Quantum Safe Lightweight Cryptography with Quantum Permutation Pad

Randy Kuang
Quantropi Inc.
Ottawa, Canada
randy.kuang@quantropi.com

Dafu Lou
Quantropi Inc.
Ottawa, Canada
dafu.lou@quantropi.com

Alex He
Quantropi Inc.
Ottawa, Canada
alex.he@quantropi.com

Alexandre Conlon
Quantropi Inc.
Ottawa, Canada
alex.conlon@quantropi.com

***Abstract*— Quantum permutation pad or QPP was first proposed by Kuang and Bettenburg in 2020 [15]. QPP is a generic quantum algorithm consisting of multiple n-qubits quantum permutation gates. As a quantum algorithm, QPP can be implemented both in a quantum computing system as a quantum circuit operating on n-qubits' state for transformation and in a classical computing system represented by a pad of n-bit permutation matrices. QPP has two unique characteristics: huge Shannon information entropy and non-commutativity between permutation matrices or the generalized uncertainty principal. Permutation transformation is bijective mapping between input information space and output ciphertext space. That means, QPP has the property of Shannon perfect secrecy with reusability due to the uncertainty relationship. QPP is the generalization of One-Time-Pad or OTP over Hilbert space and OTP is the simplification of QPP over a Galois field. Based on those, this paper explores a variant of AES for a quantum safe lightweight cryptography by incorporating AES ShiftRows and MixColumns with QPP or called AES-QPP. AES-QPP unifies the SubBytes and AddRoundKey with the same QPP of 16 8-bit permutation matrices, essentially SubBytes to be a special 8-bit permutation matrix and AddRoundKey to be 16 8-bit permutation matrices selected from XOR operations. By randomly selecting 16 permutation matrices with a secret key material, AES-QPP could hold a total equivalent 26,944 bits of Shannon entropy. It not only improves the security against differential and linear attacks but also largely reduces the number of rounds to 5 rounds. AES-QPP could be a good candidate for quantum safe lightweight cryptography.**

Keywords*— *AES, quantum gates, permutation matrix, quantum algorithm, QPP, differential analysis, linear analysis, Shannon entropy, lightweight cryptography.

I. INTRODUCTION

The U.S. National Institute of Standards and Technology (NIST) announced that AES would adopt the Rijndael cipher as a specification for the encryption of electronic data in 2001 [1], where details of the algorithm design can be found in [2]. AES is a typical block cipher with block size of 128 bits or 16 bytes. The algorithm is standardized to only accept key lengths of 128, 192, and 256 bits, and consists of multiple rounds: 10, 12, and 14 rounds, determined by the key lengths, respectively. A typical AES round consists of four steps: SubBytes, ShiftRows, MixColumns, and AddRoundKey. SubBytes is a substitution transformation which takes a single byte as the input, and uses a static, 16x16 S-Box to substitute for an output. The S-Box is derived from mapping the input byte to its multiplicative inverse over Galois field $GF(2^8)$ = $GF(2)[x]/(x^8+x^4+x^3+x+1)$, which represents a very well selected non-linearity. ShiftRows permutes the positions of each byte inside the block based on a static permutation box, or P-Box. It can be simply considered as byte position permutation within a block. The output state is again used as the input state for the MixColumns operation. In the MixColumns step, the transformation can be represented by a left multiplication, of the AES state, by the Rijndael maximum distance separable (MDS) matrix used to increase diffusion capability

The first three steps are static and deterministic, without involving the encryption key. So, they can be considered as a pre-randomizing process for AddRoundKey, where entropy is injected into the input plaintext and the output state of MixColumns is carried out by a bitwise XOR operation with a round-key generated from the AES key scheduling process. Through multiple rounds, AES can produce good random ciphers with excellent confusion and diffusion properties.

The AES cipher is classified as a substitution-permutation network (SPN). There are two major types of attacks for SPNs: differential and linear analysis attacks [3]. The differential analysis attack arises from the non-linear transformations of the static S-Box which transforms input bytes into output bytes with some characteristic of XOR differences, especially impossible differences found at round 4, which leads to powerful impossible differential attacks [4,5]. The attackers investigate the XOR of pairs of input and output states and apply known characteristics to crack the AES key. The differential analysis attack can be further improved with sets or multisets of input and output XOR results to create a new attack called an integral attack [6]. On the other hand, AddRoundKey at the end of each round performs a linear transformation between rounds which leads

the linear analysis attack [3, 4, 5, 6], also leveraging from the AES key expansion algorithm which has strong correlations between round-keys.

Many techniques have been explored to make AES resistant against the differential and linear attacks. Tiessen et al in 2015 [6] replaced the static S-Box with a secret S-Box, selected randomly based on the supplied AES encryption key. They presented attacks based on integral cryptanalysis that can recover both the secret S-Box and the encryption key after six rounds of AES with a time complexity of 2^{90}, which is less than the general brute-force attack. Kazlauskas et al in 2015 [7] also investigated the key-dependent S-Box. Dara and Manochehri in 2014 [8] generated the S-Box with RC4 key scheduling algorithm. Das et al in 2012 [9] used different irreducible polynomials over GF(2^8) to create random S-Box.

In addition to classical attacks, classical cryptographic standards face potential quantum computing threats as rapid advancements in the development of quantum computers are being made. In September 2019, Google declared quantum supremacy [11] based on its 54-qubit quantum computer called "Sycamore" — a milestone achievement in quantum computing development. The well-known Shor's algorithm (1994) [10] provides an exponential speedup in breaking current public key exchange standards but is still largely theoretical as there is no powerful enough universal quantum computing device available. On the other hand, Wang et al (2020) [12] made a milestone achievement in prime factorization with D-Wave's annealing quantum computer. They successfully factorized all integers within 10,000 and demonstrated the best index (20-bit integer (1028171)) for deciphering RSA via the quantum computing software environment provided by D-Wave. Their method provides a completely different way than Shor's algorithm to factorize integers and opens a new path for cracking strategies for public key cryptographies based on computational difficulty. The D-wave quantum computer may be closer to cracking practical RSA codes than a general-purpose quantum computer using Shor's algorithm. The commercial availability of quantum computers, especially of the quantum annealing type, poses a major existential threat to today's public key exchanges, shaking the foundations of contemporary information security.

Moreover, L. Grover invented a new quantum search algorithm called Grover's algorithm (1996) [13], which solves the unstructured search problem of size n in $O(\sqrt{n})$ queries, while any classical algorithm needs $O(n)$ queries. This speedup requires the key length of standard AES to be raised from 128 bits up to 256 bits with true randomness to retain the security of AES. Bonnetain, Naya-Plasencia, and Schrottenloher in 2019 [14], developed their quantum security analysis of AES.

In their 2020 paper [15], Kuang and Bettenburg proposed a novel way to represent classical bits as quantum computational basis states to harden the security of classical cryptographic algorithms against the inevitable threats posed by quantum computing. The fundamental difference between classical computing and quantum computing lies in the computing algebra; it is shifted from Boolean algebra over Galois field GF(2^n), to linear algebra over Hilbert space $(\mathbb{C}^2)^{\otimes n}$, where n is the number of quantum bits. Quantum bits, or qubits, are naturally equipped with the characteristics of superposition and entanglement, which endow quantum computers with parallel computing capabilities. There remains a vast number of quantum logic gates to be created to unleash the super-computing power of qubit systems. However, it is well-known from quantum computing that there exists a special class of quantum logic gates, called quantum permutation gates, that exhibit a classical behavior. They are bijective mappings from the quantum computational basis to itself. The extremely large size of the quantum permutation gate space, $2^8!$ (factorial), for an 8-qubit quantum computational basis holds huge equivalent Shannon information entropy, desirable for information security. It turns out that this classical-behavior class of quantum logic gates forms a group representation of the symmetric group on 256 characters, S_{256}, and can be applied to both qubits and bits . Kuang and Bettenburg [15] successfully extend the Shannon perfect secrecy of the classical one-time-pad (OTP) over GF(2^n) [16], to their proposed quantum permutation pad (QPP) over a quantum computational basis. In contrast to the one-time-use nature of OTP, QPP retains Shannon perfect secrecy over multiple uses, thanks to the non-commutativity and non-involutory properties of the symmetric group. In this paper, we extend the work presented in [15] to AES and propose a lightweight quantum safe cryptography called AES-QPP.

In the remainder of this paper, Section 2 is devoted to the summary of quantum permutation pad, a proposed AES round replacement and a security analysis, is in Section 3. Section 4 will discuss the randomness of AES-QPP ciphers from AES-QPP with different numbers of rounds. The conclusion will be drawn at the end.

II. SUMMARY OF QUANTUM PERMUTATION PAD

A. *Quantum Permutation Gates*

A quantum logic gate or quantum gate is a basic quantum circuit operating on a small number of qubits. Quantum gates are simply classified into two categories: non-classical behavior and classical behavior gates. The former represents quantum superpositions and entanglements and the later is deterministic transformation from an input state of the system to an output state, or simply state permutation. For an n-qubit system with 2^n information states represented by Galois field GF(2^n), the entire state permutations form the symmetric group S_{2^n}, with total $2^n!$ unique permutations. A generic permutation gate can be physically implemented with an algorithm proposed by Shende et al in 2003 [17] using quantum NOT, CNOT and TOFFOLI gates.

An n-qubit permutation gate can be represented by a 2^nx2^n permutation matrix P[2^n, 2^n] over a quantum computational basis: {|0⟩, |1⟩, ..., |2^n-1⟩}, with only one element to be 1 on each row and each column and all others to be 0. Each permutation matrix represents a bijective mapping from input information space to output space. There exist $2^n!$ unique bijective mappings between input and output information space. Therefore, permutation transformations have the property of Shannon perfect secrecy.

Another unique feature of permutation gate is the non-commutativity, or two different permutations P and P' follow [P, P'] ≠ 0. This relationship can be considered as the generalized uncertainty principle which paves the way for reusability of a randomly selected quantum permutation pad.

B. Quantum Permutation pad for AES

AES is a block cipher with 128 bits or 16 bytes as a block. In stead to have a single static permutation matrix or the S-Box as used in the standard AES, we can randomly select 16 8-bit permutation matrices P_i, i=1, 2, …, 16, as shown in Figure 1

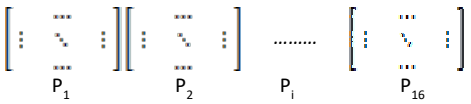

Figure 1. the illustration is a typical QPP with 16 permutation matrices of 256x256.

Each can be randomly selected by using the Fisher-Yates shuffling algorithm with 256 bytes of random numbers, total 32,768 bits. Algorithm 1 illustrates the pseudo code of QPP implemented with the Fisher-Yates shuffling.

Algorithm 1. Pseudo code of QPP mapping from the secret key

```
-- only illustrate a single permutation matrix selection
-- state array S[256] → a permutation matrix P[256][256]
-- initialize P[256][256]to all zeros
for i from 0 to 255
      S[i] = i
-- input random key k[N] in bytes with N =256
for i from 255 down to 1 do
   j = k[i]
   swap S[j] and S[i]
for i from 0 to 255
      P[i][S[i]] = 1
```

The pseudo-code requires a random input secret key of length 256 bytes for each permutation matrix. This can be extended for smaller or up to 4KB based on the actual security consideration with a suitable key expansion algorithm to make its length to 256 bytes per permutation matrix.

III. PROPOSED AES-QPP

It is a natural thought to unify both SubBytes and AddRoundKey with the same QPP of 16 8-bit permutation matrices and maintain the original ShiftRows and MixColumns steps. Then a typical round in this AES-QPP becomes QPP + ShiftRows + MixColumns + QPP

- SubBytes: → QPP of 16 permutation matrices selected from the 8-bit permutation group with the shared secret key materials
- ShitRows: no change
- MixColumns: no change
- AddRoundKey: → the same QPP as in the SubBytes step

Unlike the standard AES, the entropy injection into plaintexts in the encryption only exists in the AddRoundKey step with maximum 128 bits of entropy, AES-QPP can inject up to 32,678 bits of entropy through QPP permutation step. The actual entropy can be decided based on the security requirement.

In AES AddRoundKey, the encryption is performed by using 16 input bytes directly XORed with a round key of 16 bytes long. In AES-QPP, the encryption is similar to AddRoundKey step, but each byte in the input block is encrypted by the corresponding permutation matrix based on its byte integer value as a row index of the permutation matrix, and then the output is the column index of the non-zero element, this is the same way as in quantum gate operations.

The decryption is straightforward with QPP where permutation matrix should be transposed P^T because of its unitary and reversable properties.

A. Resistant to Differential Analyses

O'Connor (1993) [22] has shown that the highest probability p of the differential characteristic for an n-bit bijective transformation with a randomly chosen permutation matrix is at most $p = \frac{2n}{2^n}$ and for n = 8, $p = 2^{-4}$. It has been shown from [2] that the transformations of ShiftRows and MixColumns are equivalent to use active S-Boxes in the SubBytes step per round. Since the number of active S-Boxes for 4-round AES is at least 25 [2], the overall probability of the differential characteristic for 4-round AES is 2^{-100} or averagely 2^{-25} per round.

For a QPP of 16 randomly selected permutation matrices, this probability of the differential characteristic per round is averagely at $(2^{-25})^{16} = 2^{-400}$ for the proposed AES-QPP , which is less than the probability of the differential characteristic for the standard 14-round AES-256. Therefore, we then conclude that any differential analysis would not pose a security threat to AES-QPP.

B. Resistant to Linear and Integral Analyses

At the end of the MixColumns step, the 16 bytes of the output would again be non-linearly transformed by the same QPP of 16 permutation matrices. This can be considered as a natural extension from the XOR operations of AddRoundKey to generic bijective permutation transformations. This replacement eliminates the linearity between rounds and further weakens the linear cryptoanalysis.

Integral analysis [6] demonstrates that a single secret S-Box would exponentially increase the time complexity in terms of encryption equivalents from 2^{17} for 4-round AES-128 to 2^{90} for 6-round AES-128, in comparison with standard AES-128 from 2^{14} for 4-rounds to 2^{44} for 6 rounds. With the proposed AES-QPP round of 16 permutation matrices and assumption that the same integral analysis can be applied, the time complexity would be at the level $(2^{17})^{16} = 2^{272}$ for 4-rounds in terms of encryption equivalents. Based on this, we conclude that the integral analysis would not pose security threats to the proposed AES-QPP for more than 4 rounds.

Another benefit from AES-QPP is to eliminate the key schedule process which causes the round key correlations between different rounds. The proposed AES-QPP removes the dependence of implementation on the key length, that is, different key lengths would have the same implementation.

Based on the above security considerations, it possibly offers a good round reduction and leads to a better performance.

C. Round Reduction

In AES encryption, diffusion capability is majorly contributed through ShiftRows and MixColumns over a block then through repeating rounds. SubBytes only provides diffusion over GF(8). AES-QPP extends the diffusion capability from GF(8) to $GF(8)^{16}$, equivalent to extend the overall diffusion to 2048-bit field. This enlarged diffusion capability would allow us to reduce the number of rounds in AES. AES-256 needs 14 rounds to achieve good randomness in ciphertexts. We expect the number of rounds in AES-QPP to be reduced to 5 rounds.

D. Performance

The typical round in AES-QPP demonstrate a slightly less CPU time in comparison with the standard AES round. Therefore, the round reduction in AES-QPP directly adds benefits to performance gains: 3x encryption speed, lower about two 3rds of latency with about 1/3 of energy consumptions.

E. Footprint

The compiled AES-QPP is 1.39KB in comparison with 11.5KB for AES footprint in openssl. The sparse QPP matrices take 16*256 = 4KB memories, much more than AES memory usage 0.47KB. In order to reduce the memory usage, we may need to reduce the permutation matrices from 8-bits to 4-bits and use 32 permutation matrices for QPP, which can bring the memory usage down to 0.5KB.

IV. Discussions

The ciphertext indistinguishability is a great measure of a cryptosystem for security. The randomness analysis of the ciphertexts produced from a cryptosystem can be performed with the standard randomness testing tools such as NIST test suite. We have implemented the proposed AES-QPP in C and run all testing in the same system with AES-256. We then compared ciphertext randomness for the standard AES-256 ciphers with the proposed AES-QPP for 5 rounds and10 rounds. A single biased plaintext file of 100 MB is created by simply repeating an English sentence of 125 characters by 800,000 times. The same plaintext file is encrypted with the standard AES-256 and the proposed AES-QPP with 5 and 10 rounds in CBC-mode. The reason why we use a biased plaintext is because it can clearly demonstrate QPP's confusion and diffusion capability. It should be noticed that there is not required to pass randomness testing for ciphertexts produced from a cryptographic algorithm. However, ciphertexts with good randomness would increase the security of data encryptions against the statistical attacking. All output encrypted files are supplied to NIST 800-22 randomness testing suite and testing results are shown in Table 1. AES-256 and AES-QPP with 5 10 rounds pass all 15 randomness tests as what we expected, with exception of AES-QPP-10 where it failed the Overlapping Template test with their Chi-Square falling beyond 0.01-0.99. The testing results for AES-QPP indicate that the unification of SubBytes and AddRoundKey with QPP maximizes the diffusion capability and the number of rounds can be reduced to 5.

Table 1. NIST 800-22 randomness test reports are tabulated for the standard AES-256 and AES-QPP for 5 and 10 rounds. Overall randomness is observed for all ciphers.

Test name	AES-256	5 Rounds	10 Rounds
Frequency	Success	Success	Success
Block Frequency	Success	Success	Success
Cumulative Sums	Success	Success	Success
Runs	Success	Success	Success
Longest Run	Success	Success	Success
Rank	Success	Success	Success
FFT	Success	Success	Success
Non-Overlapping Template	Success	Success	Success
Overlapping Template	Success	Success	Failure
Universal	Success	Success	Success
Approximate Entropy	Success	Success	Success
Random Excursions	Success	Success	Success
Random Excursions Variant	Success	Success	Success
Serial	Success	Success	Success
Linear Complexity	Success	Success	Success

ENT is another interesting randomness testing tool created by John Walker [18]. It is not part of any official RNG evaluation scheme, but it has successfully identified flaws in RFID card key generators [19], particularly the DESFire EV1 [19], Mifare Classic and quantum random number generator Quantis devices [20]. Hurley-Smith, Patsakis and Hernandez-Castro [21] recently identified the unbearable lightness of FIPS 140-2 randomness tests with ENT. Although some image files, especially Webp format, can pass ENT randomness test, it is still the most sensitive testing tool to identify byte level bias in the testing data. It should be very interesting to perform ENT randomness testing with ciphers from AES-256 and AES-QPP. ENT testing provides six output statistics: entropy, compression, χ^2, serial correlation, arithmetic mean and Monte-Carlo estimated value for π. We use the same ciphertext files as used in NIST testing of Table 1 for ENT testing and results are shown in Table 2 for byte level and in Table 3 for bit level.

It is not surprising from Table 2 that AES-256 and both AES-QPPs successfully passes the six tests. For entropy test, they all show the same 7.999998 bits for a byte cipher data. They all produce excellent Chi-Square values around 256, especially for 5-round AES-QPP where it demonstrates a value of almost exactly 256 for Chi-Square, with a p-Value of 0.46. For 10-round AES-QPP, the Chi-Square values is slightly off of 256 with p-Values of 0.697. It is surprisingly

noticed that our AES-QPP with 5 and 10 rounds demonstrate much better randomness in their ciphers encrypted from very biased plaintexts than some hardware RNG [18, 19] and quantum random number generator Quantis devices [20]. The simple arithmetic mean is expected to be ideally 127.50 for byte randomness tests. AES-256 and AES-QPP are all passed for this test. All ciphers also give the very good Monte Carlo $\boldsymbol{\pi}$ value and show very small serial correlation for each byte to its previous byte.

Table 2. ENT tests with byte level randomness tests are tabulated for the standard AES-256, AES-QPP for 5 and 10 rounds in CBC mode, where Arith. Mean denotes arithmetical mean with an ideal value to be 127.50 and Serial Corr. denotes serial correlation coefficient measuring the extent to which each byte in the file depends upon the previous byte. Monte Carlo $\boldsymbol{\pi}$ indicates the Monte Carlo value for pi which should be close to 3.14159265. p-Value for Chi Square should be within 0.01-0.99 for good randomness data.

	AES-256	AES-QPP 5	AES-QPP 10
Entropy (bits)	7.999998	7.999998	7.999998
Chi Square	261.3	256.4	242.9
p-Value	0.380	0.463	0.697
Arith. Mean	127.51	127.51	127.50
Monte Carlo $\boldsymbol{\pi}$	3.141526685	3.141764329	3.141256669
Serial Corr.	0.000028	-0.000140	0.000047

As what we expected, AES-QPP (5 rounds) demonstrates extraordinary cryptographic characteristics with enhanced confusion and diffusion capabilities by leveraging the high entropy and strong diffusion from QPP over a 16-byte block. Through randomness analysis tools, the number of rounds in AES-QPP could be reduced to 5. This round reduction would boost AES performance by 3x times. In addition to this performance improvement, the proposed AES-QPP with 5 rounds also works with flexible input key length from 256 bits to 16x256 = 4 KB.

Table 3. ENT tests with bit level randomness tests are tabulated for the standard AES-256.

	AES-256	AES-QPP 5	AES-QPP 10
Entropy (bits)	1.000000	1.000000	1.000000
Chi Square	0.17	0.40	1.13
p-Value	0.678	0.527	0.287
Arith. Mean	0.5000	0.5000	0.5000
Monte Carlo $\boldsymbol{\pi}$	3.141526685	3. 141764329	3.141256669
Serial Corr.	0.000072	0.000049	0.000007

V. CONCLUSION

In the standard AES, SubBytes is a permutation defined by the multiplicative inverse of the input byte, designed with well-considered non-linearity. However, AddRoundkey performs linear transformations with XOR operators. These two steps belong to the same type of permutation transforms over the 8-qubit computational basis. That leads to a natural unification of SubBytes and AddRoundkey via QPP of 16 randomly selected permutation matrices based on the encryption key. The unified AES round becomes QPP + ShiftRows + MixColumns + QPP, called AES-QPP. We analysis the security improvement with this proposal against the differential and linear analysis. Our implementation and randomness analysis demonstrate that the number of AES-QPP rounds could be reduced to 5 rounds for quantum safe encryptions due to the extra diffusion contributions from QPP. The direct benefit from AES-QPP is the key length is no longer limited to 256 bits, but scalable based on the security requirement without changing the implementation. The round reduction would directly benefit to resource constraint IoT devices. In the future, we plan to explore AES-QPP for 4-bit QPP to further reduce memory space for QPP storage to allow AES-QPP to work for some special IoT devices.

ACKNOWLEDGMENT

We acknowledge N. Bettenburg for assisting with ENT testing. This research is partially funded by National Research Council – Industrial Research Assistance Program (NRC-IRAP) of Canada under project number 953186.

REFERENCES

[1] "Announcing the ADVANCED ENCRYPTION STANDARD (AES)" (PDF). *Federal Information Processing Standards Publication 197*. United States National Institute of Standards and Technology (NIST). November 26, 2001. Archived (PDF) from the original on March 12, 2017. Retrieved October 2, 2012.

[2] Joan Daemen and Vincent Rijmen, "*The Design of Rijndael*, AES - The Advanced. Encryption Standard", Springer-Verlag 2002.

[3] Heys, H.M., Tavares, S.E. "Substitution-permutation networks resistant to differential and linear cryptanalysis". *J. Cryptology* **9,** 1–19 (1996). https://doi.org/10.1007/BF02254789.

[4] O'Connor, L. On the distribution of characteristics in bijective mappings. *J. Cryptology* **8,** 67–86 (1995). https://doi.org/10.1007/BF00190756.

[5] Lu J., Dunkelman O., Keller N., Kim J. (2008) "New Impossible Differential Attacks on AES". In: Chowdhury D.R., Rijmen V., Das A. (eds) Progress in Cryptology - INDOCRYPT 2008. INDOCRYPT 2008. Lecture Notes in Computer Science, vol 5365. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-89754-5_22.

[6] Tiessen T., Knudsen L.R., Kölbl S., Lauridsen M.M. (2015) Security of the AES with a Secret S-Box. In: Leander G. (eds) Fast Software Encryption. FSE 2015. Lecture Notes in Computer Science, vol 9054. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-48116-5_9.

[7] Kazlauskas, K., Vaicekauskas, G., Smaliukas, R.: An algorithm for key-dependent S-box generation in block cipher system. Informatica. **26**(1), 51–65 (2015).

[8] Dara M, Manochehri K. "Using RC4 and AES Key Schedule to Generate Dynamic S-Box in AES." Information Security Journal: A Global Perspective. 2014;23(1-2):1-9.

[9] I. Das, S. Nath, S. Roy and S. Mondal, "Random S-Box generation in AES by changing irreducible polynomial," *2012 International Conference on Communications, Devices and Intelligent Systems (CODIS)*, Kolkata, 2012, pp. 556-559, doi: 10.1109/CODIS.2012.6422263.

[10] Shor, P.W. (1994). "Algorithms for quantum computation: discrete logarithms and factoring". *Proceedings 35th Annual Symposium on Foundations of Computer Science.* IEEE Comput. Soc. Press: 124–134

[11] Autre, K., Arya, K., *et al*. (2019). "Quantum supremacy using a programmable superconducting processor", *Nature* 574 (7779): 505–510.

[12] Wang, B., Hu, F., Yao, H. *et al.* "Prime factorization algorithm based on parameter optimization of Ising model." *Sci Rep* **10,** 7106 (2020). https://doi.org/10.1038/s41598-020-62802-5

[13] L. Grover, "A fast quantum mechanical algorithm for database search". *In: Proceedings of the 28th ACM STOC*, Philadelphia, Pennsylvania, pp. 212–219. ACM Press (1996)

[14] Bonnetain, X., Naya-Plasencia, M., & Schrottenloher, A. (2019). "Quantum Security Analysis of AES". *IACR Transactions on Symmetric Cryptology*, *2019*(2),55-93. https://doi.org/10.13154/tosc. v2019. i2.55-93.

[15] R. Kuang and N. Bettenburg, "Shannon Perfect Secrecy in a Discrete Hilbert Space," *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Denver, CO, USA, 2020, pp. 249-255, doi: 10.1109/QCE49297.2020.00039.

[16] Shannon, C.E. (October 1949). "Communication Theory of Secrecy Systems?" *Bell System Technical Journal.* 28 (4): 656–715.

[17] V. V. Shende, A. K. Prasad, I. L. Markov, and J. P. Hayes, "Synthesis of reversible logic circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 22, no. 6, pp. 710-722, June 2003.

[18] J. Walker, "ENT: a pseudorandom number sequence test program," Software and documentation available at www.fourmilab.ch/random/S, 2008.

[19] Darren Hurley-Smith, Constantinos Patsakis and Julio Hernandez-Castro. "On the unbearable lightness of FIPS 140-2 randomness tests", April 2020 *IEEE Transactions on Information Forensics and Security* PP(99):1-1, DOI: 10.1109/TIFS.2020. 2988505.

[20] D. Hurley-Smith and J. Hernandez-Castro, "Certifiably Biased: An InDepth Analysis of a Common Criteria EAL4+ Certified TRNG," IEEE Transactions on Information Forensics and Security, vol. 13, no. 4, pp. 1031–1041, 2018.

[21] D Hurley-Smith, J Hernandez-Castro. "Quam Bene Non Quantum: Bias in a Family Quantum Random Number Generators." *IACR Cryptol. ePrint Arch*. 2017, 842.

[22] O'Connor, L.: On the distribution of characteristics in bijective mappings. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 360–370. Springer, Heidelberg (1994)

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.

- **The below form will not be published, but it is necessary to help with the review process.**

Authors' Background

Name	Email	Position (Prof , Assoc. Prof. etc.)	**Research Field**	**Homepage URL**
Randy Kuang	Randy.kuang@quantropi.com	Chief Scientist	Quantum cryptography, post-quantum cryptography, quantum communications	www.quantropi.com
Dafu Lou	Dafu.lou@quantropi.com			www.quantropi.com
Alex He	Alex.he@quantropi.com			www.quantropi.com
Alexandre Conlon	Alex.conlon@gmail.com			