Secure Transmission of Compressed Sampling Data Using Edge Clouds

Yushu Zhang, *Member, IEEE*, Ping Wang, Liming Fang, *Member, IEEE*, Xing He, Hao Han, and Bing Chen

***Abstract*—Cloud capability is considered to be extended to the edge of the Internet for improving the security of data transmission. Compressive sensing (CS) has been widely studied as a built-in privacy-preserving layer to provide some cryptographic features while sampling and compressing, including data confidentiality guarantees and data integrity guarantees. Unfortunately, most existing CS-based ciphers are too lightweight or highly complex to meet the requirements of both high security of transmitting the captured data over the Internet and low energy consumption of sensing devices in the Internet of Things (IoT). In this article, a secure transmission framework for CS data by combining CS-based cipher and edge computing is proposed. From the perspective of security, the double-layer encryption mechanism and double-layer authentication mechanism are rooted in it by performing some privacy-preserving operations, including CS-based encryption, CS-based hash, information splitting, strong encryption, and feature extraction. Most significantly, the proposed framework is very useful for resource-limited IoT applications.**

***Index Terms*—Compressive sensing, data security, edge clouds, resource-constrained applications.**

I. Introduction

With the improvement in processing power and storage capability, computing resources have become less expensive, more powerful, and more ubiquitously available than ever before. Such a development trend has created a novel distributed computing model called cloud computing [1], which permits convenient and on-demand network accesses to a shared pool of configurable resources (e.g., networks, storage, and computations) that can be promptly provided and released with minimal management effort. Cloud computing has already evolved as the main computing infrastructure with diversified services, including software as a service, platform as a service, and infrastructure as a service.

The combination of the Internet of Things (IoT), artificial intelligence (AI), and the cloud is creating a smarter world, where billions of terminal devices serving many different scenarios such as environmental monitoring, industrial intelligence, smart grid, and vehicular networks are interconnected to central clouds provided with powerful intelligent processing ability. As a result, the explosive growth of data creates a heavy burden on the information and communication systems, mainly reflected in two aspects: large occupied bandwidth and high network latency. By the end of 2018, nearly 70 billion IoT devices have been connected to the Internet, creating approximately 33 ZB of digital data. Faced with such a large and intricate network, cloud capability is considered to extend to the edge of the Internet, called the edge cloud or fog [2], [3], in which a certain amount of computing and storage resources are placed in close proximity to users such that some data processing operations can be performed in the local servers instead of in the central servers. Accordingly, the quantity of IoT data to be transmitted to the core network will be significantly reduced with the popularization of edge computing infrastructure [4].

In addition to relieving the network load, edge clouds can also be used to guarantee data transmission security. Specifically, some high-complexity privacy-preserving operations, such as strong encryption, are performed in close proximity to terminal devices to guarantee the security of transmitting the acquired data over the Internet. Compressive sensing (CS) [5]–[7], an emerging sampling theory without the limit of the Shannon–Nyquist sampling theorem, can simultaneously perform sampling and compression. By virtue of low-complexity sampling, a CS-based information acquisition system is very suitable for use in resource-constrained IoT applications [8]–[10]. However, there are two noticeable issues hindering its use in practice. On the one hand, very few samples on the encoder side are at the cost of the high computing complexity of the reconstruction algorithm on the decoder side. On the other hand, most existing cryptosystems are too high powered to be embedded in the

Manuscript received September 25, 2019; revised December 2, 2019 and December 26, 2019; accepted January 10, 2020. Date of publication January 14, 2020; date of current version June 22, 2020. This work was supported in part by the National Key R&D Program of China under Grant 2017YFB0802300, in part by the Chongqing Key Laboratory of Mobile Communications Technology under Grant cqupt-mct-201901, in part by the Fundamental Research Funds for the Central Universities under Grant XDJK2020TY003, in part by the Six Talents Peak Project of Jiangsu Province under Grant RJFW-027, and in part by the National Natural Science Foundation of China under Grant 61672283, Grant 61572253, Grant 61702236, and Grant 61972200. Paper no. TII-19-4371. *(Corresponding author: Xing He.)*

Y. Zhang, L. Fang, H. Han, and B. Chen are with the College of Computer Science and Technology, the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: yushu@nuaa.edu.cn; fangliming@nuaa.edu.cn; hhao@nuaa.edu.cn; cb_china@nuaa.edu.cn).

P. Wang and X. He are with the College of Electronics and Information Engineering, Southwest University, Chongqing 400715, China (e-mail: bruce_wp@163.com; hexingdoc@swu.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2020.2966511

physical layer of some resource-limited IoT applications, which is a general contradiction between energy consumption and data security.

In recent years, some pioneering works considered CS as a symmetric cipher [11]–[14] and a robust hash function [15]–[17], but there still exist some noticeable challenges in practice. When working with a one-time sampling model, a CS-based information acquisition system is said to provide a computational but theoretical guarantee of secrecy [11], [18]. Such a security level is obtained under some limited attack scenarios, i.e., an attacker is permitted to work out the secret key by searching the entire keyspace or the original signal from CS samples. In addition, a recent study noted that the statistical properties of the sensed data can leak the energy of the original signal and what can be inferred from its energy [14], owing to the linearity of the CS-based encryption. Therefore, an additional strong but high-energy cryptosystem is still necessary in the existing CS-based privacy-preserving works [19]–[21]. The CS-based hash function is based on the dimensionality-reducing feature and the restricted isometry property[22]–[24] of CS to realize a one-way projection. The acquired additional samples are used for a message authentication code (MAC), which creates a high computation complexity in [15] and [17]. Therefore, it remains to be further studied how to design a low-complexity MAC generation method.

Note that, this article is devoted to the security of data transmitted from the resource-limited IoT devices to the central cloud. A two-layer secure transmission framework for CS data using the edge computing technique is proposed, in which high complexity of the reconstruction algorithm and high complexity of the strong encryption algorithm are transferred to the central cloud and the edge cloud, respectively. Considering that most security threats exist in close proximity to the central cloud, the confidentiality of data from the terminal devices to the local edge servers and from the edge servers to the central servers are provided with computational secrecy by CS-based encryption and information-theoretic secrecy by a novel edge processing scheme, respectively. Moreover, the proposed framework can guarantee the integrity of CS data while tolerating a certain level of noise produced in the process of transmission and reconstruction.

The contributions of this article are summarized as follows:

1) A two-layer secure transmission framework for CS data is first proposed with the help of edge computing.
2) We clearly demonstrate the confidentiality of CS measurements, and then, present an improved version of the CS-based perfect encryption.
3) We present a better CS-based hash algorithm than [17] to guarantee the authenticity of the reconstructed signal.
4) A keyed hash function with modification-localized capability is proposed to improve the robustness of signal reconstruction while resisting tampering attacks.

The rest of this article is organized as follows. In Section II, we briefly introduce some related knowledge about edge computing and CS-based cryptosystems and then highlight some existing challenges in this field. A secure transmission framework for CS data is proposed in Section III. Simulation and security analyses are presented in Section IV. Section V concludes this article.

II. Related Knowledge

In this section, what is cloud computing and edge computing and how to embed confidentiality and integrity in CS are briefly introduced, and then, some existing challenges are identified in this field.

A. From Cloud Computing to Edge Computing

Almost all zoetic or abiotic projects can be projected into the network to realize that all things are interconnected with each other, which decreases the distance between man and nature. In the IoT paradigm, thousands of terminal devices, such as drones, vehicles, smartphones, and cameras, are interconnected over the Internet. Apparently, the quantity of data increases with the increase in IoT devices. To exploit economies of scale and avoid the capital expenditure of building an isolated data center, massive data produced in IoT need to be transmitted to several central clouds for storing and computing in practice, resulting in a heavy network burden reflecting in network congestion, network delay, data loss, etc. However, in the current IoT, there is usually a high demand for real-time response under some safety-critical applications such as self-driving systems and the Internet of drones. For example, on the Internet of self-driving vehicles, when data captured from the vehicle sensors are transmitted to remote data centers, and then, the processed results are sent back to help self-driving, the high network delay makes passengers very dangerous. Accordingly, it is not a good idea that all data produced in IoT are processed and stored in central clouds.

To overcome the abovementioned difficulties, developing a 5G network and a more powerful computing model is currently a technological trend. In addition, cloud capability is being considered to extend to the edge of the Internet, i.e., edge clouds, where a certain number of computing and storage resources are placed in close proximity to terminal devices to reduce the quantity of data in the core network. Hence, a novel computing model is created, as shown in Fig. 1, in which data are processed or preprocessed in the local servers, and then, the processed results are sent to the central servers. Such a computing model efficiently avoids transmitting all data produced in IoT to the core network without distinction, which is exceedingly significant in the era of big data.

B. Embedding Confidentiality and Integrity Guarantee in CS

CS, an emerging sampling technique, can be summarized in a mathematical framework in which a K-spare signal $\mathbf{x}$ is randomly sampled under the sub-Nyquist rate through a linear dimensionality-reducing projection Φ (i.e., the measurement matrix): $\mathbb{R}^N \rightarrow \mathbb{R}^M$ to generate a vector consisting of measurements $\mathbf{y} = \Phi\mathbf{x}$ $(\mathrm{O}(K/\log(N/K)) \leq M \ll N)$ and the original signal $\mathbf{x}$ can be reconstructed from very few measurements $\mathbf{y}$ by searching the optimal solution. In recent years, some pioneering works [11]–[15], [17] suggested that CS can not only sharply

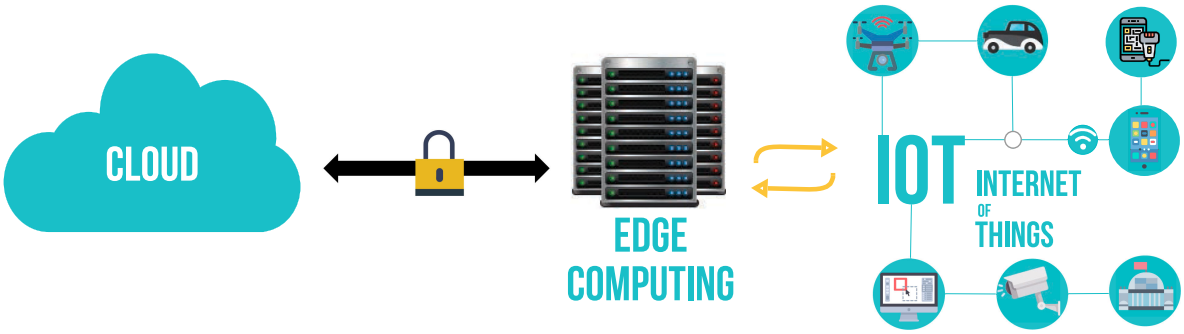


Fig. 1. Edge computing model.

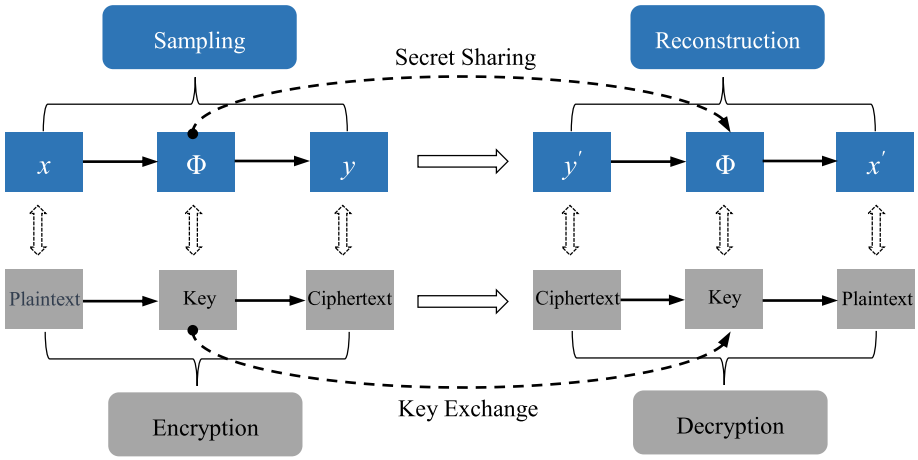


Fig. 2. CS-based symmetric cryptosystem.

reduce the number of samples corresponding to the volume of data collected but also efficiently guarantee the confidentiality and integrity of data. Specifically, the CS-based information acquisition system has the potential to encrypt data and generate hash value while sampling and compressing, which are called CS-based encryption and CS-based hash, respectively.

In view of the reconstruction infeasibility without Φ, the CS-based encryption is based on viewing Φ as a shared secret between the encoder and the decoder. As shown in Fig. 2, the original signal, the recovered signal, the measurement matrix, sampling, and reconstruction can be regarded as the plaintext, the ciphertext, the key, encryption, and decryption, respectively. Note that, ciphertext is not completely identical with plaintext owing to the lossy compression of CS. In the CS-based information acquisition systems, a keyed pseudorandom number generator (PRNG) is usually employed to construct Φ instead of directly transmitting the large-scale Φ.

In addition, the dimensionality-reducing capability of CS can be used as a hash function to generate a noise-resilient MAC, which cannot be done by some traditional hash algorithms, such as MD5, SHA-1, and SHA-3. As shown in Fig. 3, on the encoder side, some additional measurements $\mathbf{y}_{\mathrm{MAC}}$ (i.e., the authentication measurements) are obtained by Φ_{MAC} (i.e., authentication matrix) and, then, an abstract feature of $\mathbf{y}_{\mathrm{MAC}}$ is

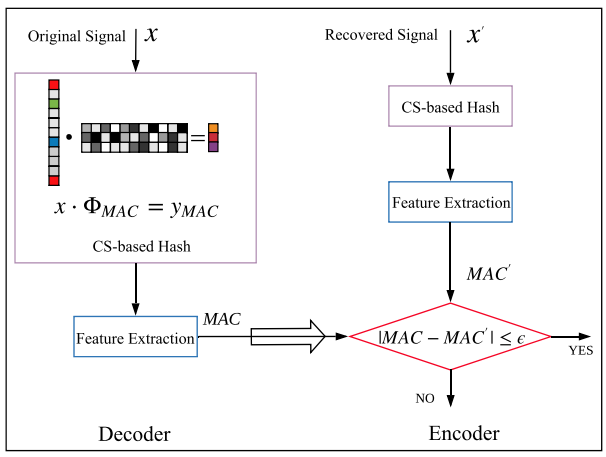


Fig. 3. CS-based integrity authentication scheme.

extracted as the corresponding MAC. In regard to data integrity, a new MAC′ generated from the reconstructed signal $\mathbf{x}'$ is compared with the received MAC. As a result, whether the received data have been tampered with can be determined by checking MAC′ and MAC. For security reasons, Φ_{MAC} and Φ should be as different as possible to avoid the correlation between MAC

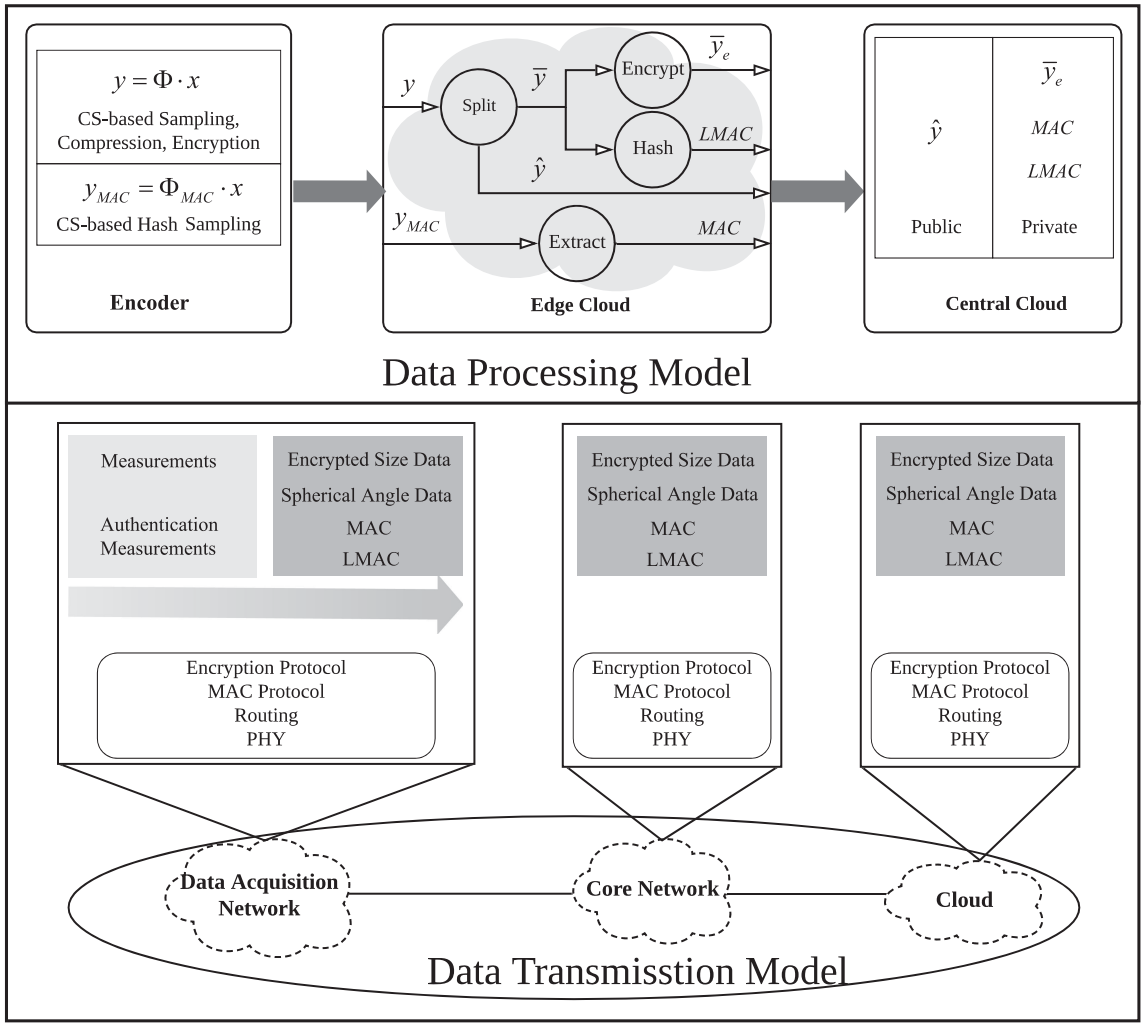


Fig. 4. Secure and efficient data transmission framework.

and $\mathbf{y}$. In addition, the row number of Φ_{MAC} should be less than $\mathcal{O}(K/\log(N/K))$ to guarantee that $\mathbf{y}_{\text{MAC}} = \Phi_{\text{MAC}}\mathbf{x}$ is a one-way projection. In practice, Φ_{MAC} is generated in the same PRNG with CS-based encryption.

Essentially, the best advantage of CS theory is to transfer the computational complexity from the encoder to the decoder. After providing confidentiality and integrity guarantees, the CS-based information acquisition system can perform sampling, compression, encryption, and hash simultaneously with minimal cost, which can, therefore, be viewed as a built-in privacy-preserving layer at most zero cost for resource-limited IoT applications.

C. Existing Challenges

In recent years, many works have attempted to embed privacy-preserving features in CS-based information acquisition systems to simplify sensing devices to the utmost in some resource-constrained IoT scenarios. However, there is no systemic transmission strategy balancing energy consumption and security demand until now.

From the perspective of confidentiality, CS-based data obfuscation is a weak form of encryption. When attackers aim to obtain Φ though brute-force attack on the key management system, the CS-based encryption technique is said to provide a computational guarantee of secrecy, whose strength depends on the size of the keyspace, rather than a theoretical guarantee of secrecy satisfying a statistical independent condition $P(X = \mathbf{x}|Y = \mathbf{y}) = P(X = \mathbf{x})$. Moreover, due to the linearity of dimensionality-reducing projection, attackers are still likely to obtain useful information about the plaintext $\mathbf{x}$, such as its energy and what can be inferred by knowing its energy through ciphertext-only attacks. To achieve a high level of secrecy, existing works are forced to employ a traditional nonlinear cryptosystem to encrypt the captured $\mathbf{x}$ again. Obviously, its high-energy consumption contradicts the core value of the CS-based sampling, compression, and encryption.

Except for confidentiality, how to design a noise-resilient but tamper-evident data integrity authentication scheme based on CS remains to be further studied until now. Kang *et al.* [15] suggested to extract the order of $\mathbf{y}_{\text{MAC}}$ as the MAC of $\mathbf{x}$. However, sorting algorithms consume considerable computing

and storage resources; for example, the time complexity of the merge sort is $\mathcal{O}(n\log_2 n)$, which is a nonnegligible load in a CS-based information acquisition system.

Generally, there is a general contradiction between data security and energy consumption, particularly in resource-constrained scenarios, and most existing nonlinear encryption techniques and hash functions are not applicable for embedding in the physical layer of CS-based sensing applications. How to provide higher security at the cost of lower energy consumption is an important challenge.

III. Secure Transmission Framework for CS Data

As shown in Fig. 4, CS-based encryption and hash operations can be embedded in the process of compressive sampling, data transmission security is improved in the edge cloud instead of on the encoding side, and high-complexity reconstruction tasks are outsourced to the central cloud. Note that the signal reconstruction tasks are performed in the edge cloud when there is a special requirement for real-time responses. Some privacy-preserving postprocessing operations, including information splitting, feature extraction, strong encryption, and keyed hash, are implemented in the edge cloud to improve the security of CS data, which is discussed in detail in the following sections.

A. CS-Based Information Acquisition

On the encoder side, the CS-based information acquisition system can be viewed as a lightweight cryptosystem to guarantee the confidentiality and integrity of sampled data while sampling and compressing. Specifically, the measurement matrix Φ and the authentication matrix Φ_{MAC} are generated from one Gaussian PRNG by inputting two different keys and, then, using them to acquire noise-like measurements $\mathbf{y}$ and authentication measurements $\mathbf{y}_{\text{MAC}}$, respectively. It is worth noting that the key controlling Φ needs to be updated frequently, even in a one-time pad model, such that eavesdroppers cannot determine the secret-shared Φ from enough ciphertext-plaintext pairs. Such a dimensionality-reducing projection can simultaneously realize sampling, compression, encryption, and hash operations in the same hardware infrastructure, meaning that the terminal sensing devices are as low cost as possible. In the absence of additional privacy-preserving operations, the abovementioned CS-based information acquisition system can only provide the sampled data with a certain degree of confidentiality, which is not enough for security-critical applications.

B. Privacy-Preserving Edge Processing

In the edge cloud, cloud computing capability is extended to the edge of the Internet to balance a load between the terminal sensing devices and the central servers. Specifically, there are two kinds of works. In one, a representative feature of $\mathbf{y}_{\text{MAC}}$ is extracted as MAC of $\mathbf{x}$ to avoid directly transmitting large-size $\mathbf{y}_{\text{MAC}}$. The other is that some privacy-preserving operations are performed to enhance the confidentiality of CS data to be set to the core network.

Algorithm 1: A More Efficient And Secure Feature Extraction Algorithm than [17].

Input: Authentication measurements $\mathbf{y}_{\text{MAC}} = [y_{MAC1}, y_{MAC2}, \ldots, y_{MACt}]$.
Output: The robust $MAC = [MAC_1, MAC_2, \ldots, MAC_t]$.

```
Initialize MAC ← [0, 0, ..., 0];
for i ← 1 to t do
    if y_MACi ≤ 0 then
        MACi ← 0;
    else
        MACi ← 1;
    end if
end for
return MAC
```

For the received $\mathbf{y}_{\text{MAC}}$, extracting an abstract feature as the robust MAC, which can not only resist tampering attacks but tolerate a certain level of error resulting from channel noise and reconstruction noise [25], also reduces the quantity of data entering the core network to some extent. Let us define such an extraction function as $\text{MAC} = E(\mathbf{y}_{\text{MAC}})$, which must have the following properties:

1) $E(\mathbf{y}_{\text{MAC}}) \approx E(\mathbf{y}_{\text{MAC}} + \varepsilon)$, if $\|y_{\text{MAC}}\|_2 \gg \|\varepsilon\|_2$;
2) $E(\mathbf{y}_{\text{MAC}}) \neq E(\mathbf{y}_{\text{MAC}} + \varepsilon)$, otherwise.

Obviously, extracting the order of $\mathbf{y}_{\text{MAC}}$ as MAC, as suggested in [17], can satisfy the abovementioned conditions but is always endowed with a high computational complexity. Here, we, therefore, suggest extracting the sign of $\mathbf{y}_{\text{MAC}}$ as a robust MAC, which can be performed by Algorithm 1. Considering that CS measurements often follow a sub-Gaussian distribution whose mean is 0, such a feature extraction algorithm can not only achieve the same performance as [17] but also spend fewer resources to make the information entropy of MAC largest.

According to a new study about the confidentiality of CS measurements [14], the information of $\mathbf{y}$ can be split into two parts, including the energy $\varepsilon = \|\mathbf{y}\|_2^2$ and the spherical angle $\omega = \mathbf{y}/\sqrt{\|\mathbf{y}\|_2^2}$, and only ε leaks the information about the original signal when using an independent identically distributed (i.i.d) Gaussian random matrix, which satisfies the following asymptotic spherical secrecy definition.

Definition 1 (Asymptotic spherical secrecy [12]): Define $\mathbf{x} = [x_1, x_2, x_3, \ldots, x_M]$ as a plaintext sequence and $\mathbf{y}$ as the corresponding ciphertext sequence. Assume that the energy of the plaintext is positive and finite, i.e., $0 < \lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} x_k^1 < +\infty$. Such a cryptosystem is said to achieve asymptotic spherical secrecy if $f_{\mathbf{y}|\mathbf{x}}(x, y) \to_{\mathcal{D}} f_{\mathbf{y}|\varepsilon_{\mathbf{x}}}(y)$, where $\mathcal{D}$ denotes the convergence in the distribution as $n \to \infty$.

Mathematically, ε is a constant, ω is a unit vector, and the measurement vector $\mathbf{y}$ can be recovered by $\mathbf{y} = \varepsilon \cdot \omega$. There is, therefore, a very significant consequence as follows.

Definition 2 (A perfectly secure version of CS-based encryption [14]): CS-based cryptosystem can provide a theoretical guarantee of secrecy if

1) Compressive sampling: $\Phi\mathbf{x} \to \mathbf{y}$, where Φ is a keyed Gaussian random matrix;

2) Information splitting: $\mathbf{y} \rightarrow \varepsilon \cdot \omega$, where ε is a constant and ω is a unit vector;
3) ε and the key controlling Φ are secretly shared between the encoder and the decoder.

The abovementioned results mean that the spherical angle of $\mathbf{y}$ does not reveal anything about $\mathbf{x}$, and the energy of $\mathbf{y}$ is a sufficient statistic for estimating the energy of $\mathbf{x}$ if employing an i.i.d Gaussian random matrix. The abovementioned version suggested in [14] attempted to hide the energy of $\mathbf{y}$ to achieve information-theoretic secrecy by normalizing CS measurements. Unfortunately, such a perfectly secure version creates some practical issues. As mentioned previously, CS measurements follow an approximate Gaussian distribution, so the normalized measurements are composed of an overwhelming number of values that are much less than 1. Hence, a high complexity is rooted in not only this normalization operation but also the process of encoding the fractional part. More importantly, extracting one energy value ε from $\mathbf{y}$ and transmitting ε secretly is extremely unreliable because every elements of the recovered measurements $\mathbf{y}'$ include noise once ε is changed slightly. Considering the abovementioned challenges and the fact that most security threats are located in close proximity to the central servers, we suggest shifting such a privacy-preserving task for CS measurements from the encoder to the edge servers and splitting $\mathbf{y}$ into more pieces to improve the stability. An improved version of CS-based perfect encryption is proposed as follows:

Proposition 1 (An improved version of CS-based perfect encryption): CS-based cryptosystem can provide a theoretical guarantee of secrecy if

1) compressive sampling $\Phi\mathbf{x} \rightarrow \mathbf{y}$ is implemented on the encoder side, where Φ is a keyed Gaussian matrix;
2) information splitting $\mathbf{y} \rightarrow \mathbf{y}_1 + \mathbf{y}_2 + \cdots + \mathbf{y}_n \rightarrow a_1\bar{\mathbf{y}}_1 + a_2\bar{\mathbf{y}}_2 + \cdots + a_n\bar{\mathbf{y}}_n$ is implemented in the edge cloud, where $\{a_i | i = 1, 2, \ldots, n\}$ is a set of constants and $\{\bar{\mathbf{y}}_i | i = 1, 2, \ldots, n\}$ is a set of sparse vectors whose elements are in the interval $[-1, 1]$;
3) $\{a_i | i = 1, 2, \ldots, n\}$ and the key controlling Φ are secretly shared between the encoder and the decoder.

The abovementioned proposition means that $\mathbf{y}$ can also be split into several pieces, one of which is composed of spherical angle information and size information, to achieve information-theoretic-secrecy if $a_i (i = 1, 2, \ldots, n)$ and the key controlling Φ is unknown to attackers. Note that, the core of CS-based perfect encryption is to scale $\mathbf{y}$ down and up at the scaling facts a_i corresponding to information split and fusion, respectively. The scaling fact has no special requirements in value. Similarly, the energy of $\mathbf{y}$ is used as the scaling fact in [14].

This information splitting in the edge cloud is a main contribution in this article, which can be realized as described in Algorithm 2. First, CS measurements are divided into several subsets according to their magnitude, i.e., $y_i \in (-\infty, -1] \cup [1, +\infty), \exists j \in N^*, 10^{j-1} \leq |y_i| < 10^j$. Next, measurements belonging to the same subset are mapped into the interval $(-1, 1)$ at the scale 10^j. Finally, normalized measurements $\bar{\mathbf{y}}$ (i.e., spherical angle data) and size data $\hat{\mathbf{y}}$ are obtained. Note

Algorithm 2: A Privacy-Preserving Edge Processing Algorithm For CS Measurements.

Input: CS measurements $\mathbf{y} = [y_1, y_2, \ldots, y_M]$.
Output: Normalized measurements $\bar{\mathbf{y}} = [\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_M]$ and size data $\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_M]$.

```
Initialize ŷ ← [0, 0, ..., 0];
Compute m ← ⌈log10(max(|y|))⌉;
for i ← 1 to M do
  for j ← 1 to m do
    if 10^(j−1) ≤ |y_i| < 10^j then
      ŷ_i ← j;
      ȳ_i ← y_i/10^j;
    end if
  end for
  ȳ_i ← y_i;
end for
return ȳ, ŷ
```

that, the size information is composed of scale value and its index, for example, $\hat{\mathbf{y}} = [0, 1, 2, 2, 0, 1]$ denotes that $\{y_1, y_4\}$ has no change but $\{y_2, y_6\}$ and $\{y_3, y_4\}$ are scaled down at 10^1 and 10^2 times, respectively. Although $\bar{\mathbf{y}}$ and $\hat{\mathbf{y}}$ have the same dimensions, the data quantity of $\hat{\mathbf{y}}$ consisting of $\{0, 1, \ldots, m\}$ is much less than that of $\bar{\mathbf{y}}$.

Considering that $\bar{\mathbf{y}}$ only contains spherical angle information leaking nothing, $\bar{\mathbf{y}}$ can be directly transmitted over the public channels. However, $\hat{\mathbf{y}}$ containing the size information about $\mathbf{y}$ needs to be secretly shared between the encoder and the decoder. Hence, strong encryption and integrity verification need to be performed in the edge cloud to make $\hat{\mathbf{y}}$ inaccessible to attackers and to determine whether the received $\hat{\mathbf{y}}$ has been tampered with. It is noteworthy that the sectional loss of $\mathbf{y}$ would not make signal reconstruction infeasible, at least theoretically, as long as the number of available measurements is not less than $\mathcal{O}(K \log(N/K))$, which is called the democracy of CS measurements [26]. Accordingly, an ideal integrity authentication scheme for $\hat{\mathbf{y}}$ should be provided with a certain level of robustness. Such a hash algorithm can detect and localize malicious tampering.

A piecewise linear chaotic map (PWLCM) is defined as follows:

$$X_{t+1} = F(X_t, P) = \begin{cases} X_t/P, & 0 \leq X_t \leq P \\ (X_t - P)/(0.5 - P), & P \leq X_t \leq 0.5 \\ (1 - P - X_t)/(0.5 - P), & 0.5 \leq X_t \leq 1 - P \\ (1 - X_t)/P, & 1 - P \leq X_t \leq 1 \end{cases}$$

where $X_t \in [0, 1]$ and $P \in (0, 0.5)$ denote the iteration trajectory value and the current iteration parameter of PWLCM, respectively.

As is shown in Algorithm 3, the size data $\hat{\mathbf{y}}$ are diffused and permuted by using two chaotic sequences, and the initial input parameters $\{X_0, X_0^{'}, P_0, P_0^{'}\}$ are viewed as shared keys between the encoder and the decoder. As shown

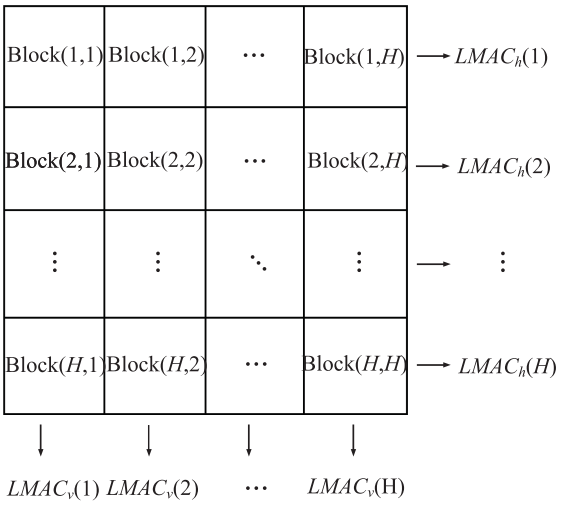


Fig. 5. Hash table.

Algorithm 3: A Bit-Level Strong Encryption Algorithm.

Input: The size data $\hat{\mathbf{y}} = \{\hat{y}_i\}_{i=1}^{M}$ and four initial parameters $X_0, X_0^{'} \in [0,1], P_0, P_0^{'} \in (0,1)$.
Output: Encrypt data $\hat{\mathbf{y}}_e = \{\hat{y}_{ei})\}_{i=1}^{3\ M}$.
1: Initialize $\{\hat{y}_{ei})\}_{i=1}^{3\ M} \leftarrow 0$;
2: Set $X_0, X_0^{'} \in [0,1], P_0, P_0^{'} \in (0,1)$ to iterate PWLCM $M+m$ times, then discard the first m values to obtain two chaotic sequences $\{q_i\}_{i=1}^{M}$, $\{p_i\}_{i=1}^{M}$;
3: Sort $\{q_i\}_{i=1}^{M}$ to obtain $\{q_i^{'}\}_{i=1}^{M}$;
4: Search every value of $\{q_i\}_{i=1}^{M}$ in $\{q_i^{'}\}_{i=1}^{M}$, and then keep the corresponding indexes as $\{ind_i\}_{i=1}^{M}$;
5: **for** $i \leftarrow 1$ to M **do**
6: $\quad \hat{y}_i \leftarrow \hat{y}_i + \lfloor p_i \cdot 10^{13} \rfloor \bmod 8$;
7: $\quad \hat{y}_{e(ind(i))} \leftarrow \hat{y}_i$;
8: **end for**
9: **return** $\hat{\mathbf{y}}_e$

in Algorithm 4, the size data $\hat{\mathbf{y}}$ are first sent into a hash table with $H \times H$ blocks as described in Fig. 5. Second, the size data are input into PWLCM in conjunction with the initial parameters column by column and row by row to obtain a set of MAC used to localize modifications, namely, $\{\text{LMAC}_h(1), \text{LMAC}_h(2), \ldots \text{LMAC}_h(H), \text{LMAC}_v(1), \text{LMAC}_v(2), \ldots \text{LMAC}_v(H)\}$. Finally, they are combined to form the localization message authentication code (LMAC). Comparing the regenerated LMAC with the received LMAC can determine whether there is a tampering attack and where the illegal modifications locate. For example, modification is said to be located in $\text{Block}(2,3)$ if both $\text{LMAC}_h(2)$ and $\text{LMAC}_v(3)$ have been changed, and modification may be located in either or both $\text{Block}(2,1)$ and $\text{Block}(2,2)$ if $\text{LMAC}_h(2)$, $\text{LMAC}_v(1)$, and $\text{LMAC}_v(2)$ are discordant.

C. Data Management in the Central Cloud

In the central cloud, the volume of spherical angle data $\bar{\mathbf{y}}$ is much larger than that of the encrypted size data $\hat{\mathbf{y}}_e$. $\bar{\mathbf{y}}$ that is meaningless for attackers and $\hat{\mathbf{y}}_e$ should, therefore, be stored in the public cloud and the private cloud, respectively. Considering that cloud security [27] is not what concerns us in this article, the security of data management in the central cloud is assumed to be strong enough to resist attackers.

When a legal user asks for the original signal, the corresponding operations are executed sequentially as follows. The integrity of the decrypted size data is first verified. Once there is a tampering attack, the location of modifications can be determined. If the number of remaining real data meets the minimum requirement of CS reconstruction, the decrypted size data are integrated with spherical angle data to obtain $\mathbf{y}$, and then, the reconstructed signal can be generated by searching the optimal solution in the central servers; otherwise, the signal reconstruction fails. After users receive the reconstructed signal, a MAC produced by Algorithm 1 is compared with the received signal to determine whether the reconstructed signal suffered from a tampering attack.

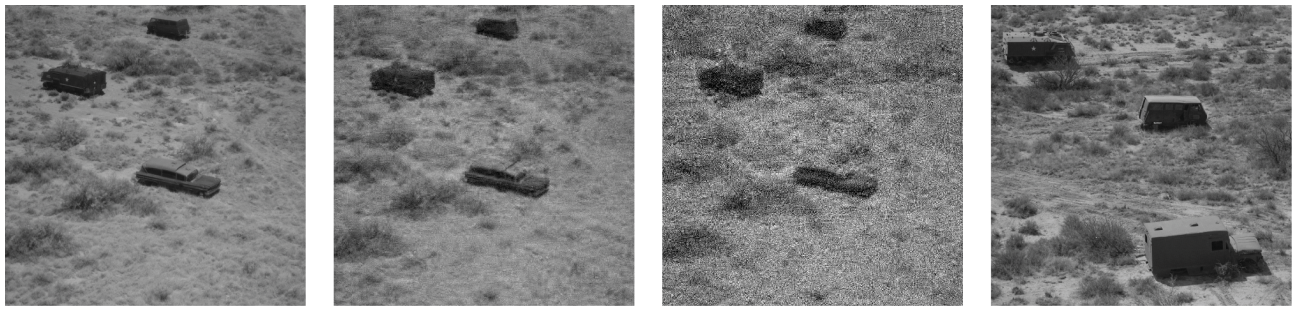
IV. Simulation and Security Analyses

In this section, the security of the proposed framework is further demonstrated by theoretical analyses and simulation experiments. Unless otherwise specified, the elements of the measurement matrix are Gaussian random variables, signal sampling takes place in the time/space domain, the compression ratio is 0.5, and the orthogonal matching pursuit (OMP) reconstruction algorithm is employed in the following experiments.

A. Double-Layer Encryption

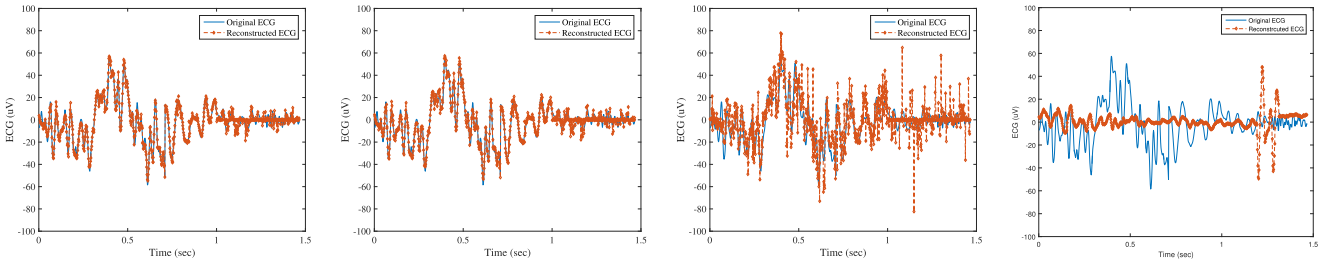
From the perspective of confidentiality, the captured data are synchronously encrypted only by the CS-based information acquisition system in the encoder side, and then, what can be inferred from CS measurements is perfectly hidden by extracting and protecting its sensitive information in the edge cloud. Finally, sensitive information and opening information are stored in the private cloud and in the public cloud, respectively. Obviously, the proposed terminal-to-cloud transmission framework works with a double-layer encryption pattern, which provides a computational guarantee of secrecy for data transmitted from IoT devices to the edge cloud and a theoretical guarantee of secrecy for data transmitted from the edge cloud to the clouds.

CS-based encryption algorithm is a symmetric cipher. The input parameters of the Gaussian PRNG controlling measurement matrix Φ are viewed as keys and need to be frequently updated. When aiming to obtain the real Φ for reconstructing the original signal $\mathbf{x}$ from the illegally intercepted measurements $\mathbf{y}$, eavesdroppers experience considerable difficulty in attacking the key management system or guessing the real key. Generally, both key management systems are sufficiently strong, and the keyspace is sufficiently large to resist the abovementioned attacks. Unfortunately, compressed sampling is a linear projection process. There exists a linear correlation between $\mathbf{x}$ and $\mathbf{y}$. Theoretically, positive attackers can obtain useful information about $\mathbf{x}$ by cryptoanalysis of only $\mathbf{y}$. Hence, the CS-based encryption is a lightweight and built-in privacy-preserving technique that



(a) (b) (c) (d)

Fig. 6. Four different situations. (a) Without any modification. (b) With normal noise. (c) With malicious pollution. (c) With tampered content.



(a) (b) (c) (d)

Fig. 7. Comparison of the original ECG and the reconstructed ECG. (a) Without any modification. (b) With normal noise. (c) With malicious pollution. (c) With tampered content.

can embed a certain level of confidentiality in the captured data at almost zero cost.

It was theoretically proven in [18] that making the size information of CS measurements unavailable for attackers can achieve information-theoretic secrecy using a one-time Gaussian random matrix. What is different from [18] in terms of confidentiality is that the size information is extracted from several components rather than a whole to reduce the computational complexity and improve the data robustness. In addition, we suggest transferring such a nonnegligible computing task from the encoder side to the edge cloud to relieve the burden on the encoder. Mathematically, the improved version of CS-based perfect encryption still satisfies the research conclusions in [18]. Hence, it can be said that the proposed framework can provide the spherical angle data $\bar{\mathbf{y}}$ (i.e., opening data) with, at least in theory, perfect secrecy.

B. Double-Layer Authentication

From the perspective of integrity, the CS-based hash function, which is determined by Algorithm 1, can determine whether the reconstructed signal is the expected one, and the modification-localized hash function, which is determined by Algorithm 4, can resist the tampering attack on size data. Therefore, the proposed framework also works with a double-layer authentication pattern.

For the signal $\mathbf{x}$, there are only two kinds of attack modes. One attack is that negative attackers pollute CS data to cause signal reconstruction to fail in the process of transmission. The other attack is that passive attackers replace the intercepted CS data with false data to change the received information. Note that the second attack is the worst situation in which the attacker has stolen the real measurement matrix, which is hardly possible in practice. To demonstrate the feasibility of the CS-based hash function, a standard gray image with 512×512 pixels and an ECG signal with 750 samples are used in our experiments. In this article, we consider performing CS reconstruction under four different situations as described in Fig. 6, including without any modification, with normal noise, with serious pollution, and with tampered content. Their effects on an ECG signal can be found in Fig. 7. Here, the distortion ratio η of MAC is used as the evaluation index. As shown in Table I, η is usually less than 5% when there is no manipulation or a certain level of normal noise; in addition, two kinds of malicious attacks can be easily detected if the threshold value of differentiation between tampering attacks and noise pollution is set to 5%.

For the sampled data $\hat{\mathbf{y}}$ and $\bar{\mathbf{y}}$, we only consider the potential attacks to $\hat{\mathbf{y}}$ in this article, seeing that $\bar{\mathbf{y}}$ is valueless for attackers. In view of the democracy of CS measurements, an ideal integrity authentication scheme for $\hat{\mathbf{y}}$ could not only detect whether the received data have been tampered with but also find where the modifications locate. Assume that $\hat{\mathbf{y}}$ is an integer sequence with a size of 250 consisting of $\{0, 1, 2, 3, 4\}$, the standard extraction bits is $t = 4$, and the initial chaotic parameters $(X_0, P_0) = (0.45, 0.27)$. As a result, the size of the hash table is 16×16, and the output is a 128-bit LMAC. To evaluate the performance of the proposed hash function with

TABLE I
CS-BASED HASH WHEN 5% IS SET AS THE THRESHOLD OF THE DISTORTION RATIO

Signals	Distortion ratio η	Verification results	Authentication results
Reconstructed image without any modification	1.2%	Pass	Success
Reconstructed image with normal noise	3.9%	Pass	Success
Reconstructed image with malicious pollution	16.2%	No Pass	Success
Reconstructed image with content tampered	14.2%	No Pass	Success
Reconstructed ECG without any modification	2.5%	Pass	Success
Reconstructed ECG with normal noise	4.7%	Pass	Success
Reconstructed ECG with malicious pollution	13.9%	No Pass	Success
Reconstructed ECG with content tampered	18.3%	No Pass	Success

Algorithm 4: A Keyed Hash Algorithm With Modification-Localized Capability.

Input: Size data $\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_M]$ and two initial parameters $X_0 \in [0,1], P_0 \in (0,1)$.
Output: Modification-localized $LMAC$.

Compute $H \leftarrow \lceil \sqrt{M} \rceil$;
Fill a $H \times H$ matrix G with $z_i (i = 1, 2, \ldots, M)$ column by column or row by row, and the remaining positions are used to denote M;
for $i \leftarrow 1$ to H **do**
  for $j \leftarrow 1$ to H **do**
    $G_{ij} \leftarrow 1/1 + e^{-G_{ij}} \in (0,1)$;
    if $j = 1$ **then**
      $P_{ij} \leftarrow (G_{ij} + P_0 + i/H)/6 \in (0, 0.5)$;
      $X_{ij} \leftarrow F(X_0, P_{ij}) \in [0,1]$;
    else
      $P_{ij} \leftarrow (G_{ij} + X_{i(j-1)})/4 \in (0, 0.5)$;
      $X_{ij} \leftarrow F(X_{i(j-1)}, P_{ij}) \in [0,1]$;
    end if
  end for
  Extract t-bits after the decimal point from the binary format of X_{ij} to form a $LMAC_h(i)$;
end for
Joint $LMAC_h(1), LMAC_h(2), \ldots LMAC_h(H)$ to obtain tH-bits $LMAC_h$ in the horizontal direction;
Another $LMAC_v$ in the vertical direction is generated similar to Step 3 $\sim$ 16;
Combine $LMAC_h$ with $LMAC_v$ to obtain $2tH$-bits $LMAC$;
return $LMAC$

modification localization capability, we perform hash simulation experiments in the following conditions:

1) original data: 4032423313400304003100413341303330342113401233033 1322044110412122003103321331010034123242401143023 1044144342134144023131240401122411301324140304 11;
2) change the first value "4" to "3";
3) change the position of the two values located in Block$(2,3)$ and Block$(5,1)$, respectively;
4) change the initial parameter X_0 from 0.450000000 to 0.450000001;
5) change the initial parameter P_0 from 0.270000000 to 0.270000001.

The corresponding LMAC in the hexadecimal format are obtained as follows:

1) 5819 6491 612 A 7E56 D6E7 516B 012D 2529;
2) A819 6491 612 A 7E56 36E7 516B 012D 2529;
3) 5119 E491 612 A 7E56 D657 516B 012D 2529;
4) D1B7 BC19 F4D3 AD0F 6B01 298E B687 D797;
5) E62D F1C6 4672 D819 8154 D35D 1923 B7CF.

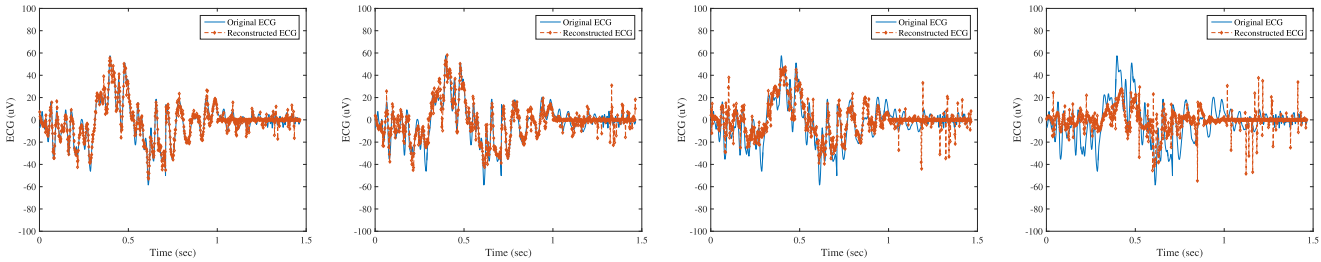
Obviously, a very small modification of the initial chaotic parameters leads to a very distinct difference and changing one number makes the two numbers of the acquired LMAC different from that of the original LMAC. Comparing 1) with 3), even though modifications can only be located in a possible range $\{\text{Block}(2,1), \text{Block}(2,3), \text{Block}(5,1), \text{Block}(5,3)\}$ rather than the accurate positions $\{\text{Block}(2,3), \text{Block}(5,1)\}$, such a localization capability is still significant for CS-based applications. After discarding the distorted data (i.e., the modified size data and the corresponding spherical angle data), the original signal $\mathbf{x}$ can still be successfully reconstructed as long as the number of CS measurements verified is not less than $\mathcal{O}(K \log(N/K))$, as shown in Figs. 7 and 8. Therefore, such a double-layer authentication mechanism in the proposed framework can avoid tampering attacks on both the spherical angle data $\bar{\mathbf{y}}$ and the original signal $\mathbf{x}$.

C. Security Analysis

The proposed framework, which is a two-layer structure, provides a reliable guarantee of security for transmitting CS data from terminal devices to the central cloud. More narrowly, the CS-based built-in privacy-preserving layer in the terminal devices provides measurements with a certain level of confidentiality, which is not enough under high-security requirements. Information splitting and strong encryption operations performed in the edge cloud advance the confidentiality to a higher level, on which the spherical angle data $\bar{\mathbf{y}}$ is said to be perfectly secured and the secrecy of the encrypted size data $\hat{\mathbf{y}}_e$ is related to the chaotic system. In addition, the CS-based hash function guarantees that the original signal (plaintext) cannot be illegally tampered with and that the proposed hash function with modification localization capability is used to identify the tampering behavior and then guarantee CS reconstruction as much as possible by discarding the distorted data.

(a) (b) (c) (d)

Fig. 8. Reconstructed images after discarding a different number of distorted data. (a) 10%. (b) 30%. (c) 50%. (c) 70%.



(a) (b) (c) (d)

Fig. 9. Reconstructed ECG after discarding a different number of distorted data. (a) 10%. (b) 30%. (c) 50%. (c) 70%.

V. Conclusion

In this article, edge computing capability was used to enhance the security of data produced in resource-limited IoT applications, and then, transmitted to the central cloud. Remarkably, the proposed framework possesses a double-layer encryption mechanism and double-layer authentication mechanism, i.e., CS-based encryption, edge postprocessing, CS-based hash, and modification-localized hash.

From the perspective of confidentiality, the CS-based encryption can provide a computational guarantee of secrecy for data transmitted from IoT devices to edge clouds, and the edge postprocessing for CS measurements raises the level of security from computational secrecy to theoretical secrecy. From the perspective of integrity, the CS-based hash can not only tolerate a certain level of noise but also avoid tamper attacks on the spherical angle data, and the modification-localized hash can not only detect tampering behavior but also localize modifications to some extent.

References

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Gener. Comput. Syst.*, vol. 25, no. 6, pp. 599–616, Jun. 2009.

[2] M. Satyanarayanan, "The emergence of edge computing," *Comput.*, vol. 50, no. 1, pp. 30–39, 2017.

[3] W. Shi, C. Jie, Z. Quan, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[4] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput.*, 2012, pp. 13–16.

[5] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[6] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[7] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[8] S. Li, L. D. Xu, and X. Wang, "Compressed sensing signal and data acquisition in wireless sensor networks and Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 9, no. 4, pp. 2177–2186, Nov. 2013.

[9] Y. Zhang *et al.*, "Low-cost and confidentiality-preserving data acquisition for Internet of Multimedia Things," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3442–3451, Oct. 2018.

[10] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed sensing for real-time energy-efficient ECG compression on wireless body sensor nodes," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2456–2466, Sep. 2011.

[11] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *Proc. 46th Annu. Allerton Conf. Commun., Control, Comput.*, 2008, pp. 813–817.

[12] V. Cambareri, M. Mangia, F. Pareschi, R. Rovatti, and G. Setti, "Low-complexity multiclass encryption by compressed sensing," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2183–2195, May 2015.

[13] L. Y. Zhang, K. W. Wong, Y. Zhang, and J. Zhou, "Bi-level protected compressive sampling," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1720–1732, Sep. 2016.

[14] T. Bianchi, V. Bioglio, and E. Magli, "Analysis of one-time random projections for privacy preserving compressed sensing," *IEEE Trans. Inf. Forensics Secur.*, vol. 11, no. 2, pp. 313–327, Feb. 2016.

[15] L. Kang, C. Lu, and Chao-Yung Hsu, "Compressive sensing-based image hashing," in *Proc. 16th IEEE Int. Conf. Image Process.*, 2009, pp. 1285–1288.

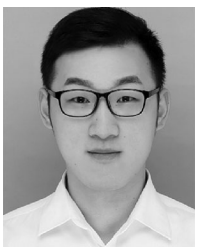
[16] R. Sun and W. Zeng, "Secure and robust image hashing via compressive sensing," *Multimed. Tools Appl.*, vol. 70, no. 3, pp. 1651–1665, 2014.

[17] T. Wu and C. Ruland, "Authenticated compressive sensing imaging," in *Proc. Int. Symp. Netw. Comput. Commun.*, 2017, pp. 1–6.
[18] V. Cambareri, M. Mangia, F. Pareschi, R. Rovatti, and G. Setti, "On known-plaintext attacks to a compressed sensing-based encryption: A quantitative analysis," *IEEE Trans. Inf. Forensics Secur*, vol. 10, no. 10, pp. 2182–2195, Oct. 2015.
[19] R. Huang, K. H. Rhee, and S. Uchida, "A parallel image encryption method based on compressive sensing," *Multimedia Tools Appl.*, vol. 72, no. 1, pp. 71–93, 2014.
[20] N. Zhou, S. Pan, S. Cheng, and Z. Zhou, "Image compression-enncryption scheme based on hyper-chaotic system and 2D compressive sensing," *Opt. Laser Technol.*, vol. 82, pp. 121–133, 2016.
[21] X. Liu, W. Mei, and H. Du, "Simultaneous image compression, fusion and encryption algorithm based on compressive sensing and chaos," *Opt. Commun.*, vol. 366, pp. 22–32, 2016.
[22] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
[23] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.
[24] R. Kueng and D. Gross, "Ripless compressed sensing from anisotropic measurements," *Linear Algebra Appl.*, vol. 441, pp. 110–123, 2014.
[25] A. Orsdemir, H. O. Altun, G. Sharma, and M. F. Bocko, "On the security and robustness of encryption via compressed sensing," in *Proc. IEEE Mil. Commun. Conf.*, 2008, pp. 1–7.
[26] J. N. Laska, P. T. Boufounos, M. A. Davenport, and R. G. Baraniuk, "Democracy in action: quantization, saturation, and compressive sensing," *Appl. Comput. Harmon. Anal.*, vol. 31, no. 3, pp. 429–443, 2011.
[27] K. Yan, W. Shen, Q. Jin, and H. Lu, "Emerging privacy issues and solutions in cyber-enabled sharing services: From multiple perspectives," *IEEE Access*, vol. 7, pp. 26031–26059, 2019.

Yushu Zhang (Member, IEEE) received the Ph.D. degree in computer science from the College of Computer Science, Chongqing University, Chongqing, China, in December 2014.

He held various research positions with the City University of Hong Kong, Southwest University, University of Macau, and Deakin University. He is currently a Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. His research interests include multimedia security, artificial intelligence, cloud computing security, Big Data security, IoT security, and blockchain.

Dr. Zhang is an Editor of Signal Processing.



Ping Wang is currently working toward the master's degree in electronics and communication engineering with the School of Electronics and Information Engineering, Southwest University, Chongqing, China.

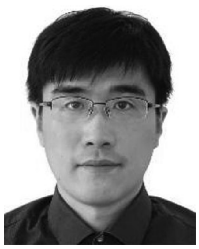
His research interests include secret sharing, compressive sensing security, multimedia security, and artificial intelligence security.

Liming Fang (Member, IEEE) received the Ph.D. degree in computer science from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2012.

He was a Postdoctoral Fellow in the Information Security with City University of Hong Kong. He is currently an Associate Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. He has authored or coauthored more than 50 papers in his field, including IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, *Theoretical Computer Science*, *Designs Codes and Cryptography*, *Information Sciences*, etc. His current research interests include cryptography and information security.

Xing He received the B.S. degree in mathematics and applied mathematics from the Department of Mathematics, Guizhou University, Guiyang, China, in 2009, and the Ph.D. degree in computer science and technology from Chongqing University, Chongqing, China, in 2013.

He is a currently a Professor with the School of Electronics and Information Engineering, Southwest University, Chongqing, China. From November 2012 to October 2013, he was a Research Assistant with the Texas A&M University at Qatar, Doha, Qatar. From December 2015 to February 2016, he was a Senior Research Associate with City University of Hong Kong. His research interests include neural networks, bifurcation theory, optimization method, smart grid, and nonlinear dynamical system.



Hao Han received the B.S. degree in computer science and technology from Nanjing University, Nanjing, China, in 2005, and the Ph.D. degree in computer science from the College of William and Mary, Williamsburg, VA, USA, in 2014.

He is currently a Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. His research interests include system and software security against various types of threats.

Bing Chen received the B.S. and M.S. degrees in computer engineering from Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 1992 and 1995, respectively, and the Ph.D. degree in computer science from the College of Information Science and Technology, NUAA, in 2008.

Since 1998, he has been with NUAA, where he is currently a Professor with the Department of Computer Science and Technology. His main research interests include cloud computing, wireless communications, and cognitive radio networks.