{512}\sum_{b}\exp(2\pi i(6b+2)c/q)\right|^2$$

Assuming that c = 427, then:

$$\left|\frac{427}{512} - \frac{d}{r}\right| \leq \frac{1}{1024} \tag{16}$$

By continued fraction expansion :

$$\frac{c}{q} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\cdots}}} \tag{17}$$

A value r=6 can be obtained, which agrees with the observation in table 1.

D. Complexity and Efficiency of Shor's Algorithm

Shor proposed that the algorithm would only have exponential-time complexity. The algorithm would only take $O((\log n)^2(\log(\log n))(\log(\log(\log n)))$ steps in quantum computer, with additional polynomial time required for the continued fraction expansion. This comes with several facts about the algorithm itself , especially due to Quantum Fourier Transform.

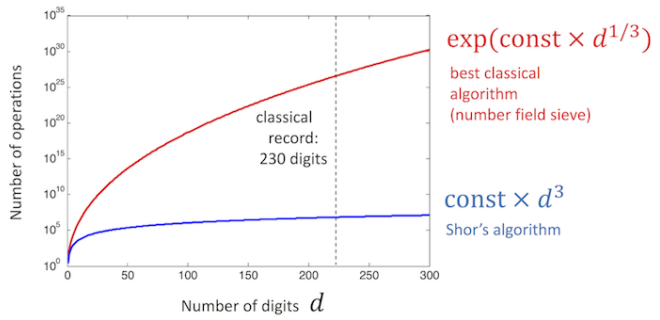

Figure 3. Comparison of the best classical algorithm (general number field sieve) with Shor's algorithm with the respect of the number of operations needed to factorize certain number with number of digits d. Source : https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/004-Quantum_Algorithms/110-Shor's_algorithm.html

IV. CHALLENGES FOR IMPLEMENTATION OF SHOR'S ALGORITHM AND FUTURE OF RSA ENCRYPTION

A. Challenges in Implementing Shor's Algorithm

Shor's algorithm provide a an example where quantum computation , theoretically, operates more efficiently than classical computing Other quantum algorithms do exhibit similar feature, such as Grover's algorithm, a search function with O(√N) evaluations. As described previously, a quantum system can also model universal Turing machine. The prospect of using such quantum machine is surely interesting. The current most advanced chips has transistors of 14 nm in diameter. A prediction based on Moore's Law would state that transistors would shrink to a size of less than 5 nm in diameters by 2020. A scale as small as that cannot be governed by classical physics solely; rather quantum physics would govern systems with such scale.[11]

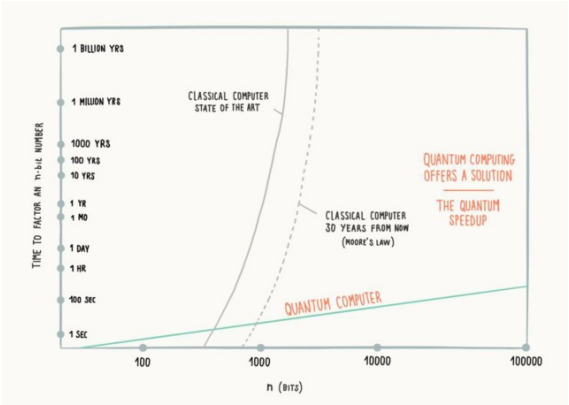

Figure 4. A prediction regarding the power of quantum factorization compared to classical factorization. The advances in chip by the Moore's law clearly do not reduces the time complexity needed by classical computer to perform factorization. Source : https://ibmcai.com/2016/03/15/quantum-computing-time-for-venture-capitalists-to-put-chips-on-the-table/

However, one should be wondering about why quantum computer has not been widely used. There are few challenges to the development of quantum computer. The first challenge to the implementation is the physical constraints and requirements to build a quantum computer. As discussed previously, there are few requirements to build such machine. One of the constraint is the fact that are quantum computers are extremely sensitive to entanglement with the environment. Larger number of qubits increase the vulnerability of a quantum system to decoherence , with the fact that larger size of an object would increase its classicality (a degree in which a system can be explained by classical mechanics).

Other than that, there has been some flaws pointed out about the Shor's algorithm. The first possible skepticism is mentioned in the paper by Z. Cao, Z. Cao and L. Liu (2014) [12]. The first possible flaw noticed is in the description by P. Shor about the transformation of |a>|0> into |a>|x^a mod n> (quantum modular exponentiation). In the original paper, Shor only described the conventional transformation (in which (a,1) -> (a,x^a mod n)), but do not really describe the transformation of the state itself. Also, the authors (Z. Cao, Z.Cao and L. Liu) noticed that it is also impossible to compose a superposition from pure states, therefore the paper boldly claim that the algorithm is flawed.

Other possible flaws are also pointed out by Z. Cao and Z.Cao (2014) [13]. The paper then suggested that the Shor's claim of polynomial-time algorithm should be reevaluated. (An interesting note about this paper : the paper is published in less than 6 months after the paper mentioned before, and contains the answer provided by MIT professor Scott Aaronson for the quantum modular exponentiation problem . However, the answer seems to be not convincing and "too vague").

B. Improvement of RSA Post-Quantum

As previously discussed, the biggest number factorized by Shor's algorithm is very small compared to the numbers used by standard RSA encryption used today. However, under the assumption that quantum computer can be developed at least in a par with modern classical computer, then the question of how

much quantum computation will be a threat to RSA encryption arose.

This certain problem is discussed in a paper published in 2017 by Bernstein, Heninger, Lou and Valenta [14]. The paper argued that the power of Shor's algorithm can be seen as exaggerated (especially with the conventional claim that the development of quantum computing means that RSA encryption will be totally useless). Furthermore, the paper also introduced a 1 terabyte RSA public key, essentially enough to push all quantum attacks to be over 2^{100} qubits. It is also demonstrated that, in terms of costs, speeding RSA key generation will much further widen the gap of cost of attack to user's cost. However, a key as large as that is still in theoretical scope, as the time needed the to generate such larger key is quiet impractical with current technology

One interesting aspect is that the paper also introduces another quantum factorization algorithm called as GEECM (Grover's method and Lenstra's elliptic-curve factorization with Edwards curve) , which is used in the paper as the main constraints for generating secure post-quantum RSA key instead of Shor's Algorithm. (Which means that the proposed GEECM can be even faster than Shor's algorithm, although there had been no (online) sources found regarding the time complexity of GEECM itself, as the paper is relatively new).

V. Conclusion

Although a more efficient algorithm to find RSA decryption key has been proposed and demonstrated, it can be concluded that RSA encryption will still be secure in the next few years, if not decades. However, it is possible to say that the power of quantum computing can be explored further, and other secure post-quantum encryption methods should start to be studied and implemented.

VII. Acknowledgment

First before all, I would give my gratitude firstly to Allah SWT, the Most Gracious and the Most Merciful, who has given me the life with all its beautiful things.

I would also give my gratitude to my parents, the ones who support me mentally, spiritually and financially to study in ITB.

I would also give my fond gratitude to all the lecturers of IF 2120 (Discrete Mathematics) : Dr. Ir. Rinaldi Munir, MT. , Dra. Harlili M.Sc. and Dr. Judhi Santoso M.Sc. for the guidance and teachings during the course. I sincerely enjoyed the course, as the course give deep insights to mathematics and computing.

References

[1] "A Brief History of Cryptology", published August 14, 2013, available in https://access.redhat.com/blogs/766093/posts/1976023, accessed at December 7, 2018.

[2] E. Milanov, "The RSA Algorithm', published June 3, 2009. [pdf] available : https://sites.math.washington.edu/~morrow/336_09/papers/Yevgeny.pdf , accessed December 6, 2018.

[3] Z. Lisa, "New largest number factored on a quantum device is 56,153", [online] available : https://phys.org/news/2014-11-largest-factored-quantum-device.html , published November 28, 2014. accessed December 6, 2018.

[4] K. Conrad, "Modular Arithmetic", [online], available : http://www.math.uconn.edu/~kconrad/blurbs/ugradnumthy/modarith.pdf , accessed December 10, 2018.

[5] Kleinjung T. et al., "Factorization of a 768-Bit RSA Modulus." in Rabin T. (eds) Advances in Cryptology – CRYPTO 2010. CRYPTO 2010. Lecture Notes in Computer Science, vol 6223. Springer, Berlin, Heidelberg, pp. 1-2. [online] , available at : https://eprint.iacr.org/2010/006.pdf, accessed at December 9, 2018

[6] E.Rieffel, W. Polak, *Quantum Computing : A Gentle Introduction.* Cambridge, MA : MIT Press, 2011, ch.2

[7] D.P. DiVincenzo , "The Physical Implementation of Quantum Computation ", in *Fortschritte der Physik*, 2000. available online at : https://onlinelibrary.wiley.com/doi/pdf/10.1002/1521-3978%28200009%2948%3A9/11%3C771%3A%3AAID-PROP771%3E3.0.CO%3B2-E , accessed December 9, 2018

[8] J. Cirasella,"Historical Bibliography of Quantum Computing" in *CUNY Academic Works*, New York, NY, pp.3-7. 2008. [online] ,available at : https://pdfs.semanticscholar.org/d7f5/03eddd3f39db53acd8495f52f357ebb707cd.pdf , accessed December 9, 2018

[9] P.W.Shor, "Polynomial-Time Algorithms For Prime Factorization and Discrete Algorithms on a Quantum Computer", in *SIAM Journal for Computing*, Vol. 26, No. 5, pp. 1484–1509, October 1997. [online] available at : http://mmrc.amss.cas.cn/tlb/201702/W020170224608150589788.pdf , accessed December 9, 2018

[10] E. Bäumer, J-G. Sobez, S. Tessarini, "Shor's Algorithm", [online] available : : https://qudev.phys.ethz.ch/content/QSIT15/Shors%20Algorithm.pdf, published May 15, 2015, accessed December 8, 2018.

[11] "Quantum computing: Time for venture capitalists to put chips on the table?",[online] , available : https://ibmcai.com/2016/03/15/quantum-computing-time-for-venture-capitalists-to-put-chips-on-the-table/ , published March 15, 2016 . accessed at December 10, 2018.

[12] Z. Cao, Z. Cao , L. Liu. "Remarks on Quantum Modular Exponentiation and Some Experimental Demonstrations of Shor's Algorithm", in *IACR Cryptology ePrint Archive* ,2014.[pdf], available : https://eprint.iacr.org/2014/828.pdf . accessed December 9 , 2018.

[13] Z. Cao, Z. Cao, "On Shor's Factoring Algorithm with More Registers and the Problem to Certify Quantum Computers" , in *IACR Cryptology ePrint Archive,* 2014. [pdf], available : https://eprint.iacr.org/2014/721.pdf , accessed December 9 ,2018.

[14] D.J. Bernstein , N. Heninger , P. Lou, L. Valenta, "Post-quantum RSA", in *Cryptology ePrint Archive*, Report 2017/351, [pdf], available : https://eprint.iacr.org/2017/351.pdf , accessed December 9, 2018.

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Desember 2018

Nur Alam Hasabie
13517096