ISSN 2278-3091
Volume 9, No.1.2, 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse2191.22020.pdf
https://doi.org/10.30534/ijatcse/2020/2191.22020

Analysis of Rabin-*p* And HIME(R) Encryption Scheme on IoT Platform

Abdul Muhaimin Abd Nashir [1], Syed Farid Syed Adnan[1], Habibah Hashim[1], Mohd Anuar Mat Isa[1], Zahari Mahad[2], Muhammad Asyraf Asbullah[2,3]
[1]InSTIL Research Laboratory, Faculty of Electrical Engineering, Universiti Teknologi MARA (UiTM) Malaysia.
[2]Institute for Mathematical Research, Universiti Putra Malaysia (UPM), Malaysia.
[3]Centre of Foundation Studies for Agricultural Science, Universiti Putra Malaysia (UPM), Malaysia

ABSTRACT

This paper focuses on the implementation and analysis of the performance of the Rabin-*p* encryption scheme on the microprocessor platform. Rabin-*p* is an asymmetric cryptosystem that comes with simpler cryptographic properties than the Rabin cryptosystem. Rabin-*p* encryption has yet been tested on any IoT platform. The study tries to analyze the Rabin-*p* behavior instead of the algorithm optimization itself on the IoT platform. The algorithm of Rabin-*p* tested by utilizing the C-programming and implemented on a microprocessor system namely Raspberry Pi 3 model B. The Raspberry Pi 3 can be a multi-sensor in an IoT environment. The Rabin-*p* runtime taken to encrypt and decrypt as well as the power consumption is then compared with the performance of another Rabin variant, the HIME(R) encryption scheme. The result shows Rabin-*p* encryption scheme runtime is faster at 50% and current withdraw less at 1.3% compared to HIME(R).

Key words: HIME(R), Encryption, IoT, Rabin-*p*, Rabin, Public Key

1. INTRODUCTION

Technology nowadays moves toward the development of wireless technology, especially in communication and networking. As the technology advance more to the Internet of Things (IoT), there will be a lot of weak spots that make the exchange of data and information between devices or parties that makes the information being transfer become vulnerable to cyberattack. Another example, when using a credit card to make an online transaction that contains vital information that needs to be protected. Solutions are needed to protect the privacy and the security of the information being exchanged. One of the solutions is to utilized cryptography for security purposes.

The Rabin encryption scheme [1] is one of an existing workable asymmetric cryptosystem that comes with nice cryptographic properties. For instance, Rabin has low-cost encryption and relatively fast to encrypt compared to the widely commercialized RSA cryptosystem [2]. However, The Rabin scheme have four to one output during decryption that leads to decryption failure. The research on Rabin are then continues and introduced to several Rabin variants such as Rabin-Takagi and HIME(R) [3]. Another variant namely Rabin-*p* was proposed by M.A. Asbullah and M.R.K. Ariffin in 2014 [3]. The Rabin-*p* encryption scheme that was developed has similar attributes like Rabin cryptosystem but more efficient and enhanced the abilities of the Rabin encryption scheme. The Rabin-*p* feature non padding needed during encryption and decryption compared to other Rabin variants. Another feature, instead of four possible plaintexts during decryption by Rabin cryptosystem, Rabin-*p* only produced one plaintext. Thus, unlike Rabin cryptosystem, Rabin-*p* produces plaintext output without decryption failure. The character *p* for Rabin-*p* represent the cryptosystem only used single prime for decryption reducing the computational efforts.

In 2018, M. Adli [4] explores the Rabin-*p* cryptosystem and compared it with RSA implementation on text file encryption. The authors concluded that Rabin-*p* has a much efficient system than RSA for both aspects, running time and size of the text. Another work proposed hybrid cryptosystem using the vigenere cipher algorithm and Rabin-*p* public key algorithm for the security of the BMP file was introduced in 2018 by M. Yogi [5]. This research tests and implements the encryption system on the BMP file which is a type of image file. The encryption performance is basically tested on the image file and the pixels of the file are observed. The author concluded that the image decryption time is longer than the time of the image encryption process [5].

Other than that, research by A. Sasmita [6] that conducts the implementation of Rabin-*p* key public algorithm and Affine cipher algorithm in a hybrid cryptosystem based on text security. The implementation is based on Java programming language and tested on Android Studio Software. The testing of the system encryption and decryption messages by changing the plain text to ASCII character code and the cipher key converted to a decimal number of ASCII character

code [6].All these papers recommend research on mobile apps [4][5][6]. The authors give a point on how technology is moving towards mobile applications and IoT.

There are some concerns regarding the security of the IoT system because most IoT devices lack no security to their system. The reason is that IoT devices simply have small processing capabilities. In order to implement the security system into IoT devices, a suitable encryption scheme that uses low energy needs to be found so that end-to-end security can be provided.

Another study was done by F. A. Alaba et al. [7] aims to serve as a useful manual of existing security threats and vulnerabilities of the IoT heterogeneous environment and proposes possible solutions for improving the IoT security architecture [7]. The authors discuss the needs of security in IoT platform and the vulnerabilities along with threats IoT devices are facing. The research paper also giving possible solutions to the problem with IoT platforms.

Another concern in the IoT platform is having the risk of the system being attacked. In a paper done by I. Yaqoob et al. [8], they stated that with the increasing miniaturization of smartphones, computers, and sensors in the Internet of Things (IoT) paradigm, strengthening the security and preventing ransomware attacks have become key concerns. This shows that a security or encryption scheme needs to be implemented in the IoT platform, so security no longer becomes a concern to users.

Thus, Rabin-*p* encryption might be a candidate for IoT encryption as it features minimal computation on the encryption and decryption process. However, the cryptosystem has not been tested on any embedded platform yet especially on the IoT platform. This makes the effectiveness of the Rabin-*p* encryption scheme on embedded platforms have yet been tested. In this paper, the Rabin-*p* encryption scheme is explored and implemented on a microprocessor or also called a System on Chip (SoC).

L. Caldas-Calle et al. [9] presents the evaluation of VPN Quality of Service (QoS) parameters over a Wireless Network (WLAN), using different Raspberry Pi models. The contribution of the results and their analysis determine the correlation between the parameters and the Raspberry Pi models. Thus, this experiment considers the Raspberry Pi as a testbed for the cryptosystems. Despite a few disadvantages, the Raspberry Pi remains an inexpensive platform with its very successful usage in a diverse range of research applications in IoT [10]. Apart from that, there are researches looking into reducing energy usage in the IoT platform such as Raspberry Pi and Arduino [11][12].

2. METHODOLOGY

2.1 Rabin-*p* Algorithm

The algorithm for Rabin-*p* encryption scheme is different than the algorithm of Rabin encryption scheme. As mentioned in introduction, Rabin-*p* encryption scheme is an improvement based on Rabin encryption scheme algorithm. The improvement made supposedly increases the efficiency and performance of the encryption scheme which is having faster encryption and decryption time.

The algorithm 1, algorithm 2 and algorithm 3 shows the algorithm of Rabin-*p* cryptosystem. There are three stages of algorithms in total that are needed to be tested. For algorithm 1, the process for generating the private key and public key for the encryption process.

This research analyzes and compares the performance of the Rabin-*p* algorithm particularly algorithms 2 and. Both stages can be measured to determine the performance of the Rabin-*p* algorithm whether it is better or worse than HIME(R) algorithm in selected IoT platform.

Algorithm 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

The encryption algorithm (i.e. Algorithm 2) takes the plaintext $m < 2^{2k-1}$ and compute $c = m^2 \pmod N$. The output produced is the ciphertext c.

Algorithm 2 Rabin-*p* Encryption Algorithm

Input: The plaintext m and 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

To decrypt a ciphertext, the Rabin-*p* decryption algorithm with the private key p does the following in Algorithm 3.

Algorithm 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 $m_p \equiv w^{(p+1)/4} \pmod{p}$

3: Compute $i = (c - m_p^2)/p$

4: Compute $j \equiv i/(2m_p) \pmod{p}$

5: Compute $m_1 = m_p + jp$

6: If $m_1 < 2^{2k-1}$, then return $m = m_1$

7: Else, return $m = p^2 - m_1$

2.2 Testing Platform

The microprocessor used in this paper was Raspberry Pi 3 model B. Raspberry Pi is an inexpensive, small-sized computer that has plenty of uses depend on user application.

Figure 1 shows the platform utilized in this paper. The Raspberry Pi 3 is the third generation of the Raspberry Pi microprocessor. It functions as a small-sized computer and can be connected with computer peripherals. The microprocessor used in this paper was Raspberry Pi 3 model B. Figure 2 shows the testbed set up with Raspberry Pi 3.

Figure 1: Raspberry Pi 3 and Connected Peripherals

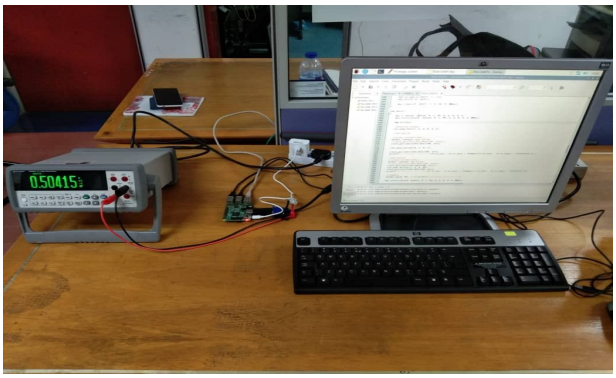

Figure 2: Testbed Setup

3. RESULT AND DISCUSSION

In this section, the results from the experiment are presented and analyzed to compare the efficiency of the cryptosystems. From there, it can be determined which system is more suitable to be implemented in IoT devices. Supposedly, the encryption scheme with lower energy usage and faster execution runtime is the better system to use.

3.1 Execution Runtime

The tables show the execution time for Rabin-*p* and HIME(R) encryption schemes that are implemented on the testbed setup. The runtime taken to encrypt and decrypt for each encryption scheme are recorded and listed as in Table 1 and Table 2.

The results show the three lengths of prime (n) which is used for each encryption scheme. The three lengths of prime are 341, 682 and 1365.

Table 1: Rabin-*p* Runtime

Length of Prime (n)	Encryption runtime (s)	Decryption runtime (s)
341	0.000019	0.0015904
682	0.000048	0.0107822
1365	0.0001474	0.0751684

Table 2: HIME(R) Runtime

Length of Prime (n)	Encryption runtime (s)	Decryption runtime (s)
341	0.000032	0.0033226
682	0.0001042	0.0226412
1365	0.0003138	0.1499782

For the Rabin-*p* encryption scheme in Table 1, each length of prime shows faster runtime when compared to HIME(R) encryption processes in Table 2 for both encryption and decryption process.

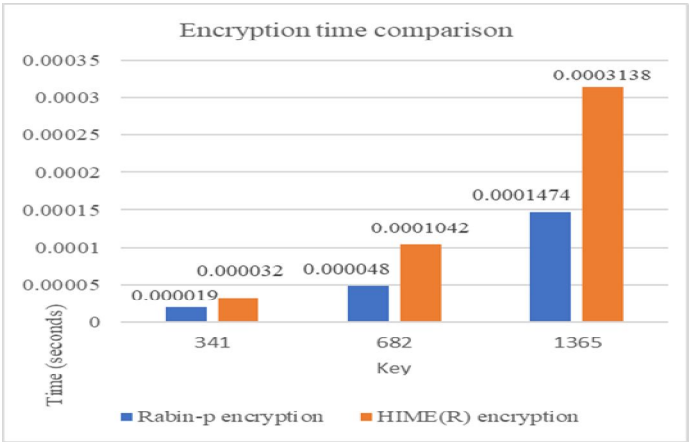

Figure 3: Rabin-*p* and HIME® Encryption Runtime

From Figure 3, it can be clearly seen that the Rabin-*p* algorithm takes less time to encrypt the plaintext compares to HIME(R) which means that it executes the program faster than HIME(R) algorithm. At highest prime number (n) 1365-bit, the Rabin-*p* encryption runtime is 53% less than HIME(R) at 0.3138ms.

In Figure 4, the comparison of decryption time between Rabin-*p* and HIME(R) encryption scheme is depicted. While both encryption scheme takes a longer time for the decryption process compared to the encryption process, Rabin-*p* algorithm still takes less time than HIME(R) algorithm. At the highest prime number (n) 1365-bit, the Rabin-*p* decryption at 0.0752 seconds is 49.88% less than HIME(R) at 0.15 seconds.

This result shows that when both encryption schemes were to be applied using a microprocessor, the Rabin-*p* algorithm is significantly more efficient when compared HIME(R) algorithm at 53% less runtime taken for encryption and 49.88% less runtime taken for decryption.

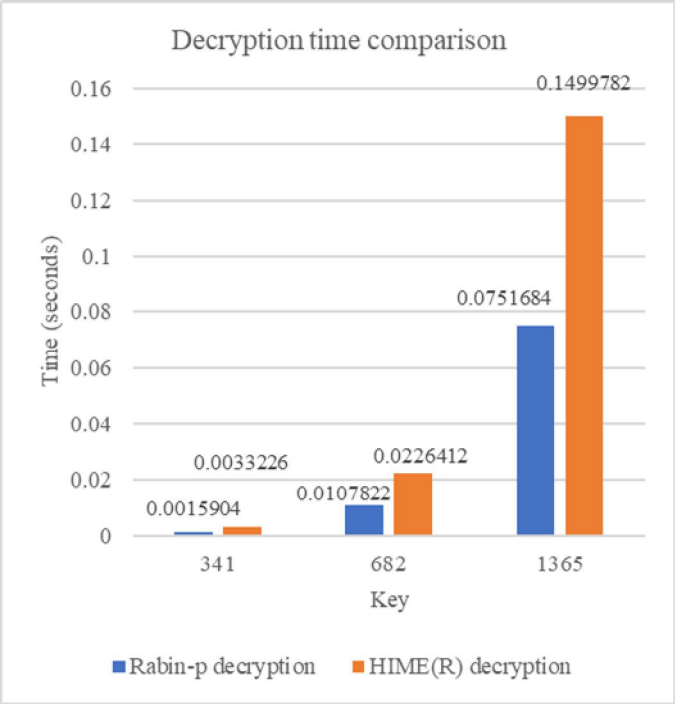

Figure 4: Rabin-*p* and HIME® Decryption Runtime

3.2 Current Consumption

In order to measure and determine the efficiency of the Rabin-*p* encryption system, the current draw from each length of prime is measured. Ideally, the closer the current output while encrypting the plaintext to the idle current is better. This is because it shows that the algorithm does not utilize massive power consumption. Initially, the current measured while the system is idle is 0.5066 Amps.

Table 3: Average current output

Length of Prime (n)	Idle current(A)	Rabin-*p* current(A)	HIME(R) current(A)
341	0.5066	0.64249	0.64375
682		0.65092	0.65971
1365		0.65449	0.66353

Table 3 listed the average current output measured while the encryption process is executed. The idle current is taken as baseline for comparison of power consumption.

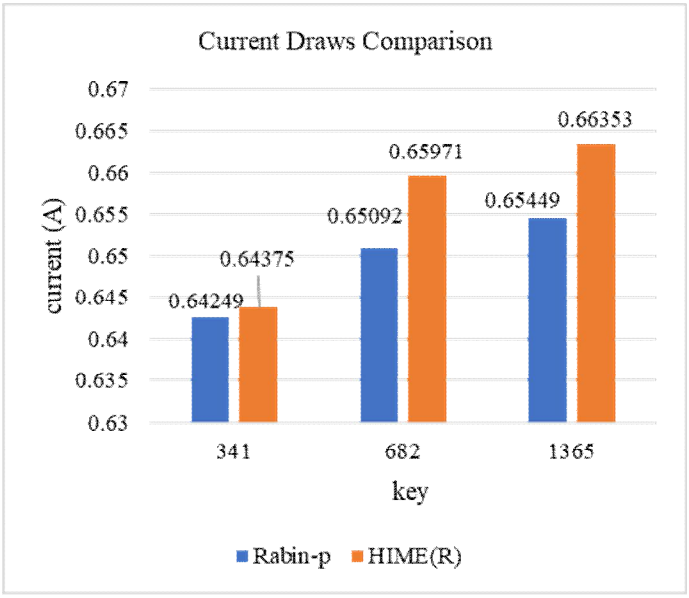

Figure 5: Current output

Based on Figure 5, the current output for HIME(R) algorithm is higher than the current output of the Rabin-*p* algorithm. Rabin-*p* achieves lower current draw while encrypting the plaintext due to the shorter time to find only one possible plaintext instead of four possible answer if using the Rabin scheme. Thus, resulting in lower energy while performing the encryption process. Compared to HIME(R) algorithm, the Rabin-*p* algorithm has a lower average current approximately at 1.3% lower for all keys that are executed.

4. CONCLUSION

As a conclusion, based on the data obtained from section III, the Rabin-*p* encryption system is more efficient when compared to HIME(R) encryption scheme. For the runtime of the Rabin-*p* encryption, it is considerably faster than HIME(R) for all three lengths of prime used. The same conclusion can be made for the decryption of data. The Rabin-*p* algorithm takes less time to decrypt the data that was encrypted.

For the current output, it can be concluded that Rabin-*p* using less power compare to HIME(R). Since the Rabin-*p* encryption scheme has fast execution time and has a low

current draw, therefore Rabin-*p* encryption scheme might be suitable for IoT devices is made. In the future, the Rabin-*p* could be compared with other Rabin variants.

ACKNOWLEDGMENT

The authors would like to acknowledge the Ministry of Education (MOE) Malaysia for providing the grant 600-IRMI/FRGS 5/3 (350/2019) and Universiti Teknologi MARA (UiTM) for supporting this research work.

REFERENCES

1. M. O. Rabin, **Digitalized Signatures and Public-Key Functions as Intractable as Factorization**, *MIT Technical Report*, MIT/LCS/TR-212, 1979.
2. R. L. Rivest, A. Shamir, and L. Adleman. **A Method for Obtaining Digital Signatures and Public-Key Cryptosystems**, *Communications Of The ACM*, 1978. https://doi.org/10.21236/ADA606588
3. M. A. Asbullah, M.R. K. Ariffin **Design of Rabin-like Cryptosystem without Decryption Failure,** *Malaysian Journal of Mathematical Sciences*, 10(S), 1-18, 2016.
4. M. Adli, **Perbandingan Efisiensi Algoritma Rabin-P dan Algoritma RSA pada Pengamanan File Txt Berbasis Desktop,** Deg. Thesis, Univ. Sumatera Utara, Indonesia, 2018.
5. M. Yogi, **Implementasi Kriptografi Hybrid Cryptosystem Menggunakan Algoritma Vigenere Cipher dan Algoritma Kunci Publik Rabin-P untuk Pengamanan File BMP**, Deg. Thesis, Univ. Sumatera Utara, Indonesia, 2018.
6. A. Sasmita, **Implementasi Algoritma Kunci Publik Rabin-p dan Algoritma Affine Cipher Secara Hybrid Cryptosystem pada Pengamanan Teks,** Deg. Thesis, Univ. Sumatera Utara, Indonesia, 2018.
7. Fadele Ayotunde Alaba, Mazliza Othman, Ibrahim Abaker Targio Hashem, Faiz Alotaibi, **Internet of Things security: A survey**, *Journal of Network and Computer Applications*, Vol. 88, pp. 10-28,2017. https://doi.org/10.1016/j.jnca.2017.04.002
8. I.Yaqoob, E. Ahmed, M. H. U. Rehman, A. I. A. Ahmed, M. A. Al-garadi, M. Imran, M. Guizani, **The Rise of Ransomware and Emerging Security Challenges in the Internet of Thing,** *Computer Networks: The International Journal of Computer and Telecommunications Networking*, Vol 129, pp. 444-458, December 2017.
9. L. Caldas-Calle; J. Jara; M. Huerta; P. Gallegos, **QoS evaluation of VPN in a Raspberry Pi devices over wireless network**, *2017 International Caribbean Conference on Devices, Circuits and Systems (ICCDCS), 2017.* https://doi.org/10.1109/ICCDCS.2017.7959718
10. M. Maksimović, V. Vujović, N. Davidović, V. Milošević, and B. Perišić, **Raspberry Pi as Internet of Things hardware: Performances and Constraints**, *Proc. in 1st International Conference on Electrical, Electronic and Computing Engineering IcETRAN 2014, Vrnjačka Banja, Serbia, June 2 – 5, 2014*, 2014.
11. Z. A. Khairul Nadilah and S. Hasan, **Survey on optimization of the energy consumption and load balancing in wireless sensor network**, *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 1.4 S1, pp. 229–235, 2019. https://doi.org/10.30534/ijatcse/2019/3581.42019
12. A. Majeed, R. Bhana, and S. Parvez, **Controlling energy consumption by internet of things (Iot) applications,** *Int. J. Adv. Trends Comput. Sci. Eng*., vol. 8, no. 1, pp. 8–11, 2019. https://doi.org/10.30534/ijatcse/2019/0281.12019