2018 IEEE Conference on Decision and Control (CDC)
Miami Beach, FL, USA, Dec. 17-19, 2018

Enhancing the Security of Multi-agent Networked Control Systems Using QKD based Homomorphic Encryption*

Hai-Jin Ding, Zi-Xiao Wang, Re-Bing Wu and Qian-Chuan Zhao[1]

Abstract— **The cyber-security of multi-agent control systems has become vital in practice. To protect the communication process between the individual agents over TCP/IP networks, one must encrypt the messages to be sent, which is costly in a large-scale network when high-level security is demanded. To lessen the burden of heavy encryption and decryption processes, we introduce the Homomorphic Encryption to reduce the number of encryption/decryption terminals that are supposed to equip with every agent. The security is enhanced by the quantum key distribution technology, which can generate secret keys that are theoretically absolutely secure even against quantum computation. We proposed a hybrid method that make good use of the randomness of quantum keys, one-time pad and symmetric encryption to make sure the overall security of homomorphic encryption algorithms. Numerical simulations are provided to illustrate how our scheme.**

I. INTRODUCTION

The modern cyber-physical systems involve a large amount of multi-agent control systems, in which the communication is exposed to potential cyber-attacks. The resulting cyber security issue was addressed in [1], where the basic design and control strategy of agent-based control are proposed. The existence of cyber attacks will lead to information leakage and the circumstance in multi-agent network, the commonest organization of social life, is much more urgent [2]. To theoretically model the problem, Liu and Basar [3] conclude the topology designing of multi-agent network to a mathematical problem of graph by taking convergence rate and security into consideration together and discussed some simple topology examples, which is quite helpful for the following study on the security of complex network.

The security protection of the entire agent-based network becomes more difficult because of the intense and complicated communication between many agents. In practice, one can encrypt the messages to be sent, but the resulting cost on encryption, decryption and communication will dramatically increase. In large-scale systems, the complicated cryptographic process may even degrade the control performance. Therefore, under realistic circumstances, there is always a trade-off between the security, the cost and the performance.

*This work was supported by TNlist and National Natural Science Foundation of China (Grant Nos. 61374091, 61134008, 11175094 and 91221205), the National Basic Research Program of China under Grant No.2011CB9216002. GLL also thanks the support of Center of Atomic and Molecular Nanoscience of Tsinghua University.

[1]The authors are all with the Department of Automation, Tsinghua University, and also with the Center for Quantum Information Science and Technology (TNlist), Beijing 100084, China (e-mail: rbwu@tsinghua.edu.cn)

Note that in practical multi-agent networked control systems, some agents only collect, process and transfer information from and to other agents. In other words, they do not produce new information (e.g., generating measurement data and control commands). We indicate that decryption and encryption at these agents can be skipped without sacrificing the security. In this way we can save the (usually large amount of) hardware and software resources for encrypting and decrypting data at these agents, and the network performance can be improved. The technologies behind this idea are the homomorphic encryption and Quantum Key distribution to be introduced in this paper.

Homomorphic encryption (HE) is an advanced encryption technology that makes it possible to do arithmetics only with the cipher text and without decryption and re-encryption. Kogiso and Fujita [4] for the first time used homomorphic encryption in networked control systems and an encrypted controller based on RSA scheme. Homomorphic encryption can be divided as Somewhat Homomorphic Encryption (SWHE) and fully homomorphic encryption (FHE), the former is able to finish some finite functions such as adding and multiply transformation while the later can calculate the product of two cipher texts without decrypting. However, for most networked control systems, SWHE is sufficient to accomplish most of the controlling algorithms. Kim *et al.* [5] used FHE to encrypt the signals of controllers and analyzed the bootstrapping of cipher texts. Farokhi *et al.* [6] used semi-homomorphic encryption, a simplified homomorphic encryption algorithm, in the controller of networked control systems. It can be seen that homomorphic encryption makes it possible to design a multi agent distributed networked control system only by exchanging the cipher text.

The security of HE mainly relies on the RSA algorithm, which is essentially determined by the hardness of factorizing a large integer (i.e., the public key). RSA is traditionally thought of strong when the integer is sufficiently large, but it is recently threatened by the fast developing quantum computation technology. Using the famous Shor's algorithm [7], the factorizaiton of an integer can be exponentially accelerated and thus RSA-based HE can be easily cracked. This may occur in a forseeable future.

To resolve this problem, we introduce the QKD to strengthen the RSA-based HE, which combines the symmetric cryptography using securely distributed keys. The reminder of this paper is as follows. In Section II, we designed a agent based networked control system encrypted with Homomorphic Encryption. In Section III, we analyzed the security problems of the homomorphic encrypted system

978-1-5386-1395-5/18/$31.00 ©2018 IEEE

and enhanced its security with quantum key distribution and one-time pad. In Section IV, we analyzed the error of homomorphic encryption and its influence on performance. Finally, in Section V conclusions were made.

II. DESIGNING OF AGENT BASED NETWORKED CONTROL SYSTEM

For illustration, we study the following leader-follower model that involves the leader, the plant and the controller [8]. The controller can be a third port, and even can be calculated via cloud computing. In the meantime, the confidential security of data should be ensured. We study on the lead-follower model which can be used to track a mobile target and the control model is showed in Fig.1.

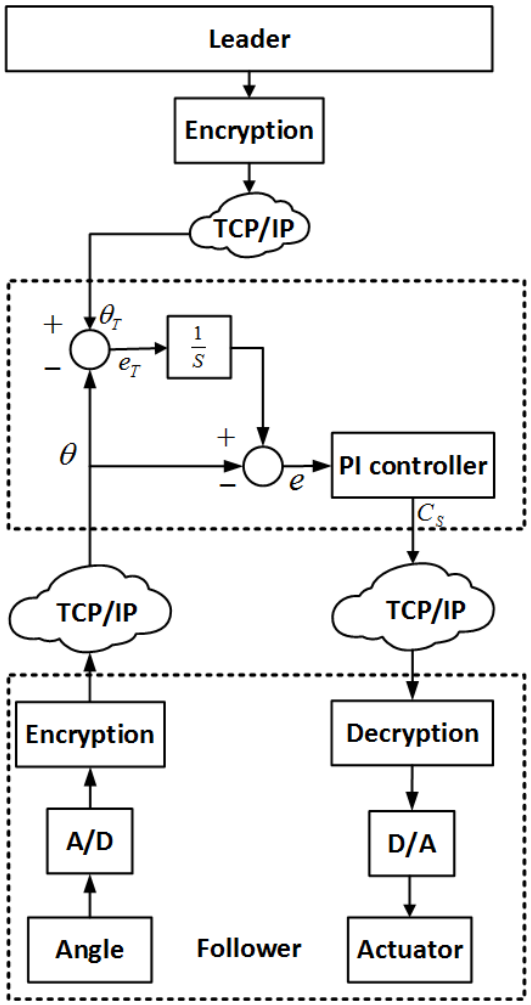

Fig. 1. Diagram of multi-agent leader-follower control model.

As is showed in the figure above, the tracker can calculate the angular distance between target and follower, it is a integrating element and its pole can help make sure accurate tracking. In practical tracking systems there are usually two sensors, one is used to find the target, for example it can be a radar located in another IP. While the other is to sense the position of follower's servo.

Among the agents, as an actuator, the servo no longer needs to take the controller and sensor altogether with it and only need to exchange data via communication. This will help release the servo's load and enhance the quickness of response.

The mathematical model of system can be showed as follows:

- θ_T —- The real-time position of target;
- θ —- The real-time position of servo;
- e_T —- The tracking error between servo and target;
- e —- The control input of servo;
- C_S —- The command produced by PI controller to servo;

$$e_T(t) = \theta_T(t) - \theta(t), \tag{1}$$

$$e(t) = \int_0^t e_T(t)dt, \tag{2}$$

$$C_S(t) = K_P e(t) + K_I \int_0^t e(t)dt. \tag{3}$$

III. SECURITY ANALYSIS OF MULTI AGENT NETWORKED CONTROL SYSTEM

Now we analyze the security of multi-agent NCS. The positions of target and servo are the two major data sources of the system. The controller is an intermediate calculating link. The optimal security strategy is that the data should be plain text only in necessary agents, and in other agents the data should be exist as cipher text. To make sure the privacy of data as well as finishing the process of tracking and controlling in multi-agent network should be a challenging work. In the following, we introduce the homomorphic encryption and analyze how it improves the security.

A. Security of Homomorphic Encryption

Homomorphic Encryption is based on RSA [9] whose security lies on the mathematical complexity of calculating the factorizing large integers. The most popular homomorphic encryption algorithm is Paillier proposed in 1999 [10]. There are three parameters associated with Homomorphic Encryption, namely p, q and g. All the three parameters are prime numbers and p, q should be large enough to guarantee the difficulty of brute-force attack. The encryption and decryption agents, namely leader and follower, know the value of prime numbers p, q and g, while the homomorphic calculating agent only need to know the value of $p \times q$ to finish add operation and scalar multiplication, which is sufficient for the PI controller.

The process of Paillier algorithm is follows:

- Choose two prime numbers p and q, such that $\gcd(pq, (p-1)(q-1)) = 1$,
- $n = pq$, and $\lambda = \frac{(p-1)(q-1)}{\gcd((p-1),(q-1))}$.
- Choose a random prime number g and make sure that $\mu = (L(g^{\lambda} \bmod n^2))^{-1} \bmod n$ can be calculated, where $L = \frac{u-1}{n}$.

Suppose that we want to send an integer m, the encryption of m can be done as follows:

$$c = g^m r^n \mathrm{mod} n^2, \tag{4}$$

The decryption for cipher text c can be showed as follows:

$$m = L(c^\lambda \mathrm{mod} n^2 \mu \mathrm{mod} n). \tag{5}$$

As illustrated above the message is encrypted with public key(n, g) and decrypted with private key (λ, μ), and the security lies on the fact that the eavesdropper has to calculate p and q given n which is large enough. However the security of Homomorphic Encryption has been challenged by the development of computation power, especially with the quick pace of quantum computation.

B. Enhancing security of Homomorphic Encryption using QKD and one-time pad

The security of encryption is a combination of security of keys and complexity of encryption algorithms. While the security of keys is the base of the overall system security. Because usually we cannot make sure the unconditional security of sharing keys, we turn to enhance the complexity of encryption algorithms to increase difficulty for eavesdropper to crack cipher text. However, considering the challenges of quantum computation on traditional RSA-based encryption, the security of keys will directly determine the security of encryption. We combine quantum key distribution with Homomorphic Encryption to make sure the security under the challenge of quantum computation. The whole system is showed in Fig.2.

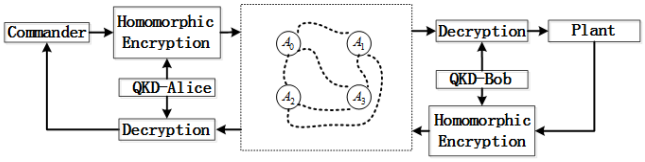

Fig. 2. Networked control systems encrypted with QKD and homomorphic encryption.

Quantum key distribution is a newly developed method of sharing keys using a photon generator and receiver. Alice can generate single photons and transmit them to Bob through fiber. Both Alice and Bob will measure the phase of single photons with two groups of independently chosen bases to get the raw keys. While the errors of measurement will lead to wrong bits in the raw keys, then a protocol call Cascade was used to find the positions of error bits and correct them by sharing the values of check bits via ethernet. The process of error correction will leakage some information about raw keys. Then Alice and Bob will decrease the volume of keys using a Hash function in privacy amplification to enhance the security of keys [11].

It can be seen that quantum key distribution can avoid the transmission of keys through ethernet and generate keys directly in two distributed locations. While at the same time, it means that in a multi agent network, wherever the data need to be encrypted or decrypted, a quantum keys generator is needed. Quantum keys are of high privacy and the keys are independent from each other because of the randomness of phases of single photons.

One-time pad means in an encrypted system a single key should be used only once. Then Eve cannot get meaningful information about plain text via cipher text. Its security can be evaluated by conditional probability as follows:

$$P[M = m_0 | C = c] = P[M = m_0]. \tag{6}$$

It has been proved by using Bias formula [12].

In the above network model, the Homomorphic Encryption is potentially insecure under quantum computation. The only way to make sure privacy security of homomorphic calculation is by making use of symmetric method and one-time pad, the only known method proved secured under unlimited calculation power.

Considering the potential attack by quantum computation, we assume that the publication of $p \times q$ is equivalent to the publication of p and q. However, if we want to share p and q symmetrically, then every node that p and q are used should be equipped with quantum keys generators. Other agents such as commander and controllers used to finish homomorphic calculation without decryption cannot function as before and should also be equipped with QKD to refresh p and q. This method is so high cost and two many quantum keys generating nodes will increase the communication load of network.

As a result of tradeoff, to make sure the homomorphic calculation and control confidentially at the same time, p and q should not be encrypted with symmetric encryption method or quantum keys. However, we can make sure the privacy of parameter g symmetrically with quantum key distribution and one-time pad, which is only used in homomorphic encryption and decryption. The parameters in homomorphic encryption can be designed as follows:

- p and q are two public prime numbers, $n = pq$, and $\lambda = \frac{(p-1)(q-1)}{\mathrm{gcd}((p-1),(q-1))}$.
- g is encrypted with symmetric encryption between Alice and Bob. That means Alice and Bob can share the value of g.
- The keys used for the encryption of g are produce by quantum key distribution and a single key will only be used only once. So the security of g can be well ensured by QKD and one-time pad.
- The encryption and decryption are not changed as (4)(5).

IV. Performance of Networked Control System Encrypted with Homomorphic Encryption

In our network, the plain texts consisting of integers and decimals are operated as combination of long integers to finish encryption by making an appointment in advance on the exact length of integers and decimals. For example, the data 3.1415926 is transferred to be $m = 31415926$ for encryption. As illustrated in (4), the cipher text of

homomorphic encryption is the function of p and q, and it satisfies that:

$$c < (p \times q)^2. \tag{7}$$

So the length of the cipher text is finite and limited by the value of p and q. Further more, p and q should be large enough to make sure that $p \times q$ is longer than plain text. We choose parameters as: $p = 20132659199$, $q = 50000500001$.

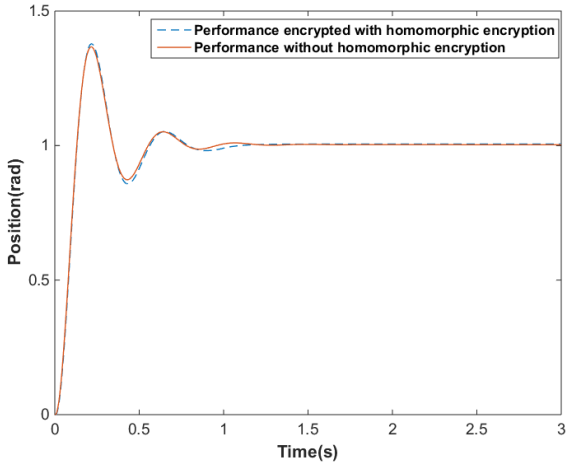

Fig. 3. Comparison of performance with or without homomorphic encryption.

It can be seen that there is small difference between the performance with or without homomorphic encryption as shown in Fig.3.

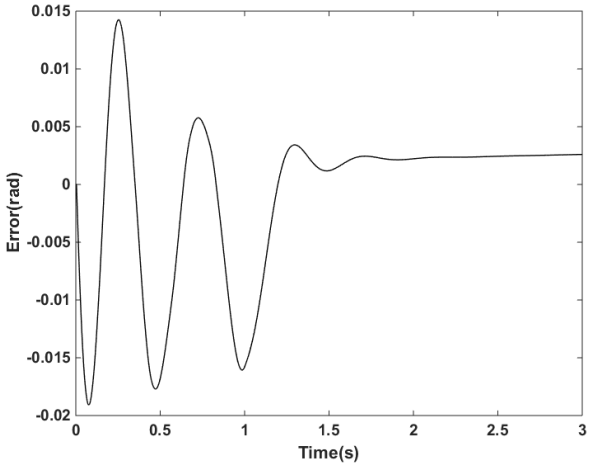

Fig. 4. Control error caused by homomorphic encryption.

The small difference is caused by the error of homomorphic encryption shown in Fig.4. The error of encryption is a tradeoff between the length of p, q and the data volume need to be encrypted. The product $p \times q$ will determine the maximum length of data once it can be encrypted in a single period. In multi-agent networked control systems, usually there are several plants need to be controlled simultaneously, while the length of prime number p and q is always finite. Also, when using predictive control to enhance the real-time performance of networked control systems [13] [14], the data of several periods may be packaged encrypted. So the lower parts of commands or feedback signals have to be ignored to make sure the privacy security encrypted with homomorphic encryption.

When the data volume need to be encrypted is determined, the accuracy of controlling will be influenced by the choosing of parameter p and q. The rounding error of plain text will be larger with small p and q, which may degrade the performance of NCS. The error of homomorphic encryption applied in PI controller is shown in Fig.5:

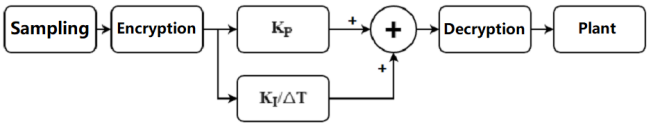

Fig. 5. The error of command in homomorphic encryption.

The accuracy of cipher text is a combination of reference command and feedback signal and we assume that the length of decimal part of plain text is m, after homomorphic encryption will truncate to n. When the data volume need to encrypted is big and p, q are not large enough and cover the whole plain text, it means $n < m$, then the round-off error of encryption will be:

$$\Delta_1 \leq \frac{1}{2} \times 10^{-n}. \tag{8}$$

After homomorphic calculation, the error will be:

$$\Delta_2 \leq (K_p + \frac{K_I}{\Delta T})\Delta_1 \leq \frac{1}{2}(K_p + \frac{K_I}{\Delta T}) \times 10^{-n}. \tag{9}$$

It can be seen that the round-off error of homomorphic encryption will enlarge the system error to be:

$$\Delta \leq \Delta_2 \leq \frac{1}{2}(K_p + \frac{K_I}{\Delta T}) \times 10^{-n}. \tag{10}$$

It means that the error of homomorphic encryption is influenced by the length of public keys assigned to each message and that will further influence the performance of network by declining the accuracy of control especially when the data volume of network is large.

V. CONCLUSION

This paper introduced QKD to homomorphic encrypted networked control systems, which can still be secure where RSA is insecure under quantum computation. Based on the multi-agent network, we applied quantum key distribution and one-time pad to make sure the privacy security of homomorphic encryption by sharing the original private key g with symmetric method. So we can accomplish homomorphic calculation safely only with cipher text in a multi-agent distributed network. The error analysis shows that homomorphic encryption has only small influence on performance when the public key is long enough while the security is dramatically enhanced. Here we considered only

a simple distributed leader-follower network. Based on this, we will go on with some further research on more complex networks.

REFERENCES

[1] C. Rieger, Q. Zhu, and T. Basar, "Agent-based cyber control strategy design for resilient control systems: Concepts, architecture and methodologies," in *International Symposium on Resilient Control Systems*, 2012, pp. 40–47.

[2] M. S. Rahman, M. A. Mahmud, A. M. T. Oo, and H. R. Pota, "Multi-agent approach for enhancing security of protection schemes in cyber-physical energy systems," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 436–447, 2017.

[3] J. Liu and T. Baar, "Toward optimal network topology design for fast and secure distributed computation," 2014.

[4] K. Kogiso and T. Fujita, "Cyber-security enhancement of networked control systems using homomorphic encryption," in *2015 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 6836–6843.

[5] J. Kim, C. Lee, H. Shim, J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Encrypting controller using fully homomorphic encryption for security of cyber-physical systems *," *Ifac Papersonline*, vol. 49, no. 22, pp. 175–180, 2016.

[6] F. Farokhi, I. Shames, and N. Batterham, "Secure and private control using semi-homomorphic encryption ," *Control Engineering Practice*, vol. 67, pp. 13–20, 2017.

[7] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," *Proceedings of Annual Symposium on the Foundations of Computer Science IEEE Computer Society Press Los Alamitos Ca*, pp. 124–134, 1994.

[8] B. Liu, T. Chu, L. Wang, and G. Xie, "Controllability of a leader-follower dynamic network with switching topology," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 1009–1013, 2008.

[9] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 26, no. 1, pp. 96–99, Jan. 1983. [Online]. Available: http://doi.acm.org/10.1145/357980.358017

[10] J. Katz and Y. Lindell, "Introduction to modern cryptography: Principles and protocols," p. 207, 2007.

[11] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *In Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, volume 175, page 8.New York*, 1984.

[12] B. Schneier, "Applied cryptography, second edition : protocols, algorithms,and source code in c," *Government Information Quarterly*, vol. 13, no. 3, p. 336, 1997.

[13] A. Rusnk, M. Fikar, K. Najim, and A. Mszros, "Generalized predictive control based on neural networks," *Neural Processing Letters*, vol. 4, no. 2, pp. 107–112, 1996.

[14] G. P. Liu, J. X. Mu, D. Rees, and S. C. Chai, "Design and stability analysis of networked control systems with random communication time delay using the modified mpc," *International Journal of Control*, vol. 79, no. 4, pp. 288–297, 2006.