PrivacyProtector: Privacy-Protected Patient Data Collection in IoT-Based Healthcare Systems

Entao Luo, Md Zakirul Alam Bhuiyan, Guojun Wang, Md Arafatur Rahman, Jie Wu, and Mohammed Atiquzzaman

Abstract

In IoT-based healthcare, medical devices are more vulnerable to numerous security threats and attacks than other network devices. Current solutions are able to provide protection to patients' data during data transmission to some extent, but cannot prevent some sophisticated threats and attacks such as collusion attacks and data leakage. In this article, we first investigate the challenges with privacy protected data collection. Then we propose a practical framework called PrivacyProtector, patient privacy protected data collection, with the objective of preventing these types of attacks. PrivacyProtector includes the ideas of secret sharing and share repairing (in case of data loss or compromise) for patients' data privacy. Since it is the first time, we apply the Slepian-Wolf-coding-based secret sharing (SW-SSS) in PrivacyProtector. In the framework, we use a distributed database consisting of multiple cloud servers, which ensures that the privacy of patients' personal data can remain protected as long as one of the servers remains uncompromised. We also present a patient access control scheme in which multiple cloud servers collaborate in shared construction to offer patients' data to healthcare providers without revealing the content of the data. The privacy performance analysis has shown that the PrivacyProtector framework is secure and privacy-protected against various attacks.

Introduction

With the capabilities of pervasive surveillance, the Internet of Things (IoT) closely transforms the way we live and work. If it reaches its fullest potential one day, it will basically change every aspect of our lives [1]. IoT is gradually starting to weave into healthcare on both the doctor and patient fronts. The IoT is increasingly becoming the key enabler in the healthcare industry by offering comprehensive improvements in patient engagement, particularly when IoT sensor networks can be used to monitor patients in hospitals and even at home. There is a good set of healthcare applications that have been developed previously, such as MobiCare [2], MEDiSN [3, 4–10].

Although the IoT is the evolution of the Internet to enable many new features to improve patients' everyday lives without interrupting their comfort, these useful features are also examples of security and privacy threats and attacks to the patients' sensitive information that is sent through open wireless channels and the data is stored in back-end servers. Examples of threats include eavesdropping, impersonation, data integrity, data breaches, collusion, and so on. In particular, these threats encompass new challenges, for example, privacy-aware management of patients' personal data and methods to control or avoid pervasive tracking and profiling. Currently, these vulnerabilities are restricting the realization of the IoT healthcare vision when these situations are not dealt with correctly.

In fact, medical sensor network devices in the IoT are found to be more vulnerable to numerous security attacks than other network devices. Current solutions are able to provide protection to patients' data during data transmission to some extent, but may not guarantee that they can prevent some types of attacks well yet, where the administrator of the patient database may disclose sensitive physiological patients' data. To protect patients' data in IoT medical sensor networks against various security threats and attacks, many solutions have been developed. These include secret keys for encryption and authentication, message authentication codes (MACs), the public-key cryptosystem, *k*-anonymity, and so on. Often, current solutions may offer protection to patients' data privacy over transmissions to some extent, but may not prevent certain types of sophistication once a cloud server is compromised; even if a cloud server is under attack by insider attacks, such solutions may not be secure.

In this article, we attempt to overcome some of these challenges to a great extent with privacy protected data collection. We first investigate the challenges with privacy protected data collection. Then we propose a practical framework called PrivacyProtector, novel patients' privacy protected data collection with the objective of preventing threats and attacks. PrivacyProtector includes a new idea of secret sharing and share repairing (in case of data loss or compromise) for patients' data privacy. Since it is the first time, we apply Slepian-Wolf-coding-based secret sharing (SW-SSS) in PrivacyProtector. We consider

In IoT-based healthcare, medical devices are more vulnerable to numerous security threats and attacks than other network devices. Current solutions are able to provide protection to patients' data during data transmission to some extent, but cannot prevent some sophisticated threats and attacks such as collusion attacks and data leakage.

Entao Luo is with Hunan University of Science and Engineering; Md Zakirul Alam Bhuiyan is with Fordham University and Guangzhou University; Guojun Wang is with Guangzhou University; Md Arafatur Rahman is with University Malaysia Pahang; Jie Wu is with Temple University; Muhammed Atiquzzaman is with the University of Oklahoma.

Digital Object Identifier: 10.1109/MCOM.2018.1700364

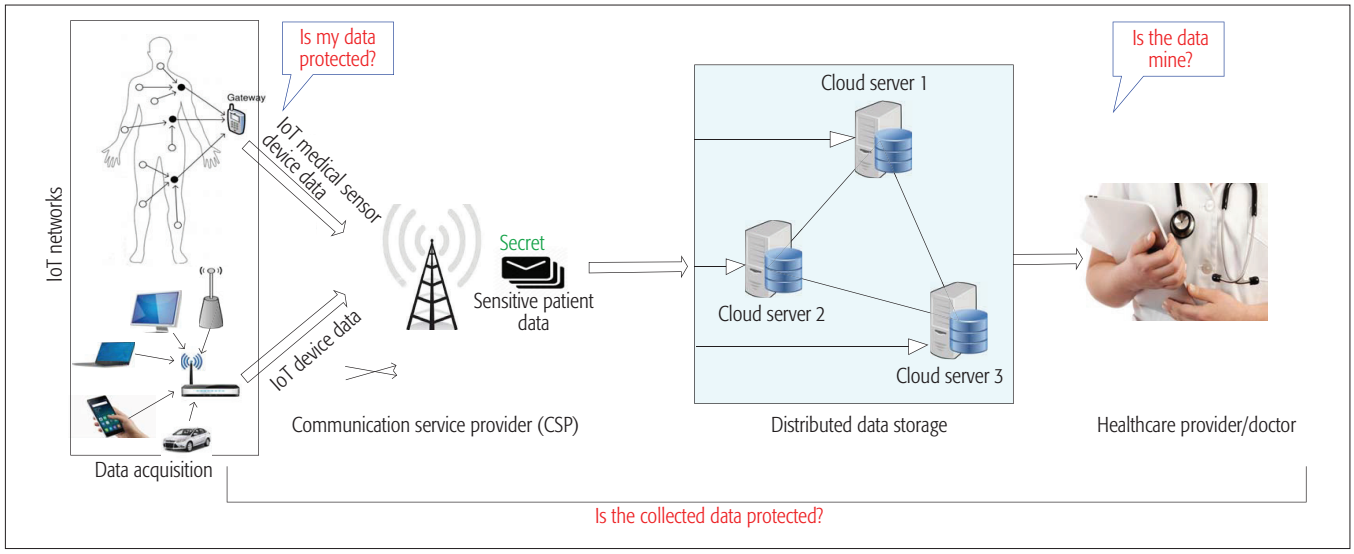


Figure 1. The PrivacyProtector framework.

using multiple cloud servers, which ensures that the privacy of patients' personal data can still be protected as long as one of the servers remains uncompromised, to facilitate the accessibility to patients' data for healthcare threats and attacks. These types of attacks are considered in the attack model of PrivacyProtector:

- Patients' data leakage and destruction
- Collusion attacks
- Insider attacks
- The amount of big data handling
- The amount of data storage

In terms of insider attacks, the system administrator of patients' health databases may also disclose sensitive physiological data. Also, traditional solutions may protect patients' data privacy when one cloud server is compromised. When there are more providers, we present a patient access control scheme by which multiple cloud servers can collaborate to offer the data without revealing the content of the data. The privacy performance analysis shows that the PrivacyProtector framework is privacy-protected against various security and privacy attacks.

This article is organized as follows. First, we give the design of the PrivacyProtector framework. Then we propose the secret sharing and share construction method. Next, we present the data access control method. Then we perform the privacy analysis. Finally, we conclude the article.

The Design of PrivacyProtector

In this part, we propose PrivacyProtector, a patients' privacy protected data collection framework. Similar to traditional healthcare applications within IoT networks, PrivacyProtector has four schemes:

- An IoT network that consist of numerous medical sensor devices and other device network devices. The IoT sensor devices sense the patients' bodies to acquire data.
- The medical sensor device then transmits the collected patients' data to a data storage system through the communication service provider (CSP). The CSP is an important factor that needs to prepare secret shares and distribute the secret shares to the cloud servers, which can be a part of a distributed data storage system.
- The storage system has the patients' data forwarded from the medical sensors in the IoT network and offers querying services to various users that include healthcare providers, doctors, and health professionals.
- A patients' data access control (PDAC) system is utilized by medical users (e.g., healthcare providers) in order to get access to the patients' data and monitor the patients' health performance.

In this article, we do not focus on the first two schemes of PrivacyProtector: the privacy in the data acquisition stage and the CSP stage. Rather, we focus on the security and privacy in the communication aspects that cover the third and fourth schemes. In the third scheme, in PrivacyProtector, the patients' data storage is distributed data storage composed of several cloud servers. We assume that any of the cloud servers can be compromised or data at any of the cloud servers can be revealed by attackers. The PrivacyProtector framework with IoT medical devices, CSPs, distributed storage, and healthcare providers can be seen in Fig. 1. In such a framework, other network devices may interfere with the medical sensor devices under similar IoT networks. They may bring various privacy and security threats to medical devices. Thus, the transmitted data provided by the medical sensor devices is usually unprotected. There can be concern among patients or healthcare users: "is my data protected?," as shown in Fig. 1. Without identifying any alterations of the acquired patients' data, when the data is transmitted toward the upstream cloud server over the CSP, the data cannot be protected. Such data highly impact the overall health monitoring quality and the quality of lives.

Besides, the dominance of IoT medical device sensing is unleashed only by properly collecting unprotected information from different medical sensors whose submitted data may be modified before transmission. Regardless of whether the transmitted data is protected or not, they can be further altered during transmission from the medical to the cloud server by the CSP. Some med-

ical sensors constantly provide protected data, while others may generate biased, compromised, or even fake data due to security attacks such as the collusion attack [1]. There can be another concern of a patient or healthcare user: "Is the collected data protected?," as shown in Fig. 1. Regarding this concern, data collected at the cloud server has to be accompanied by an assessment of the trustworthiness of the data from individual sensor nodes.

Furthermore, there remains a concern: "Is the data mine?," also shown in Fig. 1. Regarding these concerns, data authenticity and confidentiality are obtained using secure signatures and encryption schemes, respectively. To jointly offer confidentiality, integrity, non-repudiation, and authentication, one can use the conventional "signature-then-encryption" strategy, which allows the sender to sign a message prior to encrypting the signed message. However, this approach does not fit the purpose in an IoT medical big data environment. Therefore, the received data should be "protected" before processing. Thus, it is crucial to distinguish whether or not the received data is protected before processing it at a cloud. We use a scheme of secret share and share repair if data is altered or corrupted.

Based on the discussion above, protected data collection requires both outside and inside attacks. For these we consider two kinds of protection: key cryptosystem and secret sharing. Outside attackers are not able to learn a secret key in the system, but try to get the patients' data from different perspectives of the framework, alter or collude the patients' data, or impersonate an IoT medical sensor. Inside attackers are colluders or suspicious cloud servers who somehow know some secret keys in the framework and attempt to learn patients' personal data. However, this becomes difficult for the attackers after having the two types of protection.

Secret Share and Exact Share Repair

The General Idea

To protect against potential threats and attacks, the secret sharing scheme is an ideal method that has been used more popularly in distributed systems. The secret sharing scheme is used for distributing a secret among a group of cloud servers with the help of a CSP. Each cloud server holds a share of the secret. Such a secret needs to be reconstructed. When enough secret shares are combined, they can be constructed. Each share cannot be used alone to extract meaningful information. A case of the secret sharing scheme is depicted in Fig. 1.

In the CSP, the data collected from patients increases dramatically. Storing, managing, and backing up patients' data are troublesome tasks for any CSP when the data are big. Therefore, they tend to use cloud or distributed storage systems to store such patients' data. To preserve the privacy of the sensitive data, a secret sharing scheme is promising. The secret data is encoded and distributed to a set of participants in such a manner that it can only be reconstructed from authorized subsets of cloud servers, or healthcare providers.

The general idea of a secret sharing scheme is that a secret is encoded into a number of shares. Each cloud server receives one share. It can be like the (n, m)-threshold secret sharing in which any m or more shares are utilized to reconstruct the secret and in which the secret share size is the same as the size of the secret. To improve the share size in Shamir's and Blakeley's secret sharing schemes, the Ramp secret sharing was proposed in [2, 7, 11], in which the size of a share is $1/m$ the size of the secret where m denotes the number of blocks in the secret S. However, the drawback of these schemes is the heavy computation cost because the shares are constructed using polynomials such as Reed-Solomon code. To compute the shares, it takes $O(n\log n)$ field operations. To construct the secret, it takes $O(m2)$ field operations. To reduce the computation cost, some secret sharing schemes were proposed using XOR operations instead of polynomials.

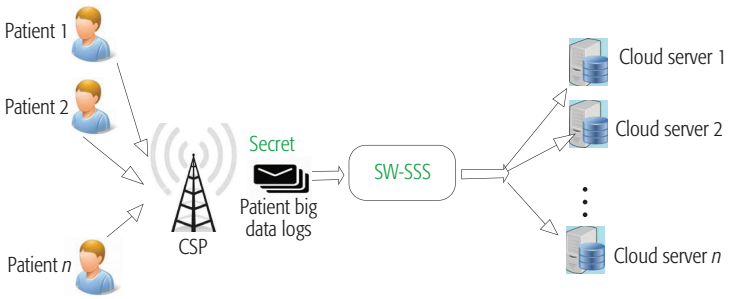
Although the schemes achieve better computation cost, they may not arrange a direct share repair property. This is to say, when a share is compromised due to an unknown reason or compromised due to failure of a cryptosystem, without direct share repair, the CSP must reconstruct the secret S first, and then generate a new share and send it to the participant that has the corrupted or compromised share. If direct share repair is arranged, the compromised share is repaired directly from the remaining healthy shares without the need to reconstruct S. To enable this property, XOR network coding can be applied to secret sharing schemes [1, 12]. Note that there is also another coding, called *error correcting coding* or *erasure coding*, that can be applied in secret sharing to enable share repairs, but it is less efficient than network coding due to the high computation during the repair process and the inability to maximize information flow as network coding. Furthermore, there are also some other types of network coding that can be applied in secret sharing such as *linear network coding*, but we only focus on the *XOR network coding* [8] due to its efficient computation cost. Unfortunately, the following two problems always exist in most XOR network coding-based secret sharing schemes. First, the size of a share is increased; instead of $|S|/m$ as in Ramp secret sharing, the size of a share is now $(|S|/m + m)$, which is the result of enabling direct share repair. Here, there is a trade-off between share size and direct share repair. Second, the newly repaired share is not the same as the original corrupted share.

New Secret Share and Share Repair

Under PrivacyProtector, we propose a new secret sharing scheme using Slepian-Wolf coding (SWC) [13]. The scheme can achieve an optimal share size utilizing the simple binning idea of the coding. There are several approaches in SWC such as syndrome-based, binning-based, LDPC-based, and parity-based [14, 15].

It also enhances the exact-share repair feature whereby the shares remain consistent even if they are compromised or corrupted. The efficiency can be enhanced by reducing the share size, storage, and communication costs. The robustness can be enhanced by supporting the exact-share repair feature in which when a share is corrupted or compromised, a new share is generated that is

The secret sharing scheme is used for distributing a secret among a group of cloud servers with the help of a CSP. Each cloud server holds a share of the secret. Such a secret needs to be reconstructed. When there are enough secret shares combining together, they can be constructed. Each share cannot be used alone to extract meaningful information.



Each day a large number of patients are involved in healthcare-related activities. A huge volume of data with different varieties and velocities can be summarized at the CSP. This data is considered big data. The CSP needs to regenerate a large number of secret shares and forward the shares to the distributed database. It uses the SW-SSS scheme for the forwarding. Each cloud server receives a share, and all the shares are gathered together when a healthcare provider wishes to view them with security access controls.

Figure 2. An application of PrivacyProtector in a case of patients' big data.

exactly the same as the original share. The share size of PrivacyProtector is greatly reduced compared to that of the XOR network *coding-based secret sharing scheme* while still keeping all the benefits:

- The secret shares are constructed using XOR for fast computation.
- Direct share repair is arranged.

The idea is to use another coding, SWC. This is being used in data compression in the network research field.

In PrivacyProtector, we use *network coding*, which has been widely accepted as a unique technique to obtain XOR-based operation and efficient direct share repairs in secret sharing scheme literature, and can be replaced in SWC by carefully designing the protocol to achieve better features. Furthermore, the share size reduction can result in reducing the communication cost between the CSP and cloud servers. The storage cost for each cloud server is also reduced due to a shorter share size.

The exact-share repair feature is supported unlike any previous network-coding-based secret sharing scheme. A corrupted share can be repaired in exactly the same way as its original share. This exact-share repair can make the scheme consistent with the beginning state. Thus, our scheme is stateless, which is more appropriate than stateful where operational costs for state management are required.

Applying SWC when constructing our secret sharing scheme is not straightforward. First, in the concept of SWC, one source block is compressed into one encoded block. However, in the concept of the secret sharing scheme or in any distributed storage system, a secret consists of multiple blocks. The challenge is that we cannot just simply apply SWC on each individual secret block because the shares (coded blocks) are independent; thus, there is no way to repair a corrupted share (coded block) using other unrelated healthy shares (coded blocks). The solution in PrivacyProtector is to construct an XOR between a set of secret blocks, then apply SWC on the XOR instead of each secret block itself. The shares are finally constructed from the correlated XORs, and thus share repair is possible. Second, in SWC, a coded block always gets along with its side information.

The challenge is choosing the side information so that the size distributed in total (both the main coded block and its side information) is efficient. In PrivacyProtector, the side information is the number of 1 bits of the main coded block that can be inferred from the block itself. Instead of using an augmented vector in each share (as in network coding), we manage this scheme such that the indices of the secret blocks can be inferred from the share without depending on such augmented vectors.

SW-SSS Scheme

PrivacyProtector is the first work that applies SWC in a secret sharing scheme. We name this scheme SW-SSS. SW-SSS helps to reduce the share size and achieves exact-share repair property. The scheme is more efficient in terms of storage, communication, and computation costs.

An example of our application in patients' big data is given in Fig. 2, where we consider a CSP that stores all the patients' data access logs every day. The logs are sensitive to the patients' big data containing all the patients' IDs, names, and health data information. The CSP can apply this scheme to mitigate a lot of burdens on the patients' web log storage, management, and communication while being able to preserve its privacy.

In this scheme, a share is not constructed as in the XOR network secret sharing scheme, which is coding-based. Instead, the share becomes the index of the bin to which the XOR corresponds. This is the general idea when applying SWC in the SW-SSS scheme. Note that SW-SSS concentrates on:

- Share generation
- Secret reconstruction
- Share repair algorithms

The share repair is executed when a share is corrupted or compromised. Checking the corrupted or compromised share is beyond the scope of this article; however, several countermeasures can be used to deal with this problem such as homomorphic signature [14].

Patient Data Access Control System

We need to provide the patient data access control (PDAC) security for accessing collected patients' data. PDAC functions with the data stored in distributed cloud servers. We think that an authorized healthcare provider is only able to have access to patients' data. Patients' data must not be exposed outside (e.g., any other data server) during data access. The main procedure of PDAC can be seen in Fig. 3. In the initialization stage, any healthcare provider is able to have access to patients' personal data. In this stage, a healthcare provider produces a private and public key pair for an ID-based signcryption scheme. It also produces a signature verification scheme and a signing key pair method for the digital signature standard (DSS) scheme. The key generation process is shown in Fig. 3. We consider that there is a private and public key infrastructure that is outsourced to an untrusted cloud server and that there is a certificate authority (CA). This authority can certify the public keys for the healthcare provider in a digital certificate. Also, according to SW-SSS, the healthcare provider establishes *n*

(= 1, 2, ...) secure channels with n cloud servers, respectively.

The healthcare provider sends a request that includes a patient's identity (ID, the data attribute, the signature of the healthcare provider on the query, and the certificate of the user to the n cloud servers through the n secure channels. We utilize the secure channels for the healthcare provider to send queries. The reason for this is that the patient's data in the transmitted queries require protection against various outside security attackers. Then the outside attackers must be prohibited by methods including encryption, authentication, and data access control. If the healthcare provider's request passes through the signature verification and satisfies the access control policies, the n cloud servers seek the secret shares of the data, according to the patient's identity and the attributes of the data.

As described, we use signcryption, which is a promising scheme for simultaneously obtaining patients' personal data confidentiality and authenticity. However, basic signcryption has shortcomings, including the limitation of not being able to revoke a healthcare provider from a large-scale system efficiently. We then apply an ID-based signcryption scheme. ID-based signcryption has efficient revocation as well as the advantage of outsourcing the unsigncryption (as shown in Fig. 3) in order to facilitate secure patients' personal data exchanges between the healthcare providers and cloud server(s). ID-based signcryption facilitates a shorter ciphertext size and faster signcryption compared to the sign-then-encrypt approach. Furthermore, the key-update overhead at the private key increases logarithmically with the number of healthcare providers, which is particularly attractive in large-scale patients' big data environments. It also achieves a short-term key exposure resistance against the chosen ciphertext attacks and chosen message text attacks. ID-based signcryption is designed to ensure end-to-end confidentiality, authentication, non-repudiation, and integrity simultaneously, while offering revocation functionality.

Privacy Protection Analysis

In the case of the data collection of PrivacyProtector, the sensor devices in the IoT split the patients' data into several numbers, say n numbers, and then transmit the data to n cloud servers using some secure channels. We think that $n - 1$ secret shares are produced through the SHA-3 algorithm. It needs a secret key and an initial vector (IV), as shown in Fig. 4.

The IV can be a random number so that it can be considered jointly with the counter using any lossless operation (XOR) to produce the actual unique counter-block for encryption. However, we combine it with the key to prevent inside attackers. The reason for this is that adding or XORing straightforwardly to the IV and countering into a single value may break the security under a chosen-plaintext attack, since inside attackers may be able to employ the entire IV-counter pair to cause a collision. Once an attacker takes control of the IV-counter pair and plaintext, the XOR operation of the ciphertext with a known plaintext may result in a value that, when XOR operations made with the ciphertext of the other block sharing the same IV-counter pair, may decrypt that block.

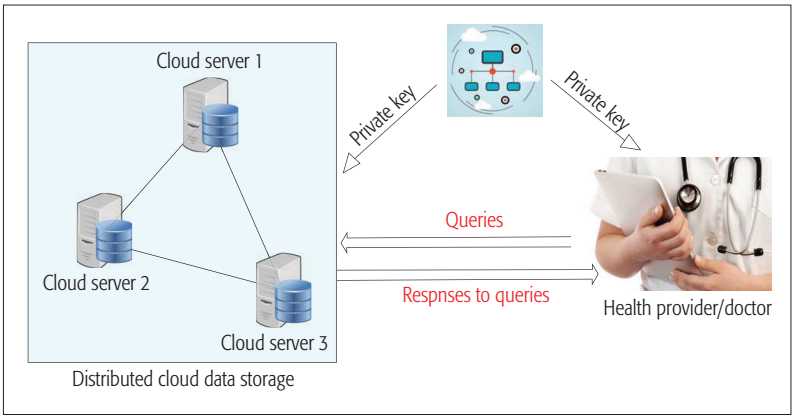


Figure 3. Key generation for a healthcare provider for patients' data access.

In the Key-IV pair, the key is pre-deployed and is known to the medical sensor devices only in the IoT network. That is, any security attacker, including inside attackers such as each cloud server, is not able to predict the IV's random numbers without using the secret key. As long as at least one cloud server remains uncompromised by an inside attacker, no one is able to disclose patients' personal data at data collection and access. In the case of the PDAC scheme, patients' data needs to be encrypted by a public key of a healthcare provider. Without having the private key of the healthcare provider, even in a case where $n - 1$ cloud servers are compromised by the inside attackers, these attackers are never able to get access to the patients' data. Besides, the random numbers in IV are also produced with the SHA-3 algorithm, as shown in Fig. 4.

Conclusions

In this article, we have investigated the privacy protected data collection and access in IoT-based healthcare applications and proposed a new framework called PrivacyProtector to preserve the privacy of patients' personal data. Then we have presented a secret sharing scheme named SW-SSS in order to optimize the secret share size and to support exact-share repairs while still keeping the advantages of the previous scheme. The scheme devises the patients' data and stores it in several cloud servers. If one or two data servers are compromised, the patients' personal data privacy is still protected. For healthcare providers, we present a patient access control scheme, where several cloud servers collaborate to offer patients' data to healthcare providers, but do not reveal the content of the data. The performance analysis shows that the PrivacyProtector framework is protected against various attacks.

References

[1] N. Cai and W. Raymond, "Secure Network Coding," *Proc. IEEE Int'l. Symp. Info. Theory*, 2002, pp. 1–8.
[2] A. Aragues *et al.*, "Trends and Challenges of the Emerging Technologies toward Interoperability and Standardization in E-Health Communications," *IEEE Commun. Mag.*, vol. 49, no. 11, Nov. 2011, pp. 182–88.
[3] J. Ko *et al.*, "Medisn: Medical Emergency Detection in Sensor Networks," *ACM Trans. Embedded Comp. Sys*, vol. 10, no. 1, 2010, pp. 1–29.
[4] X. Yi *et al.*, "Privacy Protection for Wireless Medical Sensor Data," *IEEE Commun. Mag.*, vol. 13, no. 3, Mar. 2016, pp. 369–83.

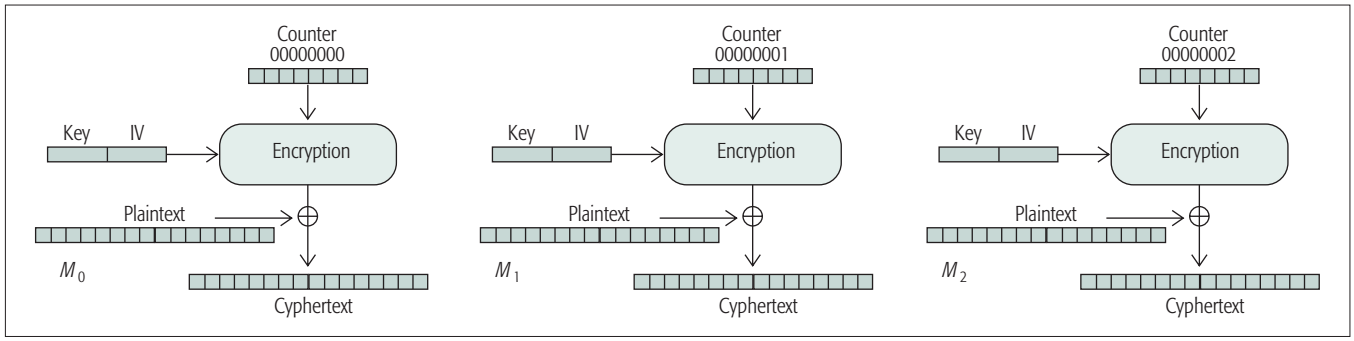


Figure 4. Key-IV pair for better security.

[5] P. Belsis and G. Pantziou, "A *k*-Anonymity Privacy-Preserving Approach in Wireless Medical Monitoring Environments," *J. Personal Ubiquitous Comp.*, vol. 18, no. 1, 2014, p. 6174.

[6] M. A. Chowdhury, W. Mciver, and J. Light, "Data Association in Remote Health Monitoring Systems," *IEEE Commun. Mag.*, vol. 50, no. 6, June 2012, pp. 369–83.

[7] J. Li *et al.*, "Secure Deduplication with Effcient and Reliable Convergent Key Management," *IEEE Trans. Parallel Distrib. Sys.*, vol. 25, no. 6, 2014, pp. 1615–25.

[8] Y. Wang, "Privacy-Preserving Data Storage in Cloud Using Array BP-XOR Codes," *IEEE Trans. Cloud Comp.*, vol. 3, no. 4, 2015, pp. 425–35.

[9] J. Son *et al.*, "Privacy-Preserving Electrocardiogram Monitoring for Intelligent Arrhythmia Detection," *Sensors*, vol. 17, no. 6, 2017, pp. 31–22.

[10] M. Z. A. Bhuiyan *et al.*, "Deploying Wireless Sensor Networks with Fault-Tolerance for Structural Health Monitoring," *IEEE Trans. Computers*, vol. 64, no. 2, 2015, pp. 382–95.

[11] O. Farras *et al.*, "Channel Simulation and Coded Source Compression," *34th Cryptology Conf. Advances in Cryptology*, vol. 62, no. 11, 2014, pp. 217–34.

[12] A. Kalantari *et al.*, "Secrecy Analysis on Network Coding in Bidirectional Multibeam Satellite Communications," *IEEE Trans. Info. Forensics Security*, vol. 10, no. 9, 2015, pp. 1862–74.

[13] D. Slepian and J. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Trans. Info. Theory*, vol. 19, no. 4, 1973, pp. 471–80.

[14] M. Hsieh and S. Watanabe, "Channel Simulation and Coded Source Compression," *IEEE Trans. Info. Theory*, vol. 62, no. 11, 2016, pp. 1–8.

[15] M. Hayashi and R. Matsumoto, "Secure Multiplex Coding with Dependent and Non-Uniform Multiple Messages," *IEEE Trans. Info. Theory*, vol. 19, no. 4, 2016, pp. 2355–2409.

Biographies

Entao Luo is an associate professor in the School of Electronics and Information Engineering, Hunan University of Science and Engineering, China. He worked for Huawei, Shenzhen, China, from 2005 to 2007. His research interests include security and privacy issues in cloud computing and mobile social networks..

Md Zakirul Alam Bhuiyan [M'09, SM'17] is currently an assistant professor in the Department of Computer and Information Sciences at Fordham University. Previously, he worked as an assistant professor at Temple University. His research focuses on dependable cyber physical systems, WSN applications, big data, and cyber security.

Guojun Wang received his B.Sc. in geophysics, M.Sc. in computer science, and Ph.D. in computer science from Central South University, China. He is currently the Pearl River Scholarship Distinguished Professor at Guangzhou University, China. His research interests include cloud computing, trusted computing, and information security.

Md Arafatur Rahman is currently an assistant professor in the Faculty of Computer Systems and Software Engineering, University Malaysia Pahang. He got his Ph.D. degree from the University of Naples Federico II, Italy, and his Master's Degree from the International Islamic University Malaysia. His research interests include wireless communication and cognitive radio networks.

Jie Wu [F] is the Associate Vice Provost for International Affairs at Temple University. He also serves as the director of the Center for Networked Computing and as a Laura H. Carnell Professor. Prior to joining Temple University, he was a program director at the National Science Foundation. His current research interests include mobile computing and wireless networks, cloud computing, network trust and security, and social network applications.

Muhammed Atiquzzaman [M'87, SM'95] is currently an Edith Kinney Gaylord Presidential Professor of Computer Science at the University of Oklahoma. He serves as the Editor-in-Chief of the *Journal of Network and Computer Applications*, the founding Editor-in-Chief of the *Vehicular Communications* journal, and an Associate Editor of several journals, including *IEEE Communications Magazine*. His research interests are in next generation networks, wireless and mobile networks, satellite networks, and more.