DATA ENCRYPTION DEVICE USING RADIOACTIVE DECAY AND A HYBRID QUANTUM ENCRYPTION ALGORITM

Anthony Kunkel, Karthik Paidi, Dennis Guster, Renat Sultanov, and Erich Rice
Department of Information Systems
Saint Cloud State University
Saint Cloud, MN 56301
kuan0902@stcloudstate.edu

Abstract

Quantum computers are the future in computing as data encoded in modern computers is limited by the space to store it. Quantum computers encode data in atoms that don't follow classical physics but rather quantum physics. Quantum physics provide an advantage in computing, as it allows data to be processed exponentially faster. However, increases in speed of such magnitude pose a risk to modern day encryption standards. Thus, to protect the transfer of data one must look towards developing innovative ways of encryption that shields it from the speed of quantum computers. This paper discusses a method to secure information using radioactive decay in conjunction with an encryption algorithm. The main purpose of this paper is to develop and implement an encryption device that can be interfaced with a computer system. The device then utilizes the randomness of radioactive decay as a seed used in the encryption algorithm.

Keywords: Quantum Computers, RSA, Encryption, Radioactive Decay

1 Introduction

Encryption is a relatively unknown important aspect of our daily lives. In a world where information is readily available to any person with an Internet connection, this potential can cause an internal fear that personal information is at risk of being illicitly obtained. Fortunately, the information that one would like to keep secure is encrypted so that the information is unreadable to a potentially malicious entity. Most data is currently encrypted with an algorithm designed by Rivest, Shamir, and Adleman (RSA). The algorithm has been extensively tested and continues to securely protect information. However, RSA may not be an acceptable encryption method for the foreseeable future. Progress is being made in the application of quantum physics, which shows particular promise in changing the way we currently use cryptography.

The concept of a computer with data encoded on an atomic scale is known as a quantum computer and may pose a security concern for modern encryption. Quantum computers have been theoretically shown to have the ability to break RSA encryption much faster than the best classical computer algorithms [1-3]. Presently, quantum computers are still in their infancy but it is foolish to ignore the future in which they may become fully functional. It is the purpose of this paper that we prepare for a future where RSA encryption is no longer reliable. We propose a design for a hybrid encryption algorithmic device using a quantum random number source in conjunction with a classical encryption algorithm.

The creation of an encryption device is important in establishing a pathway from RSA encryption towards developing algorithms that are resistant to quantum computer attacks. To optimize the proposed device's design for experimental applications it must meet a set of requirements. First, the generation of random numbers must be analyzed and shown that they are sufficiently random. Second, the encryption scheme used must prove to be complex in design to protect against guessing and breaking the encryption, and one should be able to interface the device with a computer system as a plug-and-play device. And third, it is our hope that the encryption method has greater security against quantum computer attacks than RSA encryption alone.

2 Literature Review

Quantum cryptography was an idea devised by Stephen Wiesner in the 1970's. Wiesner wrote an article "Conjugate Coding" which eventually was published in 1983 [4]. Wiesner argued that if you isolated a quantum system from the environment it would not be reproducible. Extending the idea to real-world applications, Wiesner believed that if money were encoded by these quantum systems it would be impossible to make counterfeit copies. Eventually Charles Bennet brought Wiesner's idea to Gilles Brassard and the two developed the first quantum cryptography protocol known as BB84 [5].

BB84 appended the ideas dictated by quantum mechanics with public-key distribution simply known as quantum key distribution (QKD) [6]. The protocol sends a quantum state $|\psi\rangle$ through a secure quantum channel. $|\psi\rangle$ is a two-state quantum particle called a quantum bit (qubit) [7]. The qubit acts like a classical bit except for the fact that it is in a superposition of states and holds the values of "0" and "1" simultaneously. Superposition is just one of three important rules in quantum physics that BB84 exploits. Another important feature used is that any observation of a quantum particle transforms the particles state [8]. Lastly, the no-cloning theorem of quantum mechanics forbids a quantum state to be reproduced [9]. The last two rules illustrate the usefulness of quantum cryptography. If an eavesdropper tried to acquire information from the quantum states, the states would transform signaling to the sender and receiver that their channel is insecure [10]. Additionally, it would be impossible for the eavesdropper to record the information of the state and then recreate the exact quantum system to pass to the receiver. This reiterates the idea that the sender and receiver are aware of any changes to their original quantum system and must establish a secure quantum channel.

While the BB84 protocol in theory is very powerful, quantum physics makes it difficult to develop and also expensive [11]. Isolating quantum systems from the environment is one of the biggest challenges quantum cryptography faces. The tendency for quantum particles to interact with its environment requires the state to be placed in a vacuum and decreasing temperatures to only a few degrees Kelvin.

Motivation to push the ideas of QKD is in the development of the first efficiently working quantum computer. Richard Feynman suggested that due to the properties of quantum mechanics could be utilized in computing [12]. Feynman argued that such a computing device would be much faster than classical computers. To better understand how this works we follow through an example using the property of superposition.

Thinking about a qubit that is represented by an electron, we can describe the qubit using the following equation.

$$|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

where $|0\rangle$ and $|1\rangle$ are the electron spin states. $|0\rangle$ represents the state where the electron is spin up and $|1\rangle$ represents the spin down state. The factor of $\frac{1}{\sqrt{2}}$ is the square root probability of observing the respective states. A quantum computer uses unitary quantum operators on the qubit state in computations, these operators are known as quantum gates. The quantum gates are similar to a classical computer's logic gates and transform $|\psi\rangle$ to a particular configuration. In quantum computing one or more qubits are sent through quantum gates until the designed algorithm is completed. The final state of qubit is observed to give the desired $|0\rangle$ or $|1\rangle$. Since by definition the qubit has two simultaneous values, each quantum gate computes two operations. Comparing the computations of a single bit to a qubit the qubit is two times faster.

Leaving the example and extending to several qubits one can show that for n-qubits, a quantum computer processes 2^n times more information than its classical counterpart. The incredible increase in speed is the reason quantum computers are such a tantalizing idea. The computation power ultimately will lead to solving problems that are difficult on classical computers more easily.

A difficult problem to solve on classical computers is integer factorization. RSA encryption is based on the principle that computers take a very long time to solve integer factorization problems. RSA uses the multiplication of two large prime numbers to encrypt keys for public-key distribution. As the number of prime number digits increases the longer it takes the computer to factor the product. According to Kirsch [13] “factoring time grows exponentially with input length in bits”.

The issue with the RSA algorithm is that the security of the encryption is only reliable if the speed of computers stays relatively slow. Peter Shor attacked the integer factorization problem but he used a quantum algorithm known presently as Shor’s Algorithm. He developed a way to use the computational speed of qubits to solve integer factorization in polynomial time, as opposed to classical algorithms, which takes exponential time [14]. Shor’s Algorithm poses a direct threat to RSA encryption if implemented on a quantum computer with a sufficient number of qubits. Fortunately for RSA encryption, quantum computers are not stable or large enough to break 2048 binary digit semi-prime used currently. However, progress is being made in the designs of quantum computers and has the potential to solve this problem in the near future.

To prepare for a potential future where RSA encryption is no longer reliable, we propose a hybrid approach between classical and quantum cryptography. The proposal is a design for an encryption device that uses quantum principles and can be implemented using classical computers. The quantum property used in the device is the non-deterministic time between consecutive decay events [15]. The time between two decay events will be treated as a random number source to be used in a proposed encryption algorithm [16]. The original algorithm was intended to use photon spins as the quantum principle, but it is less expensive to use a radioactive source and a Geiger-Müller detector. Showing that the radioactive source acts as a random number generator will further motivate its use in the encryption process.

3 Methodology

In the interest of making this discussion as structured as possible the methods used in the research will be broken into three subsections. The first section being the actual design of the device, which includes the various components used. The second section is the description of the encryption algorithm developed for the use of a radioactive source. The final section describes the randomness testing suites used in the analysis of the generated numbers.

3.1 Device Design

The encryption device includes several components including: A radioactive decay detector, radioactive source, a data acquisition apparatus, and a computer interface communicator. Such a device will be incorporated with the hybrid encryption algorithm described in the next subsection.

The radioactive source acts as a quantum random number generator by utilizing the time difference of radioactive events. The events are detected and represented by electrical pulses that are sent to the data acquisition apparatus. After the data is recorded and extracted by the computer interface as random integers, the algorithm can then utilize these random numbers as a seed for its operations.

3.2 Hybrid Encryption Algorithm

The algorithm begins by taking in a user specified set of bits to be encrypted. For real-world applications, the set of bits are considered to be the secret key that will be sent to the receiver. The complexity of the algorithm is customizable by the user to create a unique and more complex encryption, but for simplicity we will keep it at a proof-of-principle level discussion.

If we separate our randomly generated numbers into two sections around an average time N we can create four different cases that begin the encryption process.

- Case 1: The time generated is less than or equal to N and the first bit is 0
- Case 2: The time generated is less than or equal to N and the first bit is 1
- Case 3: The time generated is greater than N and the first bit is 0
- Case 4: The time generated is greater than N and the first bit is 1

According to which case the algorithm finds true, a bit conversion phase is applied to the string of bits.

- For case 1 the rest of the bits in the string are converted by a pseudo-randomly generated number between 100- 549 if the bit is a 0, else a number between 550-999 is generated for a 1.
- For case 2 the rest of the bits in the string are converted by a pseudo-randomly generated number between 550-999 if the bit is a 0, else a number between 100-549 is generated for a 1.
- For case 3 the rest of the bits in the string are converted by a pseudo-randomly generated number between 550-999 if the bit is a 0, else a number between 100-549 is generated for a 1.
- For case 4 the rest of the bits in the string are converted by a pseudo-randomly generated number between 100-549 if the bit is a 0, else a number between 550-999 is generated for a 1.

The pseudo-randomly generated numbers are arbitrary in its range but for this discussion the ranges are chosen because each number generated contains three digits. The ranges also contain the same amount of numbers so no bias is introduced in the converted bits.

To decrypt the bits converted using the algorithm the same cases presented before stay the same but process is reversed. Given the knowledge of the generated time and the first bit in the string one can convert back to the original bit string using the following instructions.

- For Case 1 each converted bit is checked and if it is in the range of 100-549 then it is converted back to 0, else it is converted back to 1.
- For Case 2 each converted bit is checked and if it is in the range of 550-999 then it is converted back to 0, else it is converted back to 1.
- For Case 3 each converted bit is checked and if it is in the range of 550-999 then it is converted back to 0, else it is converted back to 1.
- For Case 4 each converted bit is checked and if it is in the range of 100-549 then it is converted back to 0, else it is converted back to 1.

Advantages of this method lie in the idea that for each random decay event could restart the process making the ability to guess the method of encryption much more difficult. The quantum process that seeds the algorithm is independent of the encryption process, eliminating the predictability of which case is used. Although, as the complexity of the algorithm increases it also increases the amount of information the receiver must know to decrypt the converted bits.

3.3 Randomness testing

To accurately analyze the random numbers generated by the radioactive source it is important that we test them. The way most standard true-random and pseudo-random number generators are tested is by test suites developed over the years. Some of these tests include: NIST, DieHarder, ENT, and TESTU01 [17-19]. Each test suites have their advantages and disadvantages but each hold numerous statistical evaluations of random numbers. For the scope of this paper we will use the NIST and DieHarder test suites. It is our intention to use some of the tests inside the suites to evaluate the randomness of the numbers produced for the encryption algorithm.

4 Results

Original Bits	Converted Bits
0	136
1	781
0	120
1	971
1	789
0	271
1	860
0	102
0	460
1	654
1	771
1	983
1	865
1	822
1	766
1	620
1	551
1	878
1	995
1	751
1	934
1	851
1	975
0	475

Table 1: Table of encrypted bits with a random time of 3591 μs.

As an example, we include a demonstration of the encryption process on a set of hypothetical secret key bits. For simplicity, we investigate a small piece of the original bit string and show the conversion of bits in Table 1.

In the example, the random time generated was 3591 μs with N being 4024. Checking this against the cases described above with the first bit being 0, the algorithm converts the bits using case 1. Therefore, each 0 and 1 is converted between 100-549 and 550-999, respectively. Decrypting the converted bits is also trivial to see as we work backwards through the case description we could then obtain our original bits.

As we leave a simple example and extend the number of bits to the order of 10^4 it is interesting to see what the converted bits look. It is unreasonable to look at these bits and their conversions in a table, we plot the set of numbers in Figure 1.

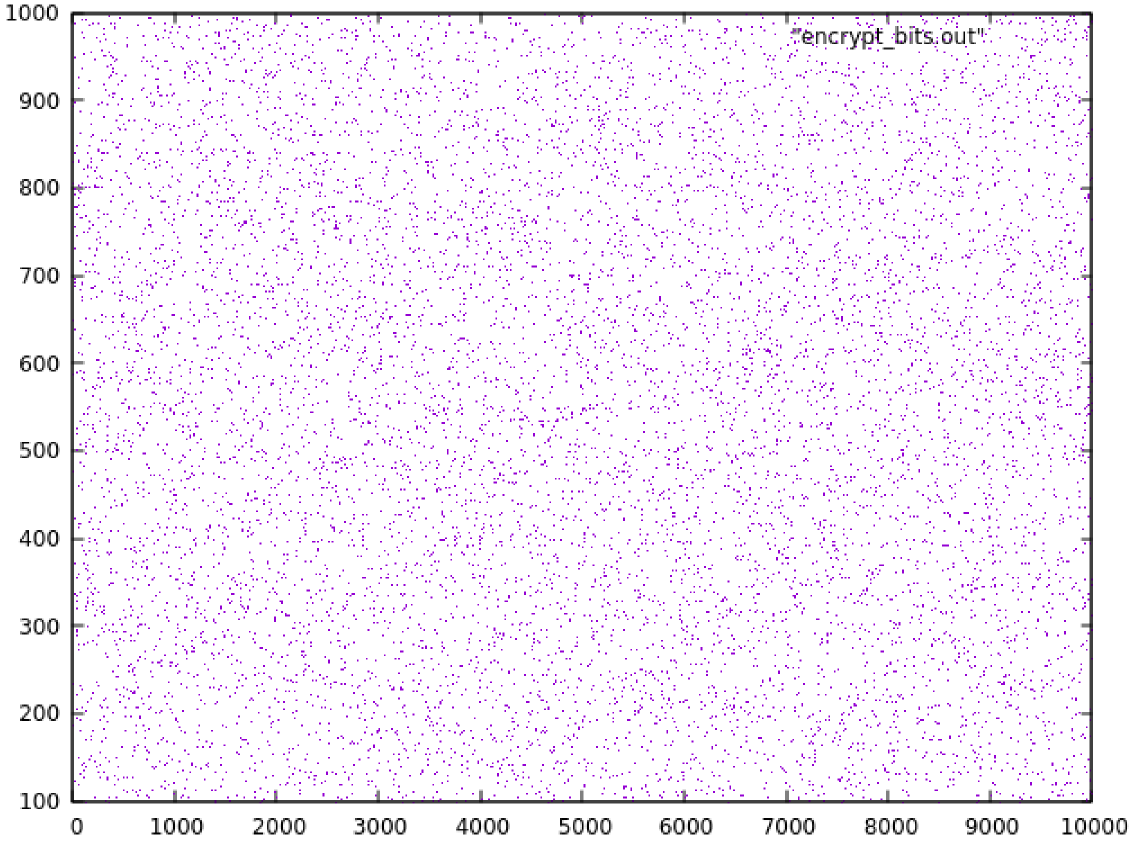

Figure 1: Plot of 10000 converted bits.

The advantages to plotting the data in this fashion is to look for patterns in the sequence that emerge. It is not a conclusive test of randomness but it does support evidence that the converted bits are well represented.

The generated time between consecutive decays were put through the DieHarder and NIST test suites to aide in the analysis of the numbers. The random numbers tested weakly passed the Overlapping 5-Permutation Test with a p-value of 0.00040904. They also passed the Runs and permutation test with p-values between 0.08567849 and 0.92911441. As for the NIST test suite, the numbers passed both the Nonoverlapping template and the Linear Complexity tests with p-values of 0.350485. The low p-values could be attributed to the sample of random numbers used in the test. Due to the robustness of the tests they require a significant amount of data points to run all its tests. The samples used in the tests were on the order of 2.5×10^6 numbers. Some tests require upwards to ten times more points to run.

5 Discussion

The ideas presented in this paper might appear simplistic in design but they provide several useful properties. The algorithm holds exponential complexity in the ability to change the number of cases used in the encryption process. Defining multiple relations to N would allow for an expansion in the amount of cases used in the algorithm. The number of quantum random numbers required for the encryption remains relatively small when using the algorithm. A low amount is advantageous since the use of a weaker source of radiation is acceptable. A highly radioactive source would be required for producing larger amounts of numbers. However, this would pose as a health concern if not properly shielded from the user. The most important advantage to the encryption process is it is not based on integer factorization. This property provides protection against quantum computer attacks where RSA fails.

The compactness of the device allows it to act as portable hardware that can be interfaced easily with any computer system. It would act as a black box hardware device and would require a 5V power source to operate. The low voltage property is unique given that many radiation detectors alone require a high voltage source. This allows the device to be more practical and work in the confines of current computer hardware. Secret keys could be produced and sent to the device to begin the encryption process. After the original bits are converted the information would be sent back to the computer for distribution.

Alongside the advantages there are also some disadvantages. As the encryption process becomes more complex the more information sent to the receiver is required to decrypt the original key. For example, the more random numbers used in the algorithm the more numbers a receiver must get for the decryption process. Each number sent is therefore a security concern as an eavesdropper might be able to obtain the numbers. Padding could be used within the sent message as to complicate an eavesdropper's ability to receive the sent information.

Further analysis of the random nature of the numbers used in the device is required to increase confidence in radioactive decay. A greater analysis will be conducted with a larger sample of random data points. The use of more statistical test suites will also provide higher confidence in its use for the algorithm. Implementation of this algorithm on a local system of information transfer would be ideal for security testing in the future.

Given the security concerns that quantum computers could inflict in the future of information encryption, it is of the utmost importance that new algorithms are pursued. It is our belief that the first stage in protecting ourselves is to hybridize our current systems as a precautionary step. Implementing an encryption algorithm using radioactive decay would provide a reliable and cost-effective device that would not be limited by integer factorization.

References

[1] Bimpikis, K., & Jaiswal, R. (2005). Modern factoring algorithms. *University of California, San Diego,*

[2] Lenstra, A. K., Lenstra Jr., H. W., Manasse, M. S., & Pollard, J. M. (1990). Number field sieve. 564-572.

[3] Sengupta, B., & Das, A. (2017). Use of SIMD-based data parallelism to speed up sieving in integer-factoring algorithms. *Applied Mathematics and Computation, 293*, 204-217. doi:http://dx.doi.org.libproxy.stcloudstate.edu/10.1016/j.amc.2016.08.019

[4] Wiesner, S. (1983). Conjugate coding. *ACM Sigact News, 15*(1), 78-88.

[5] Brassard, G. (2005). Brief history of quantum cryptography: A personal perspective. *Theory and Practice in Information-Theoretic Security, 2005. IEEE Information Theory Workshop on,* 19-23.

[6] Bennett, C., & Brassard, G. (1984). bb84. *Proc. IEEE International Conference on Computers, Systems, and Signal Processing, IEEE Press, Los Alamitos, Calif,* 175.

[7] Ballentine, L. E. (1970). The statistical interpretation of quantum mechanics. *Reviews of Modern Physics, 42*(4), 358-381. doi:10.1103/RevModPhys.42.358

[8] Nisticò, G., & Sestito, A. (2016). 'Evaluations' of observables versus measurements in quantum theory. *International Journal of Theoretical Physics*, *55*(3), 1798-1810. doi:10.1007/s10773-015-2819-4

[9] Wootters, W. K., & Zurek, W. H. (1982). A single quantum cannot be cloned. *Nature, 299*(5886), 802-803.

[10] Anghel, C. (2011). New eavesdropper detection method in quantum cryptograph. *Annals of Dunarea De Jos, Vol 34, Iss 1, Pp 1-8 (2011),* (1), 1.

[11] Barde N., Thaku, D., Bardapurkar, P., & Dalvi, S. (2012). Consequences and limitations of conventional computers and their solutions through quantum computers. *Leonardo Electronic Journal of Practices and Technologies, 10(19),* pp. 161-171.

[12] Feynman, R. P. (1986). Quantum mechanical computers. *Foundations of Physics, 16*(6), 507-531.

[13] Kirsch, Z., & Chow, M. (2015). *Quantum Computing: The Risk to Existing Encryption Methods,*

[14] Shor, P. W. (1999). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Review, 41*(2), 303-332.

[15] Rohe, M. (2003). RANDy-A true-random generator based on radioactive decay. *Saarland University,* , 1-36.

[16] Paidi, K., Kunkel, A., Guster, D., Sultanov, R., & Rice, E.a hybrid quantum encryption algorithm that utilizes photon rotation to insure secure transmission of data.

[17] L'Ecuyer, P., & Simard, R. (2007). TestU01: AC library for empirical testing of random number generators. *ACM Transactions on Mathematical Software (TOMS), 33*(4), 22.

[18] Marsaglia, G. (1996). DIEHARD: A battery of tests of randomness. *See Http://stat.Fsu.Edu/~ geo/diehard.Html,*

[19] Walker, J. (2008). Ent: A pseudorandom number sequence test program. *Software and Documentation Available at/www.Fourmilab.ch/random/S,*