VIBGYOR ePress

International Journal of Electronics and Device Physics

ISSN: 2631-5041

Data Encryption Device Using Radioactive Decay and a Hybrid Quantum Encryption Algorithm

Anthony Kunkel*, Karthik Paidi, Dennis Guster, Renat Sultanov and Erich Rice

Department of Information Systems, Saint Cloud State University, USA

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, the increase in computing speeds of such magnitude poses a potential risk to modern day encryption standards. Thus, to protect the transfer of data one must look towards developing innovative ways of encryption that shield it from the speed of quantum computers. This paper discusses a method to secure data by 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

Introduction

Encryption is a key element in the daily routine of most people's lives. Without it the use of social media, on-line banking and e-commerce could not function securely. While the internet offers numerous benefits one must accept that their personal information stored on the internet is at risk [1]. Fortunately, the information that one would like to keep secure is typically encrypted. To date, for the most part, the current classic encryption algorithms have been successful in keeping sensitive information from potential hackers [2]. However, in recent years classical algorithms have shown limitations [3]. To illustrate the point one could surmise that a significant amount of data is encrypted using the algorithm designed by Rivest, Shamir, and Adleman (RSA). The algorithm has been extensively tested and continues to securely protect information [4]. Due to the potential availability of almost limitless processing power in the future through quantum computers, many feel that RSA may not be an acceptable encryption method in the foreseeable future [5-8]. Progress is being made in the applications of quantum physics to encryption technology. This intersection has shown promise in changing the manner in which cryptography is currently used [9]. The idea of computers using qubit technology encoded on the atomic scale known as quantum computers may pose as a security concern for classical encryption. Quantum computers have been theoretically shown to have the ability to break RSA encryption much faster than the best classical computers and are becoming closer to being fully operational. Presently, quantum computers are still in their infancy but it is foolish to ignore the future in which they will have the functionality to break algorithms such as RSA. It is therefore, the purpose of this paper to suggest a solution that would be viable in a future in which RSA encryption is no longer reliable. There are numerous and robust communication solutions being proposed, but in many cases, they are exotic and expensive [10]. The solution being proposed herein is

***Corresponding author:** *Anthony Kunkel, Department of Information Systems, Saint Cloud State University, Saint Cloud, MN 56301, USA, E-mail: kuan0902@stcloudstate.edu*

Received: March 31, 2017: **Accepted:** July 12, 2017: **Published:** July 15, 2017

Citation: Kunkel A, Paidi K, Guster D, Sultanov R, Rice E (2017) Data Encryption Device Using Radioactive Decay and a Hybrid Quantum Encryption Algorithm. Int J Electron Device Phys 1:002

designed to be inexpensive and easily implemented. The design features a hybrid encryption algorithmic device using a quantum random number source in conjunction with an encryption algorithm.

The creation of such an encryption device is an important step in establishing a path away from RSA encryption and towards developing algorithms that are resistant to quantum computer attacks. In order to optimize the device's design for a production environment it must meet several requirements. First, the generation of random numbers must be analyzed and confirmed that the numbers generated are sufficiently random. Second, the encryption scheme used must prove to be complex enough in design to protect against guessing/breaking the encryption. Third, the devices should be easy to interface with a computer system's open architecture plug-and-play schema. And finally, the primary goal is to achieve an encryption methodology that provides adequate security against quantum computer based attacks offering an improvement in that realm over classical RSA encryption.

Literature Review

The origins of quantum computing begin with the work of Stephen Wiesner in the 1970's. Specifically, the defining article "Conjugate Coding" was published in 1983 [11]. "Conjugate Coding" has been simply defined as: noisy transmission of two or more "complementary messages" by using single photons in two or more complementary polarization directions/bases [12]. Extending the idea of "Conjugate Coding" to secure quantum based communication was a natural extension because Wiesner correctly pointed out that if you isolated a quantum system from the environment it would not be reproducible. One of the weaknesses of classical systems is that they are predictable in a sense in that the key is generating the dynamic portion of the algorithm used. In contrast, in a quantum world the entire system is dynamic. There were numerous potential applications to real-world applications. For example, Wiesner believed that if money was encoded using quantum systems it would be impossible to counterfeit. Of course, the application most pertinent to this paper would be secure quantum data transmission. The first such application occurred when Charles Bennet brought Wiesner's idea to Gilles Brassard and the two developed the first quantum cryptography protocol known as BB84 [13,14]. The use of conjugate coding is still important and numerous enhancements to the basic process have been devised [15].

From an operational perspective, BB84 appended the ideas dictated by quantum mechanics to a public-key distribution system and became known as Quantum Key Distribution (QKD) [16]. Specifically, the protocol sends a particle in the quantum state $|\psi\rangle$ through a secure quantum channel. Where, $|\psi\rangle$ is a two-state quantum particle deemed a quantum bit (qubit) [17]. The qubit acts like a classical bit except for the fact that it is in a superposition of states. This means that it can hold the values of "0" and "1" simultaneously. Superposition is just one of three important rules related to quantum physics that BB84 exploits. A second important feature that is used is that any observation of a quantum particle transforms the particles state [18]. From a data communications perspective, this is important because it can be determined if someone tried to read the transmitted message. Last, the no-cloning theorem of quantum mechanics forbids a quantum state to be reproduced [19]. Hence the symbolic logic in each transmitted object would be unique. The last two rules illustrate the usefulness of quantum cryptography. The possible advantages will follow in more detail. For example, if an eavesdropper tried to acquire information from the quantum states, the states through transformation would signal the sender and receiver that their channel is now insecure [20]. Additionally, by using simple observation or any other means it would be impossible for the eavesdropper to record the information of the state and then recreate the exact quantum system to try to spoof the receiver. This reiterates the idea that the sender and receiver monitor the quantum channel and would be aware of any changes to their original quantum system. If changes are detected it is just a matter of establishing a new secure quantum channel.

While the BB84 protocol in theory is very powerful, quantum physics makes its development difficult and expensive [21]. Isolating quantum systems from the environment is one of the biggest challenges quantum cryptography faces. There has been much written in the literature about "weak quantum values". The underlying architecture needed to represent qubits is related to this problem. For example, the tendency for quantum particles to interact with the environment often requires the state to be placed in a vacuum and cooled to very low temperature perhaps only a few degrees Kelvin [22].

The work involving the ideas of QKD facilitated the development of the first efficient working quantum computer. Specifically, Richard Feynman suggested that the efficient properties of quantum mechanics observed in quantum communication could be effectively utilized in computing [23]. Feynman argued that such a computing device would be much faster than classical computers in part due to the increased density of qubits when compared to regular bits. To provide the reader with a better understanding of this process the following simple example is provided. This example is designed to provide a quick review of the property of superposition.

In this case, a qubit that is represented by an electron and that qubit can be described by 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. Then $|0\rangle$ represents the state where the electron is in the spin up state 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 like the 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 the qubit is observed to give the desired $|0\rangle$ or $|1\rangle$. Since by definition the qubit can contain multiple values in the example above it has two simultaneous values. Therefore, each quantum gate must compute two operations. Comparing the computations of a single bit to the qubit above, it is clear that the qubit is twice as dense.

However, the number of qubits in a quantum computer can certainly be extended. In fact, when extending beyond a single qubit 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 why quantum computers are such a tantalizing idea. This enhanced computation power will ultimately lead to solving problems that are difficult and time consuming on classical computers.

A prime example that is central to this paper is solving integer factorization on classical computers. RSA encryption is based on the principle that classical 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 classical computer to factor the product. According to Kirsch [24], "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 classical computers stays relatively slow. The work of Peter Shor illustrated this potential vulnerability. His work attacked the integer factorization problem but he used a quantum algorithm known as Shor's Algorithm. This algorithm illustrated how the computational speed of qubits could be used to solve the integer factorization problem in polynomial time, as opposed to classical algorithms which takes exponential time [25]. It is clear that Shor's Algorithm poses a direct threat to RSA encryption if implemented on a quantum computer that contains a sufficient qubit capacity. For the time being quantum computers are not stable or large enough to break 2048 binary digit semi-prime used currently in RSA encryption. However, progress is being made in the designs of quantum computers and it appears that they will have the potential to solve this problem in the near future [26].

In order to prepare for a potential future where RSA encryption is no longer reliable, the authors propose a hybrid approach using elements of both classical and quantum cryptography. The proposed solution suggests 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 [27]. The time between two decay events will be treated as a random number source to be used in a previously proposed encryption algorithm but refined for this project [28]. The original algorithm was intended to use photon spins as the quantum principle, but it is less expensive to use a radioactive source in conjunction with a Geiger-Müller detector. Illustrating that the radioactive source acts as a true random number generator is critical to validating the algorithm. Further, when the generator is used in tandem with the algorithm proposed it is critical to validate that it used a method, unlike integer factorization, that cannot be easily broken by a quantum computer.

Methodology

To provide structure to the presentation the methodology section will be broken into three subsections. The first section will present the design of the device used to generate the random numbers which will include several related electronic components. The second section provides a description of the encryption algorithm devised to be used in conjunction with the radioactive source device. The final section describes the randomness testing suites used in the analysis to validate the generated numbers.

Device design

The quantum random number generator device used to support the hybrid encryption algorithm utilizes several components. These components include: a Geiger-Müller detector, an Arduino, an Arduino interface shield, a Raspberry Pi 3, an encryption algorithm, and a Cesium-137 source. To customize these devices for the intended purpose three programs were written, one on the Arduino and two on the Raspberry Pi. A block diagram of the device is displayed in Figure 1.

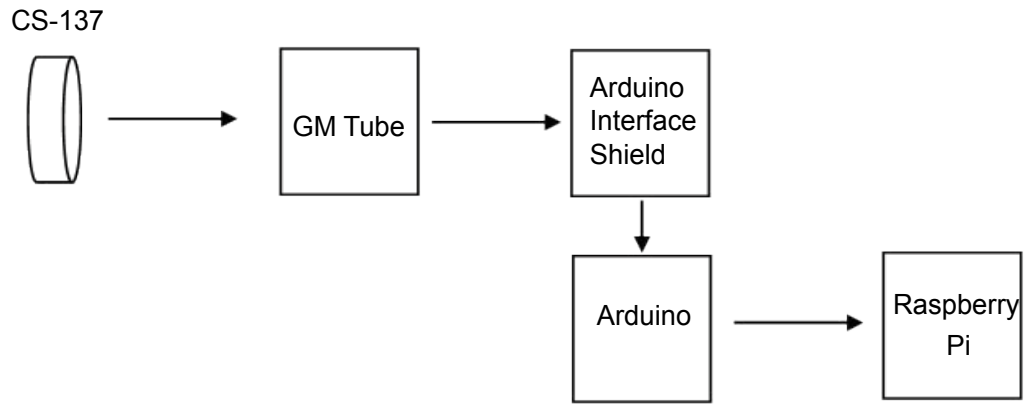

Figure 1: Block diagram of the encryption device.

The Geiger-Müller detector is used to convert the decay process of radioactive particles into a voltage signal which is sent to the Arduino using the interface shield. After the detection of a particle the detector goes through a discharge phase, which is known as dead time. During this period, no detections can be made. To be as efficient as possible, the selected detector should have a relatively short dead time. The detector used in the experimentation had a dead time of 50 microseconds (μs) which is adequate for proof of concept. Further, for the sake of simplicity the detector utilized requiredonly a 5-volt (V) power source. The radioactive source that was used in the detection process is a 1 microcurie Cesium-137 radioactive isotope. Specifically, Cesium-137 was chosen for its long half-life of 30.17 years.

The primary function of the Arduino is to act as a micro-controller so that data can be recorded. So that random numbers can be generated the Arduino will record the time between consecutive decay cycles following the logic of the program that is resident on this device. The program sends the data collected to the serial port, which is then extracted by using a python program stored on the Raspberry Pi. The time is recorded in microseconds, thereby providing a granularity that is more favorable than the value often used in classical computing, milliseconds.

The Raspberry Pi is designed as a mobile computer interface that can easily communicate with other computer systems. The data recorded from the Arduino and detector is then passed to the encryption algorithm on the Raspberry Pi, which is described in more detail in the next section. In a production system, the Raspberry Pi's OS will automate the detection, recording, and encryption process, but in this proof of concept example data collection and analysis this was skipped.

Since the device is based on a Raspberry Pi and the components are relatively small, it easily could be made portable. If conversion to a production system was desired, the device could be placed in a case that would be transportable and connected to a computer system using a USB cable. The radioactive source is considerably weak and would only require a small amount of shielding. However, this may not even be necessary depending on its expected distance from the user. It may also be possible to convert the device to a card that could fit into PCI Bus with little change in the design configuration.

Table 1: Table of four cases built from the possible permutations of the first number generated and the first bit in the string.

	Relation between the generated time and the average	The first bit
Case 1	τ	0
Case 2	τ	1
Case 3	τ	0
Case 4	τ	1

Hybrid encryption algorithm

The first step in the algorithm is to accept a set of user specified set bits to be encrypted. In a production system, the set of bits would be analogous to a secret key that would be sent to the receiver. The complexity of this algorithm can be customized by the user to create a unique and more robust encryption process, but for the sake of simplicity the discussion in this paper will be limited to the proof-of-principle level.

By separating the randomly generated numbers, denoted as τ, into two sections around an average time *N*, four different possibilities arise. The four possibilities are displayed in Table 1 as cases that define how the bits will be converted. The time generated by the radioactive decay element is separated into two sections as to increase the number of permutations the first bit can be associated with.

Based on 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 range of the pseudo-randomly generated numbers is arbitrary, but for this discussion and greater simplicity the ranges were chosen because each number generated contains three digits. The ranges also contained an equal amount of numbers so that no bias or overlapping is introduced in the conversion process.

To decrypt the bits that were converted using the algorithm the same cases presented before are utilized but the process is reversed. By knowing 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.

The advantages of this method lie in the idea that 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. However, as the complexity of the algorithm increases it also increases the volume of information the receiver must know to decrypt the converted bits. It is also important to note that simply increasing the number of sections the generated time is put into, further increases the number of cases. For example, if N was divided in half its relation to τ would increase to three sections: 0, $\frac{N}{2}$, and N. Given that the bit can either be a 0 or a 1 would increase the number of cases to eight.

Randomness testing

To accurately analyze the random numbers generated by the radioactive source it is important that they are subjected to a series of tests. The way most standard true-random and pseudo-random number generators are tested is by test suites that have been developed over the years. Some of the most popular measurement instruments include: NIST, Dieharder, ENT, and TESTU01 [29-31]. Each test suite has its own advantages and disadvantages, but each contains numerous statistical evaluations of random numbers. For the scope of this paper, which again is merely proof-of-concept, the NIST and Dieharder test suites have been used. The primary goal is to utilize the tests contained in the suites to evaluate the randomness of the numbers produced for the encryption algorithm and thereby evaluate the validity of the device itself.

Results

As an example, a demonstration of the encryption process is included for a set of hypothetical secret key bits. For simplicity, just a small piece of the original bit string is evaluated to illustrate the conversion of bits and the results shown in Table 1.

In this example, the random time generated was 6137 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 3. Therefore, each 0 and 1 is converted between 550-999 and 100-549, respectively. Decrypting the converted bits is also trivial. To accom-

Table 2: Table of encrypted bits with τ = 6137 μs.

Original bits	Converted bits
0	569
1	186
0	930
1	250
1	440
0	752
1	546
0	783
0	933
1	253
1	425
1	115
1	186
1	447
1	494
1	225
1	453
1	221
1	163

plish this, it is just a matter to work backwards through the case description to get the original bits. For example, in the second row of Table 2 the logic of Case 3 is applied.

When expanding the analysis beyond a simple example and extending the number of bits to the order of 10^4 it is interesting to see what the pattern of converted bits looks like. It is not prudent to view these bits and their conversion in Tabular form. Rather, a graphical depiction is more effective hence a plot of the set of numbers appears below in Figure 2. The x-axis represents the number of bits converted and the y-axis represents the encrypted value associated with each bit.

The advantage to plotting the data in this fashion is it makes it easier to visually ascertain if patterns in the plot sequence emerge. Another data representation technique that is helpful in the proper understanding of a large number of converted bits is using a histogram. Figure 3 displays the distribution of converted bits as it relates to the number of times each number occurs. The histogram shows that the distribution is nearly uni-

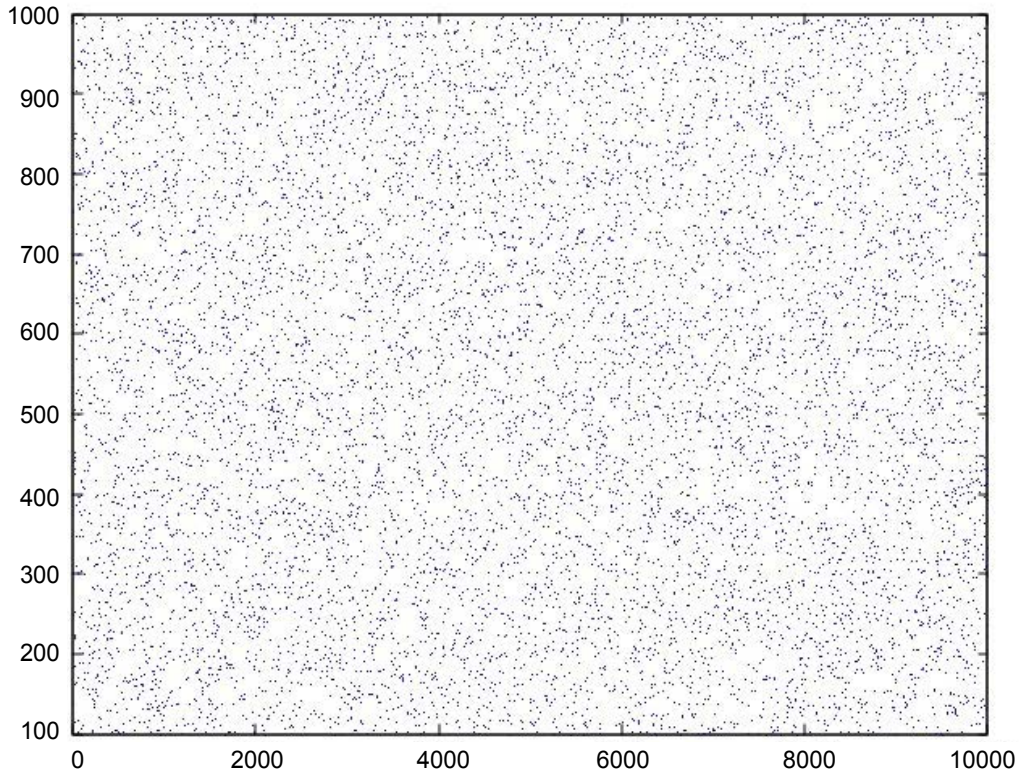

Figure 2: Plot of 10000 converted bits.

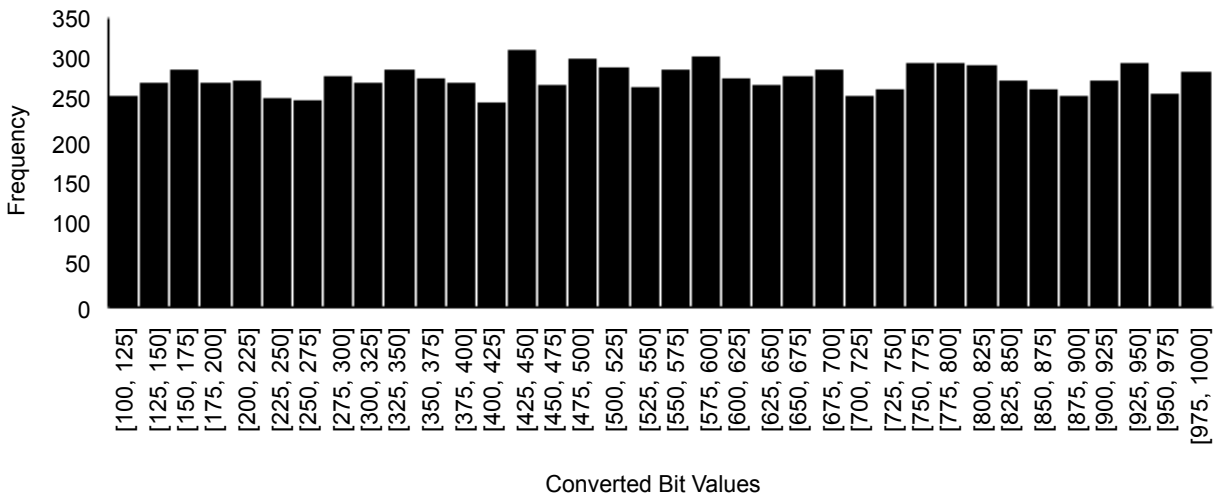

Figure 3: Histogram of the 10000 converted bits and their respective frequency counts.

form in each range. A uniform distribution motivates the proper working of the conversion process as no range is heavily weighted. The uniformity would also facilitate in hiding the original bit string as compared to a skewed distribution.

The times generated between consecutive decays within quantum random number generator were evaluated using the Dieharder and NIST test suites. 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 based on the evaluative feedback provided by the test suite with p-values between 0.08567849 and 0.92911441. As for the NIST test suite, the numbers passed both the Non over lapping template and the Linear Complexity tests according to that site's feedback with p-values of 0.350485. The marginal p-values could be attributed to the small sample size of random numbers used in the test. The robustness of the tests necessitates a significant number of data points to achieve full validity. The samples used in the tests were on the order of 2.5×10^6 numbers. Some of the more sophisticated tests require upwards to ten times more data points before an adequate sample size is reached.

Discussion

On the surface the ideas presented in this paper could appear simplistic in design but they incorporate several properties that are useful in the design of hybrid quantum security algorithms. Specifically, the algorithm holds exponential complexity in regard to the ability to change the number of cases used within the encryption process. Defining multiple relations to N would allow for an expansion in the number of cases used in the algorithm. The volume of quantum random numbers required for the encryption process would still remain relatively small even when increasing the number of cases. A small number of random numbers is advantageous because it allows the use of a weaker source of radiation. Conversely, a highly radioactive source would be required for producing long strings of random numbers. However, this design would pose a health concern if not properly shielded from the user. Further, if this hybrid algorithm were to be commercialized it would be much easier to implement as using a low intensity radiation source. The most important advantage of the encryption process is it is not based on integer factorization. This property provides protection against quantum computer attacks where even the latest RSA encryption would fail.

The compactness of the device allows it to be packaged as portable hardware that can be interfaced easily with any computer system either through a USB port or as a PCI card. To users it would appear as a black box hardware device and would require only a 5 V power source to operate. The low voltage property is unique given that many radiation detectors require a high voltage source. Particular detectors require 300-600 V to operate correctly. The low voltage allows the device to be more practical and work in the confines of current computer hardware architecture. The secret keys could be produced on the users host 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.

The decryption process is merely the inverse of the initial encryption algorithm. The information necessary for decryption includes each case instruction, the first original bit, the random time used, and the average time N. In theory, this information could be transmitted along with the key, but it would cause potential security holes in the process. For example, if someone were to obtain all of this information it would not be difficult to decrypt and obtain the original message. To combat some of these security holes a second device would be constructed that would contain the decryption algorithm and receive the encrypted message. Therefore, the only information required for decryption would be the random time and the first bit.

Besides these advantages there are also some disadvantages. If an increase in the robustness of the encryption algorithm is desired then the process becomes more complex. Specifically, this relationship can be described as follows: as complexity in the algorithm increases the volume of information that needs to be sent to the receiver to decrypt the original key increases as well. For example, the more random numbers used in the algorithm the more numbers a receiver must obtain to complete the decryption process. Each number sent is a security concern as an eavesdropper might be able to acquire the numbers. Of course, a mechanism such as padding could be used within the sent message to complicate an eavesdropper's ability to interpret the information obtained by snooping.

Further analysis of the random nature of the numbers generated by the device is required to increase confidence in the validity of the radioactive decay process used on the device. A subsequent analysis will be conducted with a larger sample of random data points. It is further planned to use more statistical test suites to help evaluate the confidence level in the application of the algorithm. The ideal test of the implementation of this algorithm would be to test it in a live client/server computing environment. This is the current state of development for the authors' system. Of course, testing will involve basic functionality, performance, and system overhead concerns as well as evaluating any vulnerabilities related to eavesdropping.

Conclusion

Given the security concerns that the advent of quantum computers pose to classical encryption, it is of the utmost importance that new algorithms are pursued. Also, given the complexity of quantum encryption it seems logical that the first stage in protecting sensitive data is to evaluate and deploy hybrid systems as a precautionary step. Therefore, implementing an encryption algorithm using a radioactive decay device would provide a reliable and cost-effective system that would not be limited by integer factorization.

No longer can we rely solely on the abilities of classical computer systems when it comes to data encryption. It is time that new encryption schemes are built to incorporate quantum systems as an aide in the future quantum computing world. The device proposed was built keeping this idea in mind. The goal in the future will be to make all information secure using quantum encryption schemes. However, as quantum encryption continues to make progress, hybrid systems must be developed to facilitate in the meantime.

References

1. Wright (2017) Mapping the Internet of Things. Commun ACM 60: 16-18.
2. DG Singh, D Garg (2005) Soft computing.
3. M Blumenthal (2007) Encryption: Strengths and Weaknesses of Public-key Cryptography. Csrs 2007 1.
4. R Oppliger (2014) Secure messaging on the internet. Artech House 57-58.
5. K Bimpikis, R Jaiswal (2005) Modern factoring algorithms. University of California, San Diego.
6. K Lenstra (1990) Number field sieve. 564-572.
7. G Mone (2013) Future-proof encryption. Commun ACM 56: 12-14.
8. Sengupta, A Das (2017) Use of SIMD-based data parallelism to speed up sieving in integer-factoring algorithms. Applied Mathematics and Computation 293: 204-217.
9. Edwards (2017) Secure Quantum Communications. Commun ACM 60: 15-17.
10. JY Haw, Jie Zhao, Josephine Dias, Syed M Assad, Mark Bradshaw, et al. (2016) Surpassing the no-cloning limit with a heralded hybrid linear amplifier for coherent states. Nat Commun 7: 13222.
11. S Wiesner (1983) Conjugate coding. ACM Sigact News 15: 78-88.
12. K Svozil (2006) Staging quantum cryptography with chocolate balls. American Journal of Physics 74: 800-803.
13. H Bennett, SJ Wiesner (1992) Communication via one-and two-particle operators on Einstein-Podolsky-Rosen states. Phys Rev Lett 69: 2881-2884.
14. G Brassard (2005) Brief history of quantum cryptography: A personal perspective. IEEE Information Theory Workshop 19-23.
15. M Hamada (2006) Conjugate codes and applications to cryptography. Tamagawa University Research Review 12: 19-25.
16. Bennett, G Brassard (1984) "bb84" in Proc. IEEE International Conference on Computers, Systems, and Signal Processing, IEEE Press, Los Alamitos, Calif, 175.
17. LE Ballentine (1970) The statistical interpretation of quantum mechanics. Reviews of Modern Physics 42: 358-381.
18. G Nisticò, A Sestito (2016) Evaluations of Observables Versus Measurements in Quantum Theory. International Journal of Theoretical Physics 55: 1798-1810.
19. WK Wootters, WH Zurek (1982) A single quantum cannot be cloned. Nature 299: 802-803.
20. Anghel (2011) New Eavesdropper Detection Method in Quantum Cryptograph. Annals of Dunarea De Jos 34: 1-8.
21. N Barde, Deepak Thakur, Pranav Bardapurkar, Sanjaykumar Dalvi (2011) Consequences and Limitations of Conventional Computers and their Solutions through Quantum Computers. Leonardo Electronic Journal of Practices and Technologies 10: 161-171.
22. J Dressel, Mehul Malik, Filippo M Miatto, Andrew N Jordan, Robert W Boyd (2014) Colloquium: Understanding quantum weak values: Basics and applications. Reviews of Modern Physics 86: 307.
23. RP Feynman (1986) Quantum mechanical computers. Foundations of Physics 16: 507-531.
24. Z Kirsch, M Chow (2015) Quantum Computing: The Risk to Existing Encryption Methods.
25. PW Shor (1999) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev 41: 303-332.
26. Nordrum (2016) Quantum Computer Comes Closer to Cracking RSA Encryption. IEEE Spectrum.
27. M Rohe (2003) RANDy-A true-random generator based on radioactive decay. Saarland University, 1-36.
28. K Paidi, Anthony Kunkel, Dennis Guster, Renat Sultanov, Erich Rice. A hybrid quantum encryption algorithm that utilizes photon rotation to insure secure transmission of data.
29. PL Ecuyer, R Simard (2007) TestU01: AC library for empirical testing of random number generators. ACM Transactions on Mathematical Software (TOMS) 33: 22.
30. G Marsaglia (1996) DIEHARD: a battery of tests of randomness.
31. J Walker (2008) Ent: A pseudorandom number sequence test program. Software and Documentation.

Citation: Kunkel A, Paidi K, Guster D, Sultanov R, Rice E (2017) Data Encryption Device Using Radioactive Decay and a Hybrid Quantum Encryption Algorithm. Int J Electron Device Phys 1:002