From Contemporary to Post-Quantum Cryptography: System Models, Threats, and Proposed Solutions

Apirath Limmanee
E-mail: apirath.li@kmitl.ac.th

Department of Engineering Education Faculty of Industrial Education and Technology
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520 Thailand

*corresponding author E-mail: apirath.li@kmitl.ac.th

(Received: November 18, 2019; Revised: December 02, 2019; Accepted: December 09, 2019)

ABSTRACT

In the presence of quantum computing and quantum key distribution (QKD), it is believed that the future of cryptography will take a sharp turn into a new direction. We propose one possible direction in this paper. A new abstract model is proposed. A brief review on the traditional secret-key cryptosystem and discussion on the credibility of the secure channel, as well as on theoretical and practical security, is given. After that, impacts from quantum computation are discussed, followed by presentation of interesting ideas from quantum key distribution (QKD) and wireless physical secret-key generation. Finally, possible integrated solutions are proposed, with some insight from the secure network coding technique.

Keywords: Quantum Computing, Cryptography, QKD

INTRODUCTION

One objective of this paper is to propose that the basic building block of post-quantum cryptography in the near future will take the following form.

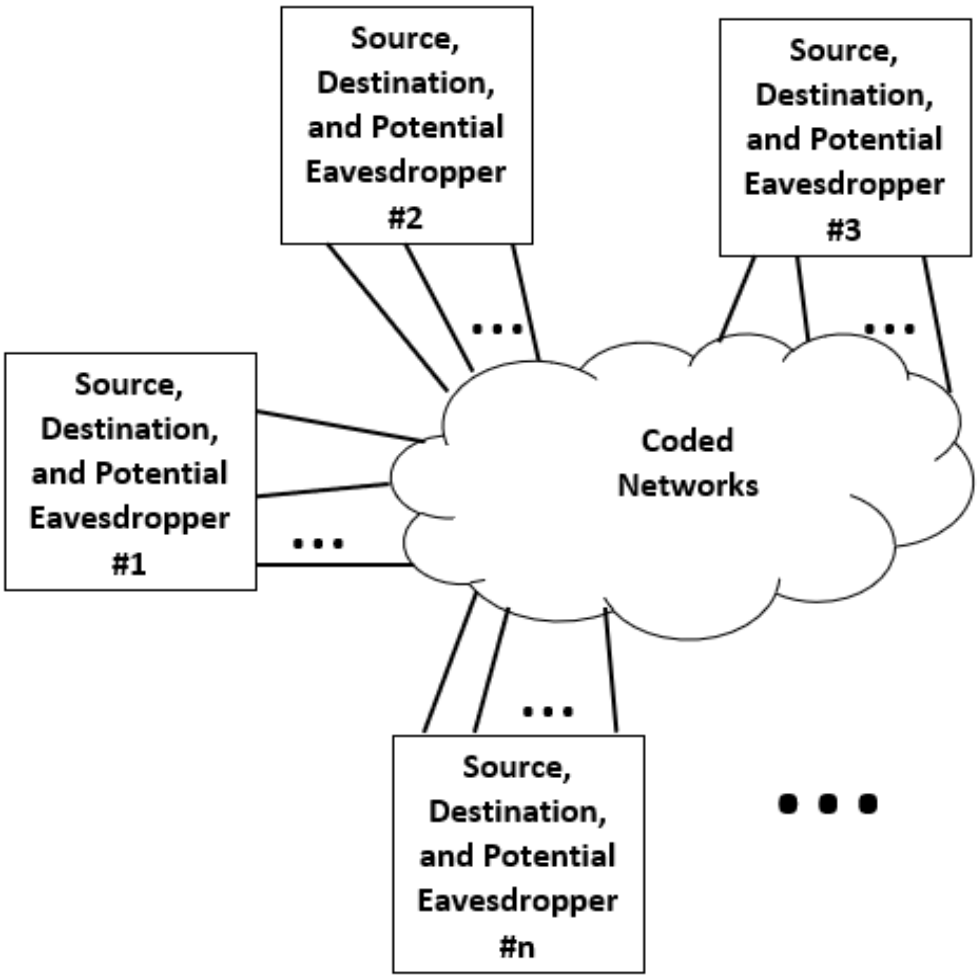


Figure 1. The Proposed Model

As the above picture is drawn conceptually as abstraction. Later, we will come to it to give examples of real-world systems. We will show that the above diagram is a generalized and convenient one to work with in post-quantum era. That is our second objective.

In order so to do, we start from discussing the traditional secret-key cryptosystem, then exclusively discuss the credibility of the secure channel, then elaborate on theoretical and practical security. After that, impacts from quantum computation are discussed and possible solutions are proposed. At the end, we will come back to Fig. 1 to conclude the article.

For the moment, let us consider the familiar block diagram in Fig. 2.

The Traditional Secret-Key Cryptosystem

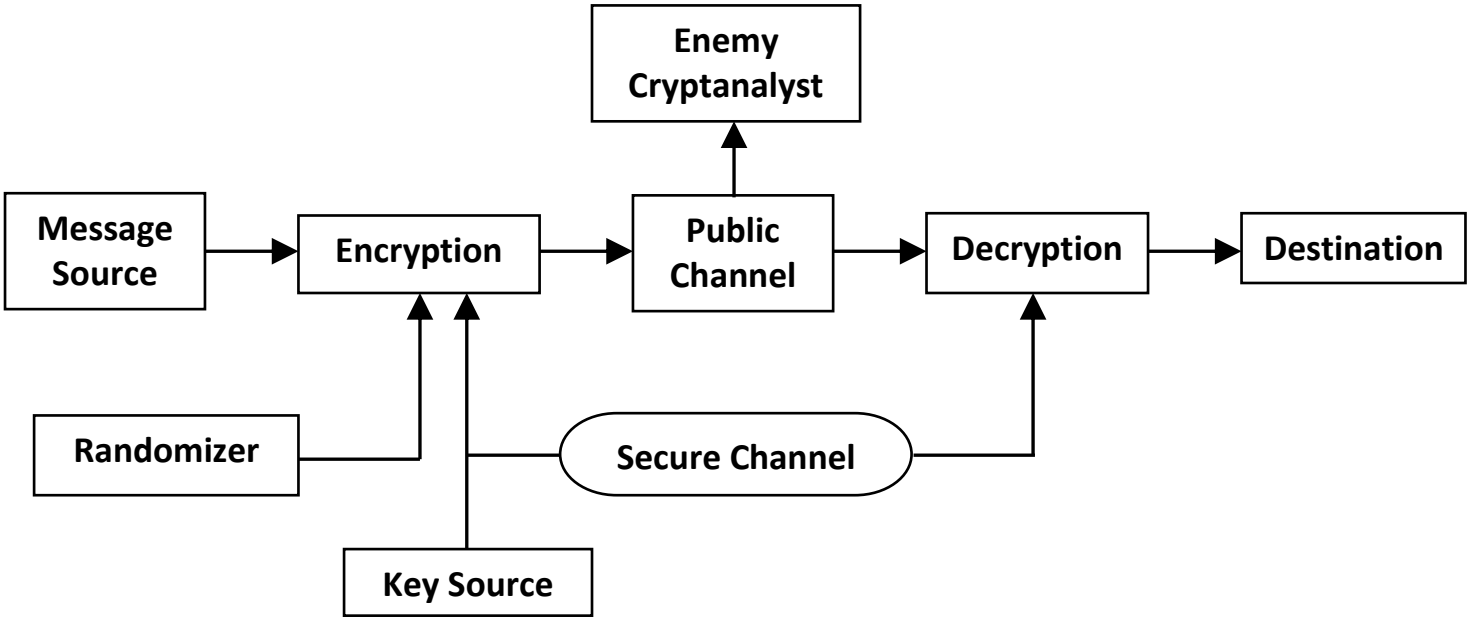


Figure 2. Traditional Secret-Key Cryptosystem [1]

In a way, the basic concept and building blocks of a secret-key cryptosystem have never changed for a long time, although the technology keeps progressing. Let us take a look at the following block diagram as I will go through every block except the message source and the destination. Also, the secure channel block is postponed to be discussed more deeply in the next section.

(1) Encryption seems to be the most important element in the diagram. Without it, we may not be able to call it a cryptosystem. This block has gone through so long a period of evolution that nobody can tell what or when the oldest form of encryption occurred. The first well-known encryption in human history is probably the one Julius Caesar used 2000 years ago for writing letters to Cicero, known as the Caesar cipher [2]. The well-known standard encryption schemes used nowadays are familiar names such as RSA, AES, ECC. Of course, some schemes are more secure than others. Within the same encryption scheme, the one using longer key is better protected. For example, AES-256 is more secure than AES-128.

In modern communications, we normally assume that the enemy knows which encryption scheme is used. The only reason that makes the system still secure is that the enemy does not know the secret key.

(2) Key source is also very important because secret key is needed in this model for the encryption purpose. As mentioned, longer key usually means better security. Indeed, Shannon has proved that in order to obtain perfect secrecy, one needs a key at least as long as the message itself. When the key is as long as the message, one does not need any encryption scheme more complicated than the “one-time pad.”

(3) The randomizer is sometimes added to make our cipher appear random. One of the oldest trick of the cryptanalysts is to employ the knowledge that the letter “e” is the most frequent one in English. So they first guess the symbol representing “e” before moving to other symbols. The randomizer aims at making sure that all our symbols are uniformly distributed.

(4) The public channel is used for the transmission of encrypted message. We assume that everybody including our enemy has an access to this channel.

(5) Decryption is the reverse process of encryption situated at the receiver.

(6) The enemy cryptanalyst is the bad guy trying to steal our information.

The Secure Channel ?

With the block diagram of the traditional secret-key cryptosystem, one may get curious about the secure channel part in the diagram, which is used for the transmission of secret key. One of the questions is that if we do have such a channel, why don’t we transmit our secret message through that channel directly? In that sense, all encryption schemes would be unnecessary.

In practice, we normally agree that, by secret channel, we mean another channel which is considered more secure than the public channel. Also, the secret channel should be more limited in terms of transmission capacity than the public channel or we will not need the latter. The implementation of secure channel has been the playground for imagination of cryptographers. In the past, it can be a spy in tuxedo carrying the secret key in his suitcase. For today’s wireless communications, the secret key may be stored in the electronic board from the start by the manufacturer. The key should be updated later because always using the same key will be vulnerable to eavesdropper’s attacks.

In modern communications, we normally assume that the enemy knows exactly how our data is encrypted. Thus, on the one hand, whether the system is secure depends on the security of the so called “secure channel” (hence the security of our key). On the other hand, the security level depends on the difficulty of breaking the cipher when the key is unknown or only partly known by the enemy.

This leads to two important question. Firstly, can we be sure that our secure channel is truly secure? Secondly, how do we determine the level of security, when we know that our key is partially or entirely safe? Although it may seem that we should start from answering the first question, we feel that starting with the latter is easier to explain. So we stop doubting the credibility of the secure channel at the moment.

Theoretical Security versus Practical Security

Although we know for sure our key is safe. The level of security varies according to our encryption scheme and key length. In this section, we introduce two concepts used for determining the security level, the theoretical security and the practical security.

Theoretical security is sometimes named as Shannon security after its originator, Claude Shannon. It is based on the information-theoretic concept, explained informally as follows: The level of theoretical security depends on how the cipher correlates with the original message. In information theory, we use the term “mutual information” to measure the correlation amount between the cipher and the original message. If there is no mutual information between them, it means one can learn nothing about the original message by looking at the cipher alone. This condition is called perfect secrecy. If there is some mutual information, then the security level decreases with the increasing mutual information.

If our original secret message is denoted by the random process **X** whereas the ciphertext is denoted by **Y**, the mutual information between them is denoted by **I(X;Y)**. The perfect secrecy is attained if and only if

$$\mathbf{I(X;Y)} = 0.$$

We can also write the perfect secrecy condition in terms of entropy as follows.

$$H(\mathbf{X|Y}) = H(\mathbf{X})$$

The above equation says that perfect secrecy is achieved if the conditional entropy of **X** given **Y** is the same as the entropy of **X**. In other words, perfect secrecy is achieved if it does not matter to let the enemy knows about the ciphertext **Y**, for this does not affect his knowledge about the secret message **X**.

Shannon proved that, in order to achieve perfect secrecy, one needs the secret key that is as long as the secret message. In that case, we can use a very simple encryption method called one-time pad or Vernam's cipher as follows [3][4].

$$y = x \bigoplus z$$

y is the cipher bit obtained by XORing the message with the key bit by bit. That is why we need the key as long as the message. We can see that as long as the enemy does not know the key z, he does not know the message x either.

Although Shannon's idea of theoretical security is proposed since 1949 [3], the idea is almost forgotten today as it is impractical to generate the key as long as the message and transmit it securely. Encryption today is based on the idea of practical security. For practical security, shorter key is generated and perfect secrecy is not achieved. However, the encryption scheme used is more complicated and has two important properties also suggested by Shannon. Those properties are called diffusion and confusion [3]. The former means the spreading the influence of a single plaintext bit over several ciphertext. The enemy will find it difficult to know which ciphertext bits refer to a particular plaintext bit. Ideally, if we change a single plaintext bit, half of the encrypted ciphertext should be changed. The confusion property is similar to the diffusion one but it intends to hide the relationship between the cipher and the key instead of that between the cipher and the plaintext.

Shannon's practical security becomes the basic concept in some of today's standards, such as AES and DES. Shorter key needed makes them practical. There is one difficulty, however, in transmitting the key via secure channel which may or may not truly secure.

In 1976, the era of public-key cryptography begins with Diffie and Hellman's proposed use of one-way functions [5]. The one-way function is easy to calculate but very computationally demanding to find the reverse (decrypt), when one does not have the private key. Decryption of public-key cryptography requires both public and private keys. The former is known publicly to everyone including the enemy, whereas the latter is known only by the legitimate receiver. The advantage of public-key over private-key cryptography is the ease of key management. The same public key is used in encryption for any intended receiver, who will use its own private key to decrypt the message. In this case, it is more practical in the sense that we do not need to implement the secure channel, for the key is transmitted publicly, as shown in Fig. 3. Public-key cryptography has become the concept of today's most popular (although not most secure) encryption scheme, the RSA.

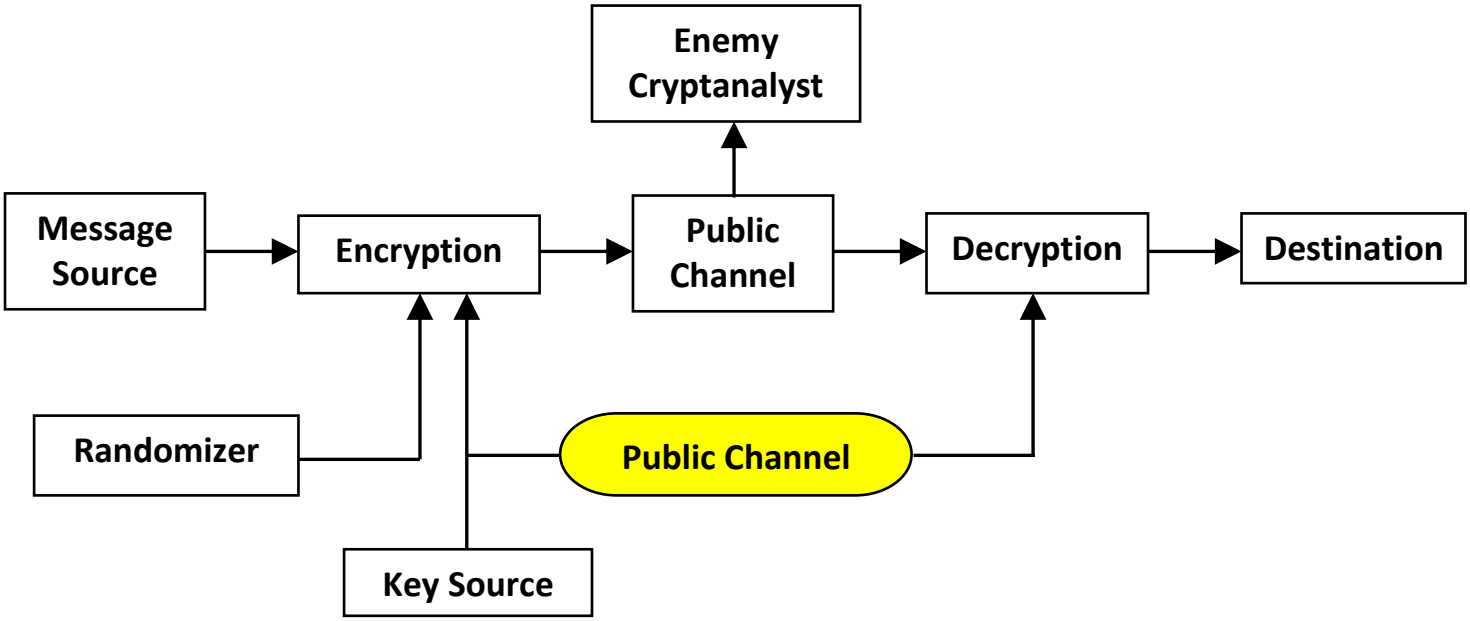


Figure 3. Public-Key Cryptosystem

We can conclude that practical security is indeed not secure. It is only extremely difficult and take too long to attack the ciphers with classical computer. We are reasonably safe, until …

Is Practical Security still Practical with Quantum Cryptanalyst?

What happened if we assume that our enemy possesses a quantum computer? Of course, the situation changes dramatically. With a quantum computer, the cryptanalyst is equipped with the potentially most powerful tool available to break the code.

Many of us know that quantum computers are much more powerful than classical ones. Indeed, its power increases exponentially with the number of available "qubits" as compared with linear increase with the number of bits in classical computers. Quantum computers have, however, some drawbacks waiting to be alleviated. For example, noise problems are difficult to deal with and the probability of obtaining correct computational result is not 100%.

Apart from the development of quantum computer hardware, the development of quantum software is also necessary for the success of cryptanalysis. We cannot write quantum algorithms in the way we are writing codes for classical computers. One must take into account the differences in hardware, such as the quantum logic gates, in order to program a quantum computer.

The first quantum algorithm that heavily threatened the cryptographical world was invented by Peter Shor in 1994 [7]. Whether his first intention is to challenge the cryptographers or not, we cannot imply from his seminal paper. However, his paper describes how to efficiently solve the integer factorization problem in polynomial time using quantum computers. Note that the NP-hardness of this problem in classical computation has for years guarantee the security of the RSA public-key algorithm. With quantum computers, today's most popular encryption has become unsafe.

Cryptographic Algorithm	Type	Purpose	Impact from large-scale quantum computer
AES	Symmetric key	Encryption	Larger key sizes needed
SHA-2, SHA-3	----------	Hash functions	Larger output needed
RSA	Public key	Signatures, key establishment	No longer secure
ECDSA, ECDH (Elliptic Curve Cryptography)	Public key	Signatures, key exchange	No longer secure
DSA (Finite Field Cryptography)	Public key	Signatures, key exchange	No longer secure

Table 1. Impact of Quantum Computer on Cryptographic Algorithms [8]

Is Cryptography Dead or Is This the Return of Shannon's Theoretical Security?

We can see that all public key schemes in Table 1 are now unsafe. Even the private key scheme, AES, widely regarded that today's most secure encryption, is affected. With the continual development of quantum computers, all available encryption schemes will sooner or later become unsafe.

Is this the dead of cryptography? For cryptography to survive, new ideas might be needed. There is current discussion about quantum-resistant cryptography or post-quantum cryptography which are more or less the same thing, cryptography designed to be safe in spite of the existence of quantum computation. USA's National Institute of Standards and Technology (NIST) has already set up a competition for the post-quantum standard. It recently announces candidates passing the second round of the competition, almost half of which are of the lattice-based encryption family known to be more difficult for quantum computers to attack [9].

In this article, we would like to suggest another alternative because we feel that the solution not only exist in novel encryption schemes, but also in the classic, long-forgotten concept. We are talking about the Shannon security!

Quantum Key Distillation (QKD): Implementation of Truly Secure Channel

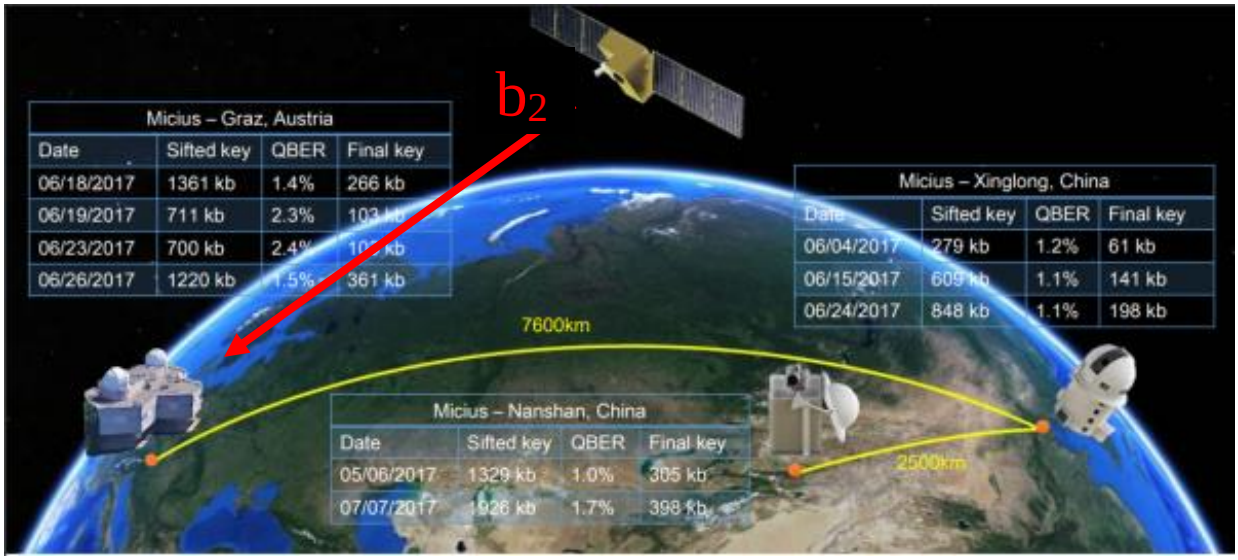
Recall two major problems that render Shannon security concept impractical. First, It requires that the key is generated at the same rate as the data. Second, truly secure channel is needed to transmit this very long key. With the idea of QKD, the second problem might no longer be a problem.

One of the most exciting quantum experiments in 2018 was performed in cooperation between China and Austria [10,11]. It successfully implemented the idea of quantum key distillation (QKD). It follows the most famous version of QKD proposed in 1984 by Bennett and Brassard (and thus called BB84) [12].

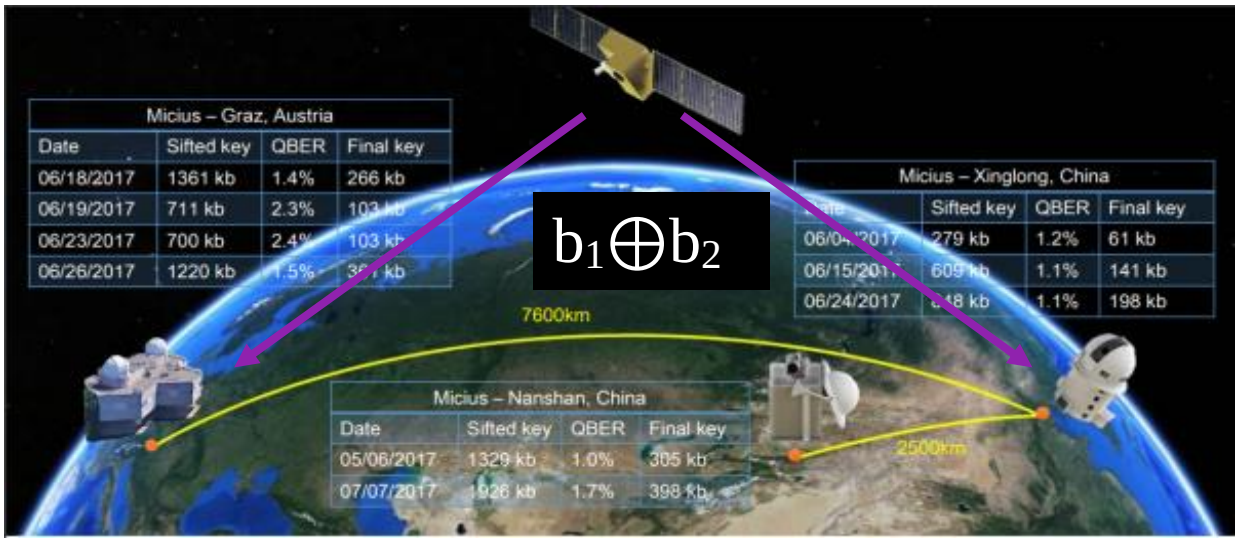
QKD is based on quantum physics uncertainty principle as well as the concept of entanglement. Two entangled photons, when not yet measured, are in the uncertain state. Once one of them is measured, both will automatically appear in the same state. Thus, by transmitting one of two entangled photons to the receiver and subsequently making measurement, we can use the measured stated as the mutual secret key. Moreover, if somebody eavesdrop the photon, the legitimate transmitter-receiver pair will know this because any quantum measurement will alter quantum property. Using this idea, truly secure channel can be implemented.

In the experiment, two pairs of entangled photons are generated at the satellite "Micius." One photon of the first pair is transmitted to Xinglong and another is kept at the satellite. The second entangled pair is processed similarly, one being sent to Graz and another kept. After measurement, one key symbol is obtained for each pair. Let us say that b_1 denotes the symbol measured at Xinglong and b_2 that measured at Graz. The satellite knows both b_1 and b_2 and therefore transmit $b_1 \oplus b_2$ via classical (non-quantum) satellite channel. The station at Xinglong deducts b_1 from $b_1 \oplus b_2$ to obtain b_2, whereas that at Graz deducts b_2 from $b_1 \oplus b_2$ to obtain b_1. After this process, both station knows both b_1 and b_2 and can concatenate them to form a mutual secret key. The key is later employed for one-time pad encryption of secret image and AES encryption for videoconference [10,11]. Note that higher data rate in videoconference makes it impractical to generate sufficiently long key for one-time pad. So AES is used instead.

(a)



(b)



(c)

Figure 4. QKD Satellite Experiment (a), (b) Transmission via Quantum Channel, and (c) via Classical Satellite Channel [10]

Physical Layer Key Generation: Wireless and Quantum

In the previous section, we discuss transforming quantum state into secret key. We are excited not only because we can implement a truly secure channel but also because, perhaps to a greater extent, it exhibits nature's counter-intuitive phenomena of quantum entanglement and Heisenberg's uncertainty principle.

However, focusing on the implementation of a truly secure channel alone, we can see that there is another physical phenomenon, less exciting but probably more practical, that can be used to generate secret key. We are talking about generating and transmitting secure key from the ordinary wireless channels instead of quantum ones.

It is widely known that wireless channel coefficients, characterized by their phases and amplitudes, depend heavily on the location, the environment, and the movement of transmitter and receiver to the extent that other terminals except the two can predict almost nothing about their channel parameters, and hence the resultant secret key.

The generation of the secret key consists of two steps, deriving the channel estimates before quantizing them into secure key symbols. Channel estimates can be derived using a known pilot sequence, which is transmitted back and forth between transmitter and receiver such that they can learn about channel coefficients from the symbols distorted by the channel [13]. The outcome of the channel estimation process is a set of complex channel coefficients which must be quantized into secret key symbols, as shown in Fig. 5. We can see that the secure channel in Fig. 2 is implemented by the key-generating channel in Fig. 5. The key source in Fig. 2 is implemented by the combination of a channel estimator and a quantizer in Fig. 5. The randomizer in Fig. 5 is shown by dashed lines to indicate that it may be needed or not
according to the encryptor used. In case we generalize our model to include both quantum and wireless key-generating channel, in the latter case incorporating the quantizer, channel estimator, and pilot-sequence source into encryption and decryption blocks. We will have a more compact model shown in Fig. 6.

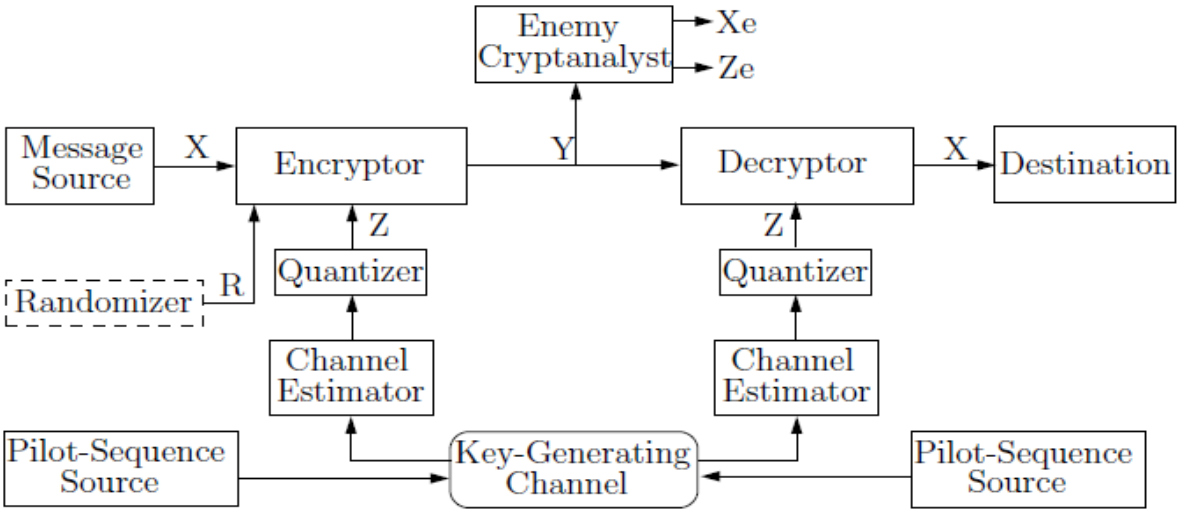


Figure 5. Secret-Key Cryptosystem with Wireless Physical-Layer Key Generation [14]

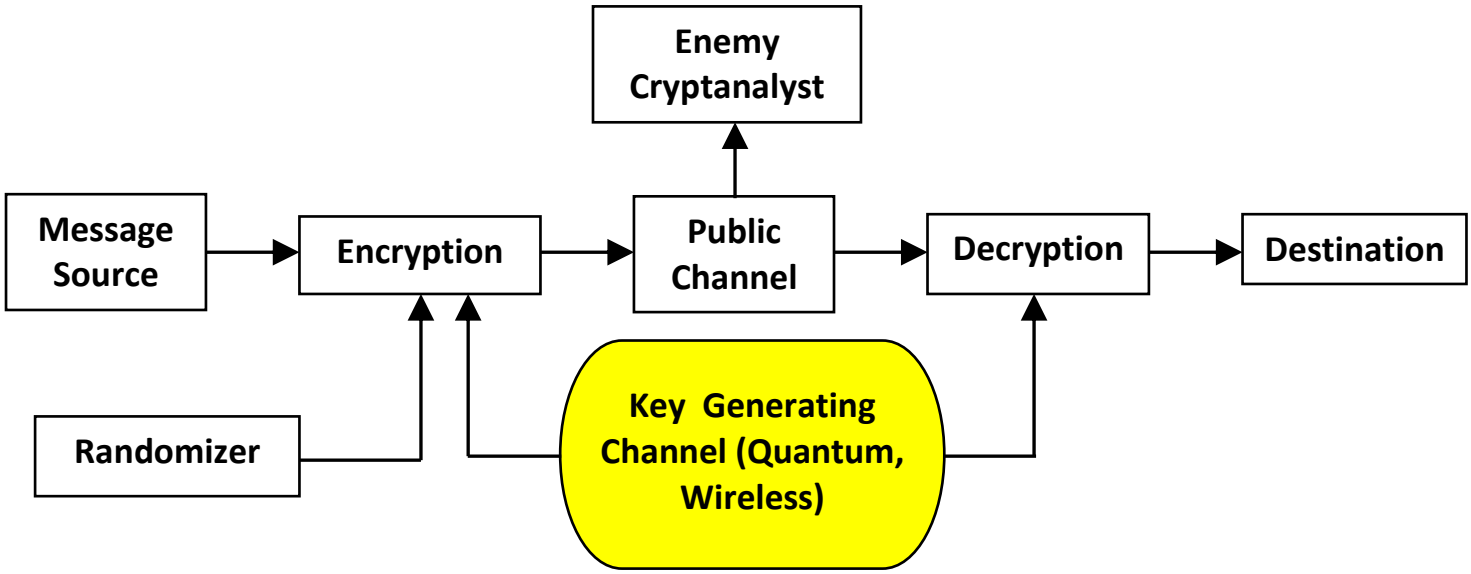


Figure 6. Generalized Model of Physical-Layer Key Generation

Secure Network Coding and the Variable Key Generation Rate

So far we have shown that the quantum and wireless channels allow us to generate random key and provide truly secure channel for its transmission. However, if we come back to consider the Shannon security concept. One important question remains unanswered: Is it possible for us to generate long enough key such that perfect secrecy is achieved? This penultimate section attempts to answer this question.

Recall the quantum satellite experiment. Note that b_1 and b_2 are simultaneously generated before being concatenated together, thus forming a longer key. If this key generation rate is not quick enough, one can add another satellite in our imagination so that we also simultaneously have b_3 and b_4 , thus doubling the rate. The fact that the two satellites can send photons to each other add another b_5 into the equation and the rate is now 2.5 times as compared to the original setting. Indeed, if we are rich enough to keep increasing the number of satellites, we can form a network of key-generating quantum channels which increases quadratically with a linear increase in the number of satellites.

If we are not rich enough and contend with the wireless channels instead. We can just replace the satellites with the nodeMCUs in our lab and can generate secret key as long as our funding allows. If our algorithm are efficient and our equipment is good enough (perhaps with higher quality than normal nodeMCUs), we can perform one-time pad for higher data transmission rate.

Let us now consider the Shannon security derived from this quantum or wireless kind of network setting. Starting from the Chinese-Austrian satellite experiment, we can reason that if b_1 and b_2 are concatenated to form a longer key, perfect secrecy is not really achieved. This is because, by sending $b_1 \oplus b_2$ via classical satellite channel, the enemy satellite can possibly intercept this information. The enemy will not know exactly what b_1 and b_2 are. Yet, by learning about the linear combination $b_1 \oplus b_2$, if the enemy happens to correctly guess either b_1 or b_2, he can easily decode for another one.

Of course, $b_1 \oplus b_2$ needs to be transmitted. In order to make the system perfectly secure, we need to sacrifice key generation rate for better security. Instead of concatenating b_1 and b_2 together, we can decide to use just one of them. This halves the key generation rate but now the enemy faces a much harder task of having to guess every key bit correctly. Perfect secrecy is achieved in this case.

Generalizing to the quantum/wireless key-generating network discussed previously, Other linear combination than just $b_1 \bigoplus b_2$ is required to be transmitted. The choice of how key bits should be linearly combine belongs to the field of secure network coding [15,16]. Unfortunately, key rate cannot increase with the network size if strict perfect secrecy is demanded. However, if we are willing to sacrifice security level for key generation rate, we can increase the rate with the network size. At the end, it is a tradeoff depending on whether the security or the data rate is more important.

CONCLUSION

Quantum technology is disrupting the cryptographical world and we should prepare for the consequences. We would like to end this article by fulfilling the objective given at the beginning, i.e. to propose the model re-presented again in our last figure below. By physically generating key from quantum/wireless networks, the model of cryptosystem becomes a coding network with several eavesdropper listening for the linear combinating clues of $b_1 \bigoplus b_2$, $b_3 \bigoplus b_9$, or $b_{200} \bigoplus b_{201} \bigoplus b_{202} \bigoplus \ldots$ depending on how our coded network is designed. If it looks as if the proposed model is only our idea which are not applicable elsewhere, we leave it for our dear reader to kindly generalizing other systems to our model for us. Certainly not a difficult assignment?

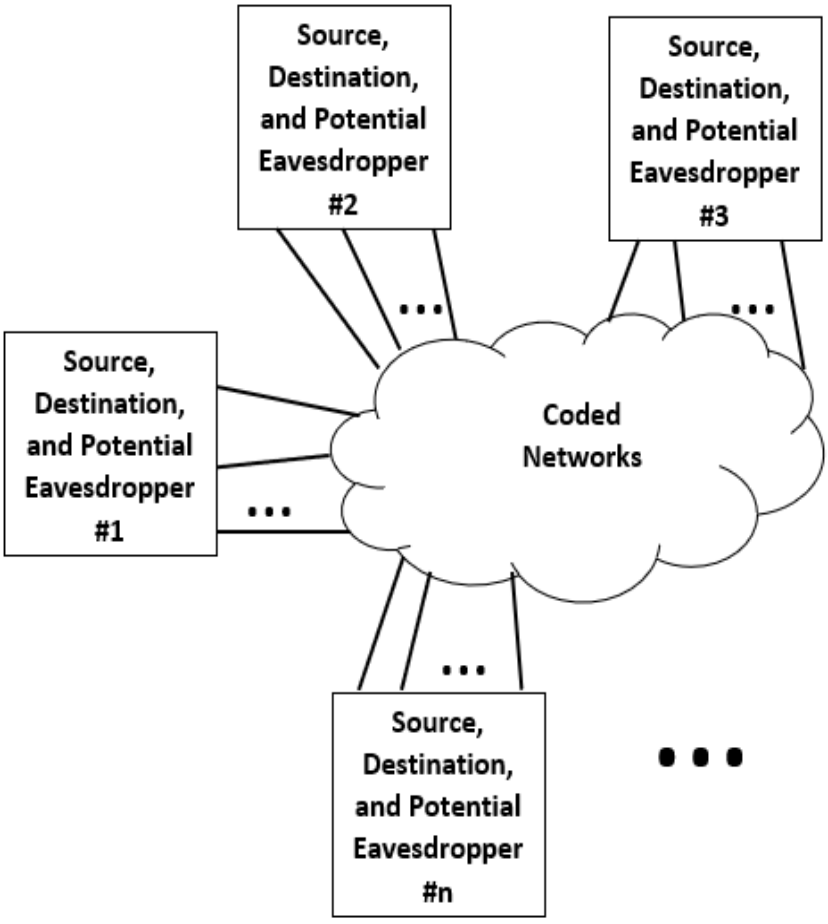


Figure 7. Final Proposed Model

REFERENCES

[1] J.L. Massey, “An introduction to contemporary cryptology”, Proceedings of the IEEE, 76, pp. 533–549, 1988.

[2] D. Kahn, The Codebreakers, The Story of Secret Writing. New York, NY: MacMillan, 1967.

[3] C. E. Shannon, “Communication theory of secrecy systems,” Bell Syst. Tech. I., vol. 28, pp. 656-715, Oct. 1949.

[4] C. E. Shannon, “A mathematical theory of communication,” Bell Syst. Tech. I., vol. 27, pp. 379-423, 623-656, July and Oct. 1948.

[5] W. Diffie and M. E. Hellman, “New directions in cryptography,” IEEE Trans. Information Theory, vol. IT-22, pp. 644-654, Nov. 1976

[6] T.S. Metodi, A.I. Faruque, and F.T. Chong, “Quantum Computing for Computer Architects,” Morgan and Claypool Publishers, 2006.

[7] P.W. Shor, “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer,” 35th Annual Symposium on Foundations of Computer Science, pp. 124–134, 1994. DOI: 10.1103/PhysRevA.52.R2493 3, 19, 23, 30, 76, 91, 131

[8] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, “NIST: Report on Post-Quantum Cryptography,” NIST, Tech. Rep., 2016.

[9] A. Khalid, S. McCarthy, M. O’Neill, and W. Liu, “Lattice-based Cryptography for IoT in A Quantum World: Are We Ready?,” In 2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI), pp. 194-199, Jun. 2019.

[10] S.-K. Liao, W.-Q. Cai, J. Handsteiner, B. Liu, J. Yin, L. Zhang, D. Rauch, M. Fink, J.-G. Ren, W.-Y. Liu, Y. Li, Q. Shen, Y. Cao, F.-Z. Li, J.-F. Wang, Y.-M. Huang, L. Deng, T. Xi, L. Ma, T. Hu, L. Li, N.-L. Liu, F. Koidl, P. Wang, Y.-A. Chen, X.-B. Wang, M. Steindorfer, G. Kirchner, C.-Y. Lu, R. Shu, R. Ursin, T. Scheidl, C.-Z. Peng, J.-Y. Wang, A. Zeilinger, and J.-W. Pan, “Satellite-Relayed Intercontinental Quantum Network,” Phys. Rev. Lett. 120(3), 2018.

[11] S.-K. Liao et al., “Satellite-to-Ground Quantum Key Distribution,” Nature, vol. 549, pp. 43–47, Aug. 2017.

[12] C.H. Bennett, and G. Brassard, “Quantum Cryptography: Public Key Distribution and Coin Tossing,” In Int. Conf. on Computers, Systems & Signal Processing, pp. 175–179, 1984.

[13] J. Wallace, ”Secure Physical Layer Key Generation Schemes: Performance and Information Theoretic Limits,” IEEE Int. Conf. Communications, Dresden, Jun. 2009.

[14] A. Limmanee, and W. Henkel, “Secure Physical-Layer Key Generation Protocol and Key Encoding in Wireless Communications,” In GLOBECOM Workshops, pp. 94-98, Dec. 2010.

[15] N. Cai, and R.W. Yeung, “Secure Network Coding,” Int. Symp. Information Theory, Jun. 2002.

[16] K. Bhattad and K.R. Narayanan, “Weakly Secure Network Coding,” in Proc. NETCOD, Apr. 2005.