An Efficient and Secure Health Data Propagation Scheme Using Steganography-Based Approach for Electronic Health Networks

Liping Zhang, Wenshuo Han, Shukai Chen, and Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*— Electronic health (e-health) networks enable users to enjoy convenient, flexible, and low-cost medical services at home, so they attract great attention and spread into the market quickly. In e-health networks, large amounts of various health data including personal privacy information and physiological signals are transmitted, which raises security risks. To protect the health data transmitted in e-health networks, steganography-based solutions have been widely researched. Although existing steganography-based solutions successfully hide health data in physiological signals such as electrocardiograms (ECG), forward secrecy is not fully considered. This means that adversaries are able to extract users' health data hidden in previous stego signals by using compromised long-term secrets. Moreover, to reduce communication overhead, compression techniques are introduced in some steganography-based methods. However, the imperceptibility and embedding capacity of these solutions are sacrificed. To solve the above issues, in this study, we adopt Singular Value Decomposition (SVD) and the Bose-Chaudhuri-Hocquenghem (BCH) codes to design an efficient and secure health data propagation scheme based on steganography and compression. In our design, the BCH codes are used to update the encryption key and change the embedding locations in each steganography process, thus achieving forward secrecy and further enhancing the security of steganography. Moreover, a two-stage compression method is proposed in our scheme to compress the signals during signal processing and compression phases, which effectively reduces the communication overhead. Security analysis and the experimental results show that our proposed scheme enhances security while achieving an elaborate balance between imperceptibility, embedding capacity, and compression.**

***Index Terms*— Electronic health networks, steganography, privacy protection, electrocardiogram, compression, singular value decomposition, Bose-Chaudhuri-Hocquenghem code.**

I. Introduction

THE electronic health (e-health) networks have been in rapid development in recent years since they can provide convenient remote real-time monitoring and other low-cost medical services through the Internet. In e-health networks, various health data (e.g., blood pressure, electrocardiogram signals) are collected by medical sensors deployed on or around a patient and then transmitted over the Internet to healthcare providers (e.g., hospitals, medical service companies) for medical services such as diagnosis or treatment [1]. However, in e-health networks, sensitive health data is transmitted through the Internet, which raises great concern about the leakage of users' health status and personal privacy [2], [3].

To protect health data transmitted in e-health networks, steganography-based solutions have been researched in recent years. In these solutions, the health data is embedded in physiological signals or medical images acting as cover carriers. Since only the communication parties can observe the existence of the sensitive information and then retrieve it, the steganography-based approach can achieve privacy protection of user's health data in unsecured public channels [4]. Furthermore, the steganography-based methods can also reduce the communication overhead by embedding the health data (e.g., blood pressure, temperature) into the medical images (e.g., Magnetic Resonance Imaging (MRI)) or physiological signals (e.g., Electrocardiogram (ECG), Electroencephalogram(PPG), Electroencephalogram (EEG)). The electrocardiogram (ECG) is one of the most important biological signals that carry a lot of important information about heart health and is commonly used for daily health monitoring in e-health networks [5]. Therefore, ECG signals are more suitable as carriers for steganography than other transmitted physiological signals such as EEG and PPG, which are not typically transmission signals in e-health networks.

Over the past decade, ECG-based steganography methods [6], [7], [8], [9], [10], [11], [12], [13] have been continuously improved to meet the availability, security, and performance requirements of e-health networks. However, early works [6], [7], [8], [9] focused on embedding methods of health data while ignoring the availability of steganography carries. To achieve availability, the stego ECG signal should be visually distinguishable from the original one, so that the stego ECG signal can be used directly for diagnosis [10]. Recently, Rahman et al. proposed a high availability steganography scheme [10] by reconstructing the lossless original biosignal via Binary Golay Code. However, the error correction approach adopted in their scheme [10] limits the embedding capacity, and some security properties such as forward secrecy are not provided in their scheme.

Manuscript received 25 January 2023; revised 24 July 2023; accepted 31 August 2023; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor J. S. Sun. Date of publication 18 September 2023; date of current version 18 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62172303 and in part by the Open Research Project of the Hubei Key Laboratory of Intelligent GeoInformation Processing under Grant KLIGIP-2019B09. The work of Kim-Kwang Raymond Choo was supported only by the Cloud Technology Endowed Professorship. *(Corresponding author: Liping Zhang.)*

Liping Zhang, Wenshuo Han, and Shukai Chen, are with the School of Computer Science, China University of Geosciences, Wuhan 430074, China (e-mail: carolyn321@163.com).

Kim-Kwang Raymond Choo is with the Department of Information Systems and Cyber Security, The University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: raymond.choo@fulbrightmail.org).

Digital Object Identifier 10.1109/TNET.2023.3313160

Forward secrecy is an important security property that needs to be satisfied in cryptography-based secure transmission schemes. In this paper, we introduce the concept of forward secrecy into the design of steganography-based schemes to provide high security. That is, forward secrecy ensures that previously embedded information will not be compromised, even if the long-term secrets and the steganography algorithm are subsequently compromised. Although forward secrecy is very important, few steganography-based schemes take this security property into account. Very recently, both Abuadbba and Khalil's scheme [12] and Zhang et al.'s scheme [13] have attempted to protect the previously hidden information by updating the long-term shared keys used to determine the embedding locations and encrypt the health data. Although Abuadbba and Khalil's scheme [12] claims that the shared keys are burned and utilized whenever biomedical signals are sent, how to generate and allocate the shared keys for each user during the steganography process remains unsolved. In Zhang et al.'s scheme [13], a key agreement scheme needs to be designed to generate three independent shared keys for each steganography, thus reducing the performance of the steganography. Therefore, how to achieve forward secrecy in ECG-based steganography is a challenge.

In addition, large amounts of health data need to be transmitted every day in e-health networks, so security mechanisms should be designed to minimize communication overhead. The compression technique can be used to compress physiological signals, so it is usually adopted in ECG signal transmission schemes to reduce communication overhead. However, introducing the compression technique into ECG-based steganography will face challenges, as compression of ECG signals sacrifices imperceptibility and embedding capacity. Both steganography and compression are considered in two schemes recently proposed by Soni et al. [14] and Banerjee and Singh [15]. Although their schemes [14], [15] reduce the communication overhead, the embedding capacity of these schemes is decreased and the imperceptibility becomes worse. So, how to achieve an elaborate balance between imperceptibility, embedding capacity, and compression is another challenge.

In this paper, we design an efficient and secure health data propagation scheme based on steganography for e-health networks. In our proposed scheme, two secrets are generated in each steganography to achieve the change of embedding locations and encryption keys, thus enhancing the security of steganography. Since the shared key is updated by one of the generated secrets, our proposed scheme successfully provides forward secrecy. In addition, the two-stage compression method is designed in our proposed steganography approach to reduce communication overhead. The main contributions of this paper are as follows:

- In this study, we present an efficient and secure health data propagation scheme based on steganography and compression to protect the sensitive information transmitted in e-health networks. In our design, two secrets are generated via Bose-Chaudhuri-Hocquenghem (BCH) codes in each steganography. One secret is used to encrypt the health data and the other secret is employed to update the shared keys that determine the embedding locations. Since the shared key is varied with the generated secret in each steganography, our scheme achieves forward secrecy. In addition, the change of embedding locations and the encryption key also effectively resist illegitimate retrieval.
- To reduce the communication overhead, in our design, the ECG signals are compressed in two stages of the whole steganography process. During the ECG signal processing phase, the truncated Singular Value Decomposition (SVD) matrix coefficients are compressed by converting the SVD coefficients to a shorter coefficient. After the embedding process, the stego ECG signals are further compressed using our proposed compression algorithm. The experimental results show that our proposed scheme achieves a satisfactory compression ratio.
- To achieve the balance between the imperceptibility, embedding capacity, and compression, in our scheme, the encrypted user's private information is embedded in each element of the SVD decomposition coefficients matrices U and V. According to our extensive experiments, the compression has less effect on the imperceptibility when replacing 3 digits of each element, which also ensures that the distortion of the ECG signal is minimal (less than 1%) even if the embedding capacity reaches the maximum.

II. Related Work

In the past decade, steganography approaches have been widely used to protect the user's health information by hiding sensitive health data in physiological signals such as ECG. Compared with other physiological signals (e.g., EEG, PPG), ECG signals are more commonly used for daily health monitoring and diagnosis, so they are adopted as a cover carrier in steganography schemes to realize the hidden of private information. Ibaida et al. [8] proposed an ECG-based steganography scheme for protecting patient confidential information. In their design, the Discrete Wavelet Transform (DWT) was employed to decompose the original ECG signals into several coefficient sets, and then the patient's confidential information was embedded into the DWT coefficients using the Least Significant Bit (LSB) algorithm. Another ECG steganography scheme based on DWT was proposed by Edward et al. [9]. Unlike the scheme proposed by Ibaida et al. [8], their scheme [9] adopted SVD to determine the embedding locations. Jero et al. [9] also presented an ECG-based steganography scheme using DWT and SVD. Their scheme further improved the robustness by introducing a Continuous Ant Colony Optimization (CACO) algorithm in the steganography process. These early ECG-based steganography works mainly focused on the design of embedding methods. Subsequently, to achieve high availability, Rahman et al. [10] proposed a reversible biosignal steganography method by introducing an extended Binary Golay Code. In their design, the secrets were embedded as the "error" of the Binary Golay code, thus enabling a lossless reconstruction of the biosignals. However, the embedding capacity of their scheme was limited due to the usage of Golay Code technology.

To enhance the security of steganography cryptography techniques were introduced in the design of ECG-based

steganography schemes. In Soni et al.'s scheme [11], the secrets were encrypted via an encryption key generated by pre-shared parameters before embedding. Although their scheme [11] achieved the effective embedding of encrypted secrets, it failed to provide forward secrecy since the encryption key was not updated in each steganography process and the embedding locations were fixed. To solve the above issues, Adama et al.'s [12] proposed a 3-D steganography scheme based on Fast Walsh-Hadamard Transform (FWHT). In their design, the original biosignals were converted into multiple FWHT coefficients for the embedding and a shared key was employed to encrypt the secrets and determine the embedding locations. They claimed that this shared key was burned and utilized when the biosignals were sent. However, how to generate and allocate these shared keys in each steganography remains unsolved in their scheme. In order to update the encryption key in each steganography, Zhang et al. [13] designed a key agreement scheme in ECG signals steganography. In their scheme, three independent shared keys generated via the proposed key negotiation scheme were employed to encrypt the user's private information and determine the embedding locations. Since the shared key was generated independently in each steganography, their scheme successfully provided forward secrecy. However, their scheme [13] requires performing the key negotiation process to update shared keys, thus reducing the efficiency of the steganography.

In addition, compression techniques were introduced in signal steganography schemes to reduce communication overhead. Soni et al.'s steganography scheme [14] realized efficient embedding and compression by introducing Adaptive Fourier Decomposition (AFD) and a chaotic map-based method. In their design, original ECG signals were first decomposed by the AFD using adaptively selected basis functions to achieve better compression ratio. Then, the chaotic map-based steganography was applied to the decomposed AFD coefficients to perform embedding operations. Although their scheme [14] achieved a good compression ratio, the embedding capacity was limited. Banerjee and Singh also proposed a multi-lead ECG steganography approach with compression [15] by adopting a 2-mode tucker decomposition. In their scheme [15], a Multi-Agent Supervised Learning System (MASLS) was employed to achieve the Multi-lead ECG (MECG) signal reconstruction with a high compression ratio. However, the imperceptibility and embedding capacity of their scheme need to be improved to meet the requirements of e-health networks. Although the above schemes reduce the communication overhead by introducing compression into the steganography, the imperceptibility and embedding capacity of these schemes [14], [15] are sacrificed due to the compression operations. In addition, several security properties such as forward secrecy are not fully considered in the above schemes.

In this paper, we propose an ECG-based steganography scheme using SVD and BCH codes. In our design, two secrets are generated to ensure the change of embedding locations and the encryption key, thus achieving forward secrecy and enhancing the security of steganography. Moreover, using the proposed two-stage compression method, our proposed scheme achieves a balance between security, imperceptibility, and embedding capacity while reducing the communication overhead of e-health networks.

III. Preliminaries

In this section, the basic concepts of SVD and BCH are briefly reviewed.

A. Singular Value Decomposition

The Singular Value Decomposition (SVD) [16], [17] is an important matrix decomposition method in the mathematical field, which is widely used to realize effective matrix decomposition in practical applications. It decomposes the input matrix $M_{L\times C}$ into three matrices U, S, and V. The decomposition process is illustrated as follows.

$$M = U_{L\times L} \times \Sigma_{L\times C} \times (V_{C\times C})^T \tag{1}$$

where, the $U_{L\times L}$ and $V_{C\times C}$ are complex unitary matrices, the $(V_{C\times C})^T$ is the conjugate transpose of the $V_{C\times C}$, and the $S = \Sigma_{L\times C}$ is a non-negative real rectangular diagonal matrix. SVD provides an efficient way to decompose a complex matrix into three simple matrices making it easier to research the original matrix. The diagonal singular value matrix S obtained by SVD can be adopted to remove the noise in the signals as well as enhance the image [16], [17]. In addition, SVD can also be applied to compress the signals and the images by removing the non-significant singular values and the corresponding SVD coefficients in the decomposed matrices U and V. In this paper, we adopt SVD to decompose the ECG signals into three coefficient matrices U, S, and V, where U and V will be further converted into a suitable steganographic carrier for embedding the user's private information.

B. Bose-Chaudhuri-Hocquenghem Codes

The Bose-Chaudhuri-Hocquenghem (BCH) is a class of cyclic error-correcting codes over a Galois field. To generate a BCH code generator, a Galois field $GF(q)$ and four positive integers m, n, d, c, need to be chosen first, where the q is a prime power, $2 \leq d \leq n, gcd(n, q) = 1$, and the m is the multiplicative order of q modulo n. Then the BCH codes generator can be constructed according to the following formula.

$$g(x) = lcm(m_c(x), \ldots, m_{c+d-2}(x)) \tag{2}$$

where the $m_i(x)$ is the minimal polynomial in the $GF(q)$. When encoding the message $p(x)$ with BCH, multiply the $p(x)$ by the generator, as shown in the following formula.

$$s(x) = p(x)g(x) \tag{3}$$

For decoding, the received message $s(x)$ can be decoded according to the following formula.

$$p(x) = s(x)/g(x) \tag{4}$$

Since the BCH codes can realize the control of the correct ability in the design process, the corresponding problems can be solved by designing different BCH codes. In addition, the BCH codes are easy to implement. Based on the above advantages of BCH codes, in our design, we employ it to generate two secrets.

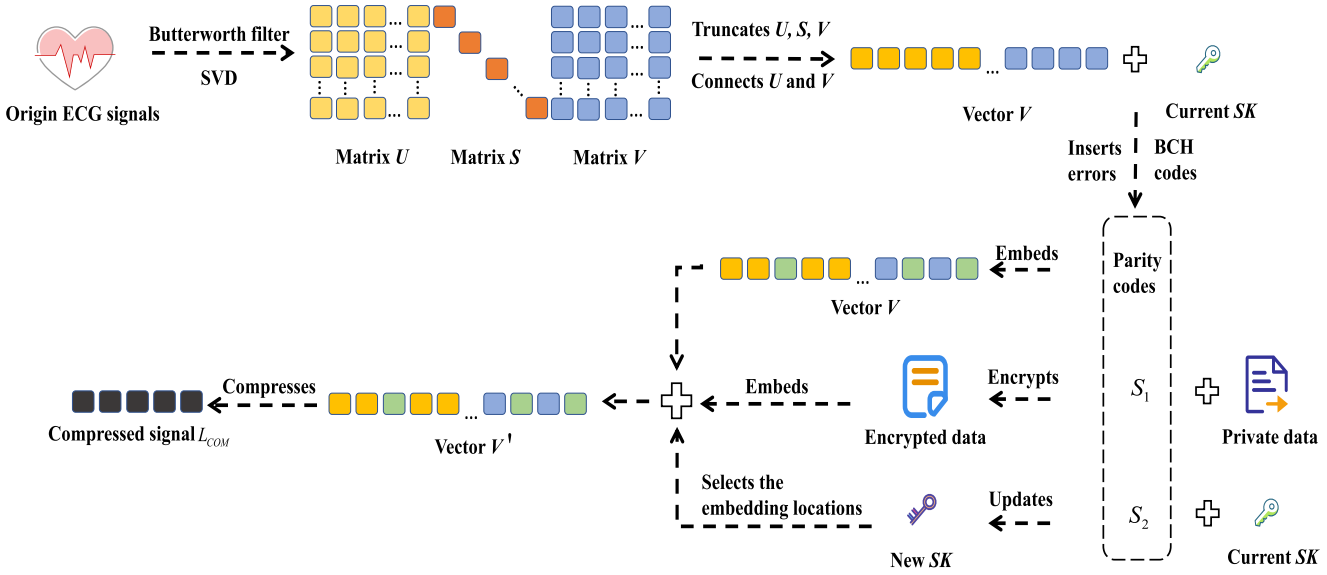

Fig. 1. Overall view of embedding and compression procedures.

IV. Our Proposed Scheme

In this section, we describe our proposed steganography-based health data propagation approach for e-health networks in detail. To enhance the security of steganography, in our design, two independent secrets are generated via BCH, one of which is used to determine the embedding locations, and the other is adopted to encrypt the private information. Since these two secrets are changed in each steganography, they ensure that the embedding locations are changed and the encryption key is varied in each run. Furthermore, the compression technique is also considered in our steganography-based approach to reduce communication overhead. In our scheme, effective compression is realized through the two-stage compression method we designed. So, the proposed approach achieves secure health data propagation with low communication overhead.

Two main phases are included in our proposed steganography scheme: the embedding and compression phases and the retrieval and decompression phases. In the first phase, the original ECG signals are denoised to obtain the more pure ECG signals. Next, two secrets are generated via BCH, and then the user's private information including health data and personal data is encrypted using one of the secrets and the embedding locations are determined via the other secret. After that, the encrypted data are embedded into the SVD truncated coefficients matrix converted by ECG signals. Finally, the stego ECG signals are compressed to complete this phase. In the next phase, the received compressed ECG signals are first decompressed, and then the user's private information is extracted and decrypted from the stego ECG signals. In addition, in our scheme, the Medical Server (MS) selects BCH codes and previously shares a 256-bit secret key SK with the legal user. The details of our proposed secure health data propagation method are described below.

A. Embedding and Compression

At this phase, the Butterworth filter denoising method is adopted to remove the noise in the original ECG signals at first. Next, two secrets S_1 and S_2 are generated by introducing the BCH codes, where the secret S_1 is used to encrypt the user's private information and the secret S_2 is employed to determine the embedding locations in each steganography. Then, the private data is encrypted with the secret S_1, and the shared key SK is updated via the secret S_2. After that, the encrypted private information is dynamically embedded in the selected locations according to the updated shared key SK. Finally, the stego ECG signals are compressed for secure transmission. The overview of the above procedures are presented in Fig. 1.

The following steps elaborate on the details of the embedding and compression phases.

1) *ECG Signal Processing*: To obtain the pure ECG signals, the Butterworth high/low pass filters are applied to remove the high-frequency and low-frequency noises in original ECG signals. Next, a 10s ECG signal that has 3600 samples is reshaped to a two-dimensional 120×30 matrix M. Then, the SVD is performed on the matrix M to generate three coefficient matrices U, S, and V, where U and V are 2-dimensional matrices of size 120×120 and 30×30, respectively, and S is a singular value matrix of size 120×30. After that, matrices U and V are truncated to a 120×7 matrix U' and a 7×30 matrix V' according to the truncate value t. Since the compression ratio and signal distortion are contradictory and both affected by the truncation values, we performed extensive experiments with different truncation values to find the most suitable one to balance them. In our experiments, the SVD coefficient matrix was truncated with various truncation values and the corresponding compression rates were recorded. Next, the signal was reconstructed using the remaining SVD coefficient matrix, and then the Percentage Residual Difference (PRD) between the reconstructed signal and the original signal was further calculated. As shown in Fig. 2, an optimal balance between compression ratio and imperceptibility can be reached with a truncation value of 7, so we set $t = 7$ in our design. Finally, all the elements in the truncated matrices U' and V' are rounded to 6th decimal and then multiplied by 1000000 to generate two corresponding integer lists $L_{U'}$ and $L_{V'}$, respectively. After the above steps, the lists $L_{U'}$ and

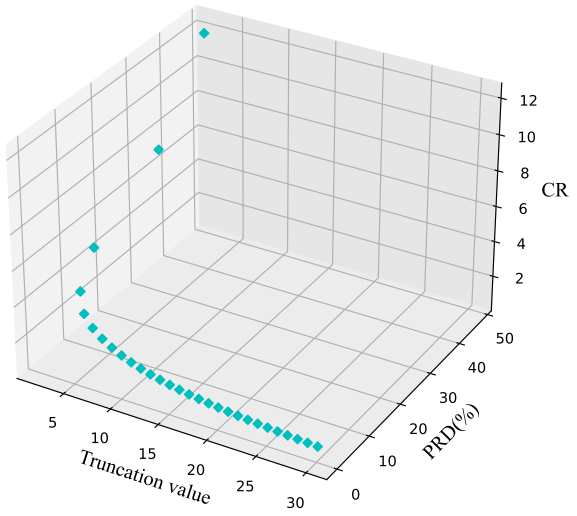

Fig. 2. The impact of truncation values on the PRD values and CR values.

$L_{V'}$ are connected into a new list L_{UV}, at which time the compression in this phase is finished and the whole signal processing phase is also completed.

2) *Two Secrets Generation*: In this step, two secrets S_1 and S_2 are constructed using BCH codes for subsequent encryption and dynamic selection of embedding locations processes. In our design, the BCH (15, 7, 2) error correction code is employed. In this error correction code, the length of the data bits is 7 and the length of the parity bits is 8. Furthermore, this code has the ability to correct 2-bit errors in the encoded information. In order to generate two secrets, the first step is to construct an integer list L_{loc} with length 64 according to Algorithm 1. In the list L_{loc}, each integer marks the position of an element in the list L_{UV}. Next, for each integer in L_{UV} marked by the list L_{loc}, the last four to six digits are extracted and then converted to a binary vector V_{bin}. Note that the last seven bits of the vector V_{bin} are prepared for the insertion of "errors". For example, assume that there is an integer 1135245 in the list L_{UV}, the digit 135 can be obtained by extracting the last four to six digits from it, and then this digit 135 is converted to a binary vector 10000111 and the last seven bits 0000111 are retained. After that, two bits are randomly selected in the vector V_{bin}, whose positions in V_{bin} are loc_1 and loc_2 with the values v_1 and v_2, respectively. Then the values of the selected two bits are reversed (i.e. 0 to 1 and 1 to 0) to prepare for the subsequent generation of S_1, S_2, and the embedding of BCH parity codes with the following steps:

- If the value of loc_1 is equal to or less than 3, set the current bit of the secret S_1 to 0, otherwise set this bit to 1. This process is repeated for all of the loc_1 and loc_2 generated from the 64 vectors (V_{bin}) until a 128-bit secret S_1 is obtained.
- To construct the secret S_2, all the v_1 and v_2 obtained from the 64 vectors (V_{bin}) are sequentially appended to the end of secret S_2. After all the 128 bits are determined, the secret S_2 is generated.
- To insert the BCH parity codes into the corresponding positions in list L_{UV}, first the list V_{bin} is encoded using the BCH codes to get the 8-bit BCH parity codes L_{BCH}. Next, the last 8-bit of the vector V_{bin} is replaced by the BCH parity codes L_{BCH}. Finally, after all marked elements in the list L_{UV} are modified according to the above operations, the embedding of BCH parity codes is completed.

After the above steps, the "errors" are removed from every vector V_{bin} by reversing the value of each error (i.e. 0 to 1 and 1 to 0).

Note that, in our design, the two secrets S_1 and S_2 are unrelated and changed in each run, ensuring that the shared keys and the encryption keys constructed by them are varied in each steganography process.

Algorithm 1 sequence Generation 1

Input:the shared key SK
Output:the sequence of rows L'_{loc}

1: SK: shared key
2: L: list of numbers
3: $INT()$: the function of converting the binary string to the decimal number
4: L_{loc}: sequence of embedded locations
5: **for** $i = 1$ to 30 **do**
6: $SK = SHA512(SK)$
7: $SK_{tmp} = SK_{tmp} + SK$
8: **end for**
9: converts SK_{tmp} to binary format
10: **for** $i = 1$ to 1050 **do**
11: appends $INT(SK_{tmp}[i : i + 16])\ mod\ 1050$ to L
12: **end for**
13: transforms the L into a dictionary, where the keys are the index of elements and the values are the value of elements.
14: sorts the dictionary by the values, and the sorted keys are the L_{loc}
15: $L'_{loc} \Leftarrow L_{loc}[1 : 64]$

3) *Encryption*: To resist a brute-force search, in our design, the standard Advanced Encryption Standard (AES) algorithm is utilized to encrypt private information and health data before embedding. Specifically, the secret S_1 generated in the above steps is adopted as the symmetry encryption key of the AES algorithm to achieve the encryption. After encryption, the generated ciphertext is prepared for the subsequent embedding process.

Notably, in our scheme, the secret S_1 is generated independently and varied in each steganography process, so that the symmetry encryption key used in each run is unrelated to other encryption keys adopted in other runs. In this case, an adversary is unable to obtain the current encryption key through the previously compromised ones, thus further enhancing the security of the steganography.

4) *Dynamic Selection of Embedding Locations*: In this step, the shared key SK is employed to determine the embedding locations. In our scheme, the initial shared key SK is adopted in the first steganography process, and the shared key is updated in each subsequent steganography process as follows.

$$SK_{new} = SK \oplus h(S_2) \tag{5}$$

where the SK is the shared key generated in the previous steganography and the S_2 is the current secret. The $h()$ is a secure one-way hash function with a 256-bit output, which is used to convert a 128-bit S_2 to a 256-bits value. In this way, the shared key used in each steganography is different and unrelated to each other. Next, the selection of embedding locations via SK_{new} is described in detail. In the first step, the first 72 bits are selected from the 128-bit shared key SK_{new} and then divided into nine 8-bit segments. Then, each segment is converted to an integer. After this, a position dictionary D_{loc} is generated, which consists of all the integers generated above and the corresponding index (the index is 1-9). Subsequently, the integers in the location dictionary D_{loc} are sorted in ascending order and then the corresponding varying indices are employed to determine the embedding locations. Finally, the 9-bit ciphertext is embedded into the list L_{UV} according to the embedding locations computed above. Furthermore, the SK_{new} is updated by computing $SK_{new} = h(SK_{new})$ after each embedding of a 9-bit secret. The above process will be repeated until all the ciphertext is embedded into the list L_{UV}. After all steps, the stego list L'_{UV} is generated and the embedding process is completed. The detailed embedding steps are described in algorithm 2.

Notably, since the shared key is updated in each steganography process, our proposed scheme achieves the change of embedding locations in each round.

Algorithm 2 sequence Generation 2

Input:the secret SK_{new}**,** L_{UV}
Output:the sequence of embedding locations L'_{UV}

L: list of numbers
L_{emb}: sequence of embedded
L_D: decimal number list
D_{loc}: a dictionary
$SK_{tmp} = SK_{new}[0:72]$
$h()$: a hash function
divides SK_{tmp} to nine binary segments.
converts nine binary segments to the decimal number and appends them to the list L_D
for $i = 1$ to 9 **do**
    appends the element to the dictionary D, the key is i, and the value is $L_D[i]$
end for
sorts the dictionary D_{loc} by the values, and the sorted keys are the L_{emb}
embeds 9 bits private information into the L_{UV} according to L_{emb}.
updates $SK_{new} = h(SK_{new})$.
repeats the above process until all private information is embedded into the L_{UV}.
after embedding, the L'_{UV} is obtained.

5) *Compression*: In this step, the list L'_{UV} is compressed through our compression algorithm. First, in the list L'_{UV}, the difference between the second and the first number is computed, and then the difference between the third number and the second number is calculated, and so on, until the difference between the last number and the previous number is computed. Then, for each pair of minuend and its corresponding difference, their absolute values are compared and the smaller one is converted to a binary number r. Next, a four-bit header is added to each binary number r to construct a new binary string r_{bin}, where the header is generated according to Table I. After that, all the constructed binary strings r_{bin} form the compressed list L_{com}.

After the compressed list L_{com} is constructed, all the binary strings r_{bin} in the list L_{com} are connected to generate a long binary string. Then, this binary string is divided into several 32-bit binary strings and each string is converted to an integer. In this way, all the binary strings in the compressed list L_{com} are converted to integers, thereby constructing a new integer compressed list L'_{com} with a shorter length than the list L_{com}. After the above steps, the compression of the ECG signal is achieved. For example, given the list L'_{UV} [225142, 225128, 6541, -6499988, -6499722, ...]. The compression process consists of the following six steps:

Step1: calculate the differences between adjacent integers starting from the second integer. Then we get a new list L_1: [225142, -14, -218587, -6506529, 266, ...].

Step2: compare the absolute values of the corresponding positions in the original list L'_{UV} and the new list L_1, retaining the smaller value. This operation results in a new list L_2: [225142, -14, 6541, -6499988, 266, ...].

Step3: convert the values in L_2 to binary, then we obtain a binary list L_3: [00110110111011101 10, -1110, 000110011 0001101, -011000110010111010010100, 11100010, ...].

Step4: prepend a 4-bit header to each binary string in L_3 according to Table I. This operation generates a compressed list L_{com}: [0010001101101111011101 10, 1100000000001110, 0001000110011000 1101, 1011011000 110010111010010100, 0100000011100010, ...].

Step5: concatenate all the binary strings in L_{com} into a single binary string: 0010001101101111011101 1011000000000011100001000110011000110110110110001 1001011101001010001000000011100010...

Step6: divide the concatenated binary string into several 32-bit segments and convert each segment into an integer. This operation yields an integer compressed list L'_{com}: [594507456, 236034267, 1663996992, ...].

After the above steps, the list L'_{UV} [225142, 225128, 6541, -6499988, -6499722, ...] is compressed into a new integer list L'_{com} [594507456, 236034267, 1663996992, ...].

B. Decompression and Retrieval

In this phase, the legal user can obtain the private information through three steps: decompression, retrieval, and decryption. The whole process is the inverse process of the embedding and compression. First, the stego list L'_{UV} is computed by decompressing the received information. Then, the locations of the BCH parity codes can be found and the "errors" can be extracted by using the generated stego list L_{UV} and the current shared key SK. After that, the two secrets S'_1 and S'_2 are constructed via the BCH parity codes obtained above. Subsequently, the secret S'_2 is used to determine the embedding locations and the secret S'_1 is employed for decryption. Finally, the ECG signals are reconstructed by the

TABLE I
GENERATION RULES FOR THE HEADER

Positive or Negative	Difference	Length of the binary number	Header	Length of the binary number after padding
Negative	Y	$L \leq 12$	1100	12
		$12 < L \leq 16$	1101	16
		$16 < L \leq 20$	1110	20
		$20 < L \leq 24$	1111	24
	N	$L \leq 12$	1000	12
		$12 < L \leq 16$	1001	16
		$16 < L \leq 20$	1010	20
		$20 < L \leq 24$	1011	24
Positive	Y	$L \leq 12$	0100	12
		$12 < L \leq 16$	0101	16
		$16 < L \leq 20$	0110	20
		$20 < L \leq 24$	0111	24
	N	$L \leq 12$	0000	12
		$12 < L \leq 16$	0001	16
		$16 < L \leq 20$	0010	20
		$20 < L \leq 24$	0011	24

inverse SVD and the legal user obtains the private information successfully.

V. SECURITY ANALYSIS

In this section, we discuss the security of our proposed steganography-based health data propagation scheme by analyzing the following security properties and possible attacks: illegitimate retrieval with/without the steganography algorithm, forward secrecy, and modification attacks.

A. Adversary Model

In this paper, we adopt the classic Dolev and Yao adversary model [18], which is one of the most widely used adversary models in applied security research [19]. In this model, the adversary is assumed to have full control over the communication channel and possesses the ability to obtain all the transmitted messages. Then the adversary can launch the following activities:

- Interception: The adversary can be a passive adversary, in this case, he/she has the ability to intercept all the messages transmitted between the user and the server in e-health networks.
- Modifying/Tampering: The adversary can also be an active adversary, under this case, he/she possesses the ability to modify the transmitted message and then send the fraud message to the server to impersonate the legal user.
- Inference: Based on the intercepted message, the adversary has the ability to figure out the signal processing method. In addition, if the encryption keys are correlated, the adversary also has the ability to guess the encryption key by using the compromised one.

B. Security Requirement Discussion

1) *Illegitimate retrieval without steganography algorithm*

Assume that an adversary intercepts all the messages transmitted between the communication parties and has knowledge of the signal-process algorithm. Then he/she attempts to retrieve the user's private information from the intercepted messages. In our design, the original ECG signal is converted to an integer list using SVD, and the user's private information is embedded in this list. In our design, 9 bits in each integer are replaced by secrets according to our embedding algorithm 2, so that an integer in the list can hide a 9-bit secret. Furthermore, in our scheme, the user's private information is encrypted using the generated 128-bit secret S_1 before embedding. It is worth noting that the secret S_1 is generated independently by BCH codes and varied in each steganography process. In the above case, if the adversary intercepts a transmitted compressed stego ECG signal containing 1050 integers, he/she needs to decompress the message first and then try $1050! \times 2^9 \times 2^{128} \approx 2.4569 \times 10^{2759}$ times to obtain the user's private information. Obviously, it is infeasible to retrieve the user's private information by the above exhaustive search.

Therefore, our proposed steganography-based health data propagation scheme can resist illegitimate retrieval without a steganography algorithm.

2) *Illegitimate Retrieval With Steganography Algorithm*

Compared with illegitimate retrieval without a steganography algorithm, the additional information known by the adversary is the steganography algorithm. In this case, to retrieve the user's private information from the intercepted messages, the adversary may launch attacks using the following two methods.

- The adversary tries to guess the shared key SK generated in previous steganography process and then computes the secret S_1 and S_2 via the guessed SK. Then he/she executes the steganography algorithm to retrieve the embedded private information via the two computed secrets.
- The adversary directly guesses the encryption key and the embedding locations and then extracts the hidden private information with the steganography algorithm.

In our design, the secret S_2 is generated for updating the shared key which is used to determine the embedding locations. Since the secret S_2 varies in each steganography, the shared key also changes in each run. So, under the first case, the adversary needs to try $2^{256} \approx 1.1579 \times 10^{77}$ times to get the shared key successfully. Obviously, it is highly infeasible

for the adversary to retrieve the user's private information by correctly guessing the shared key. Next, we analyze the second case as mentioned above. Suppose an adversary obtains a stego list L'_{UV}, which contains 1050 integers. Then, he/she guesses an encryption key and attempts to determine the embedding locations through brute-force search. In this case, the adversary needs to try $2^{9\times1050} \times 2^{128} \approx 1.8420 \times 10^{2883}$ times to obtain the user's private information successfully. Therefore, the adversary has no ability to get the hidden information by adopting the above method. In summary, the adversary cannot successfully retrieve the user's private information from the stego list L'_{UV}, even if he/she has the steganography algorithm.

So, our proposed steganography-based health data propagation scheme resists illegitimate retrieval with a steganography algorithm successfully.

3) *provision of forward secrecy*

The forward secrecy property ensures that even if the long-term secrets of one or more communication parties are compromised, the user's private information hidden in previous stego signals cannot be retrieved. This security property provides protection for previously hidden information when both the steganography algorithm and the long-term keys are compromised. In our scheme, the shared key is updated after every successful steganography, and only the updated shared key needs to be stored. Furthermore, in our design, the shared key adopted in each steganography is unrelated to the other shared keys employed in other steganography processes. In this case, even if the adversary compromises the current shared key, he/she cannot obtain the previously shared keys using the compromised one. So the adversary cannot retrieve the user's private information from previously intercepted messages using the compromised shared keys.

Therefore, our proposed steganography-based health data propagation scheme achieves forward secrecy.

4) *resistance of modification attacks*

Suppose an adversary modifies the transmitted message and then sends the fraud one to the server to launch a modification attack. However, these attacks can be detected when the server performs the retrieve operations. This is because the adversary cannot generate a valid stego ECG signal without the knowledge of the current encryption key and the embedding locations or the shared key. Similar to the security analysis of illegitimate retrieval, it is impossible for an adversary to construct a valid stego ECG signal by correctly guessing the encryption key and the embedding locations or the shared key.

So, our proposed steganography-based health data propagation scheme resists modification attacks successfully.

VI. Implementations and Discussion

A. Datasets

In this subsection, we introduce the datasets adopted in our experiments. Since the MIT-BIH Arrhythmia database (MITDB) is widely used in experiments for steganography performance evaluation, we adopt it to simulate real-world application scenarios and evaluate the performance of our proposed steganography-based health data propagation scheme. This database includes clinical ECG signals recorded at 360 samples per second per channel with an 11-bit resolution from 47 individuals [20], [21]. Furthermore, there are five categories in MITDB, namely normal (N), left and right bundle branch block (L/R), premature ventricular contraction (V), and aberrated atrial premature (A). In particular, it is noted that each ECG recording contains the data from two channels, and we only adopt the ECG signals collected from the MLII channel to perform our experiments, as this channel has better noise resistance than the other one [20], [21].

B. Experiments and Results

We execute extensive simulation experiments to evaluate our proposed steganography-based health data propagation scheme in terms of compression, imperceptibility, embedding capacity, and availability. To simulate the communication of e-health networks, two computers are connected through the Internet with a bandwidth speed of 100 Mbps. One of the computers is an Intel Core 7300H CPU 2.50GHz personal computer with 16 GB random-access memory (RAM). This computer is used to simulate decompression and retrieval operations performed by the server. The other computer is employed as a user to simulate the embedding and compression processes, which has an Intel Core I5-10400 CPU with a clock speed of 2.90 GHz and 16 GB RAM. The experimental code is written in python 3.8.

In our experiments, the first 10-second segments of each ECG signal were employed to hide the user's private information, which contained a total of 3600 floating point values. These segments were first decomposed to generate three coefficient matrices U, S, and V using SVD, and then an integer list L_{UV} was constructed for embedding. Meanwhile, two secrets S_1 and S_2 were generated using our proposed method via BCH codes. After that the user's private information was encrypted and hidden in the list L_{UV} by using the two generated secrets. Finally, the compression process was executed. Subsequently, after receiving the compressed stego ECG signals, the server performed inverse operations to obtain the user's private information.

Next, we summarize and illustrate the experimental results from the following four aspects:

1) *compression*

To evaluate the efficiency of the compression method, the Compression Ratio (CR) is introduced in our experiments, whose value reflects the compression ability [22]. The CR value can be calculated as follows.

$$CR = \frac{fsize(carrierdata) + fsize(embeddeddata)}{fsize(compresseddata)} \quad (6)$$

where fsize() represents the size of the file. In general, the larger value of CR indicates a better efficiency of compression. As shown in Table II and Fig. 3, for the different categories of the ECG signals in the MIT-BIH database, the CR values in our experiments are all greater than 8, indicating that our proposed scheme achieves a satisfactory compression ratio.

2) *Imperceptibility*

In our experiments, two statistical indicators are used to quantitatively measure imperceptibility. One of the indicators is Percentage Residual Difference (PRD) [23], which is widely adopted to measure the differences between the original signal

TABLE II
CR VALUES FOR VARIOUS CATEGORIES OF ECG SIGNALS IN MIT-BIH DATABASE

Sample No.	Category	CR
100	N	8.15
109	L	8.06
111	L	8.28
118	N	8.37
124	N	8.21
200	V	8.26
201	A	8.10
208	V	8.29
231	R	8.15

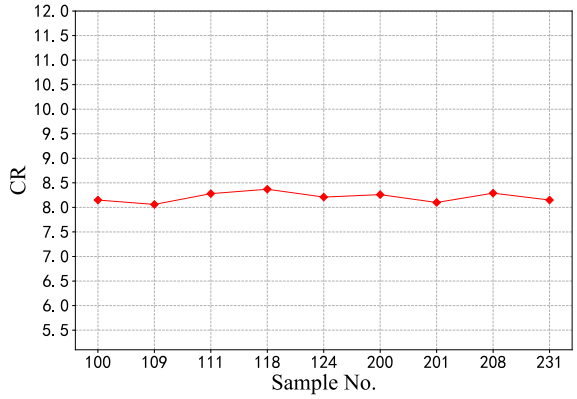

Fig. 3. The CR values of different ECG signals (10s).

TABLE III
PRD VALUES FOR VARIOUS CATEGORIES OF ECG SIGNALS IN MIT-BIH DATABASE

Sample No.	Category	PRD(%)
100	N	0.57
109	L	0.58
111	L	0.53
118	N	0.52
124	N	0.54
200	V	0.54
201	A	0.58
208	V	0.52
231	R	0.54

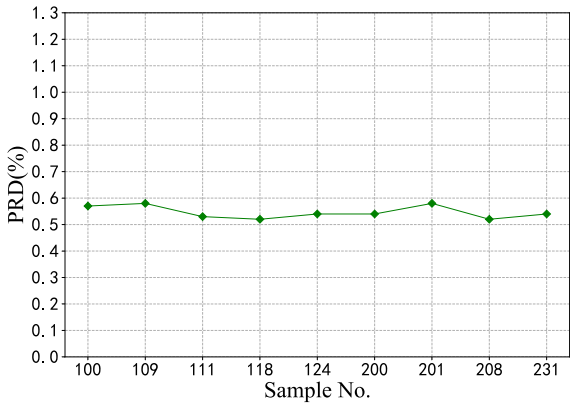

Fig. 4. The PRD values of stego ECG signals (10s).

and the stego one. The PRD can be computed as follows.

$$PRD = \sqrt{\frac{\sum_{i=1}^{N}(X_i - Y_i)^2}{\sum_{i=1}^{N} X_i^2}} \times 100\% \tag{7}$$

where X_i and Y_i present the original signal and the stego one, respectively. In addition, the symbol N represents the length of the signal. Generally, the effect of steganography operation on the original signal can be ignored when the PRD value is less than 1%. Table III and Fig 4 present the experimental results of the PRD values for the different categories of ECG signals in MIT-BIH. From Table III and Fig 4, all the PRD values are around 0.5% and less than 1%, indicating that our steganography method has less effect on original ECG signals.

Another statistical indicator employed in our experiments is Peak Signal-to-Noise Ratio (PSNR) [24]. The value of PSNR quantifies the similarity between the original signal and the stego one. This value is calculated by the following formula.

$$PSNR(dB) = \frac{max(X_c)}{\frac{1}{N}\sum_{n=1}^{N}(X_c - X_w)^2} \tag{8}$$

where the symbol X_c and X_w represent the original signal and the stego original and the N is the signal length. Obviously, the larger the PSNR value, the more similar the original signal is to the stego one. As shown in table IV and Fig. 5, the PSNR values are all greater than 55dB for various categories of ECG signals in the MIT-BIH database. These experimental results show that the PSNR values of our proposed steganography method reach a high level, that is, the original ECG signal is highly similar to the stego one and is difficult to distinguish.

TABLE IV
PSNR VALUES FOR VARIOUS CATEGORIES OF ECG SIGNALS IN MIT-BIH DATABASE

Sample No.	Category	PSNR(dB)
100	N	60.62
109	L	57.02
111	L	59.02
118	N	60.85
124	N	59.77
200	V	62.43
201	A	58.13
208	V	58.67
231	R	56.44

In summary, our proposed steganography-based health data propagation scheme achieves high imperceptibility for various categories of ECG signals.

3) *Embedding Capacity*

In our experiments, the embedding capacity is quantified by the number of secret bits that can be hidden in an ECG signal. Since the embedding capacity beyond a certain value can affect the visual fidelity, we performed extensive experiments to achieve an elaborate balance between the embedding capacity and the imperceptibility. According to our experiments, this trade-off can be realized when an integer in the list L_{UV} is embedded with a 9-bit secret. For example, in our proposed scheme, the embedding capacity of a 10-second ECG signal can reach 8874 bits (the 64 SVD coefficients are used to embed BCH parity codes.).

4) *Availability*

We evaluate the availability of our proposed scheme from the following two aspects: 1) whether the stego ECG signals

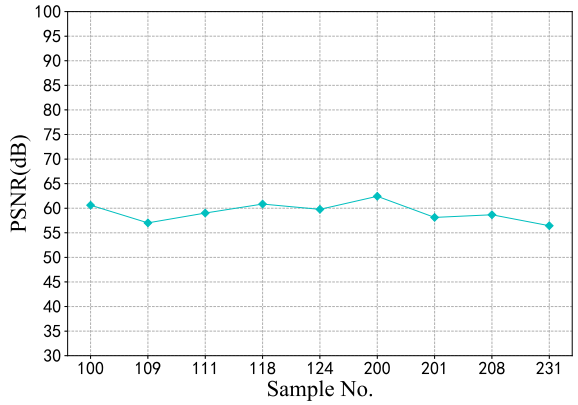

Fig. 5. The PSNR values of stego ECG signals (10s).

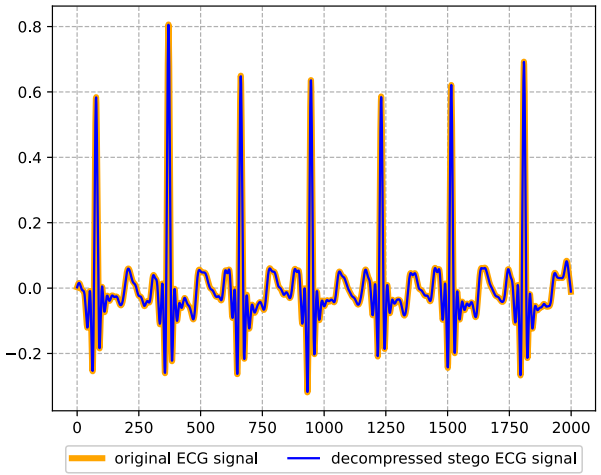

Fig. 6. Comparison of imperceptibility of the original ECG signal vs the decompressed stego ECG signal.

can be directly used for clinical diagnosis; 2) whether the retrieval process causes the loss of the embedded data. Our experiment results show that the PRD values are less than 1% and the PSNR values reach a high level. As shown in Fig. 6, the decompressed stego ECG signal is visually indistinguishable from the original one. So, in our proposed scheme, the doctor can directly use the decompressed stego ECG signals for clinical diagnosis without extracting the embedded data. In addition, the bit error rate (BER) is employed in our experiments to evaluate whether the retrieval operation causes the loss of the embedded data. The BER value can be computed by the following formula:

$$BER = \frac{error\ bits\ received}{total\ bits\ sent} \times 100\% \tag{9}$$

Table V presents the BER values of our proposed scheme. From table V, the BER values of the retrieved secret messages are all zero, indicating that our proposed scheme achieves lossless retrieval.

In conclusion, the experimental results show that our proposed scheme achieves lossless retrieval and meets the requirements of clinical diagnosis. So, our proposed steganography-based health data propagation satisfies the availability requirements.

TABLE V
BIT ERROR RATE (BER) OF RETRIEVED SECRET MESSAGE FROM ECG SIGNALS

Sample No.	Category	BER(%) for Vary Secret Length (bits)				
		1000	2000	3000	4000	5000
100	N	0	0	0	0	0
109	L	0	0	0	0	0
111	L	0	0	0	0	0
118	N	0	0	0	0	0
124	N	0	0	0	0	0
200	V	0	0	0	0	0
201	A	0	0	0	0	0
208	V	0	0	0	0	0
231	R	0	0	0	0	0

C. Comparative Summary

In this subsection, we compare our proposed steganography-based health data propagation scheme with four other related schemes [10], [13], [14], [15] on performance metrics, including CR, PRD, BER, embedding capacity, and forward secrecy. The ECG signal used in the comparisons is a 10s signal.

As summarized in Table VI, Soni et al.'s scheme [14], Banerjee and Singh's scheme (adopted multilead electrocardiogram signal) [15] and our scheme provide both steganogaphy and compression while Rahman et al.'s scheme [10] and Zhang et al.'s scheme [13] only consider the steganography without compression. From Table VI, the CR value of our scheme (8.21) is close to Soni et al.'s scheme [14] (11.79) and lower than Banerjee and Singh's scheme [15] (22). Although Banerjee and Singh's scheme [15] achieves the highest compression ratio (22), the embedding capacity and imperceptibility are sacrificed. According to Table VI, the PRD values of schemes [10], [13] without compression are better than the schemes [14], [15] that provide both steganography and compression. That is because the compression operations affect the PRD values. In addition, among the three schemes with compression, the PRD value of our scheme (0.55%) and Soni et al.'s scheme [14] (0.32%) are all less than 1% while Banerjee and Singh's scheme [15] (3.62%) is greater than 1%, indicating that their scheme [15] cannot satisfy the requirement of imperceptibility. So the stego ECG signals in Banerjee and Singh's scheme [15] cannot be directly used for clinical diagnosis. However, through our design, the PRD value of our scheme (0.55%) is less than 1%, thus providing satisfactory imperceptibility and enabling clinical diagnosis using stego ECG signals.

Moreover, as shown in Table VI, the BER values of the four steganography schemes [10], [13], [14], [15] and our scheme are all zero, indicating that the four steganography methods all realize lossless retrieval. Furthermore, for a 10s ECG signal, the embedding capacity of the other three steganography [10], [13], [14] schemes and our scheme are 10800bits, 13500bits, 216bits, and 8874bits respectively. From Table VI, our scheme can embed more user's private information in compressed ECG signals than Soni et al's scheme [14]. In addition, since the compression ratio of our scheme is 8.21, for the same communication overhead, the embedding capacity of our scheme is larger than other two schemes without compression [10], [13]. According to the above analysis, our proposed scheme

TABLE VI
PERFORMANCE COMPARISON BETWEEN OUR PROPOSED SCHEME AND FOUR OTHER SCHEMES

	Scheme	CR	PRD(%)	BER(%)	Bits embeded	Forward secrecy
steganography	Rahman et al. [10]	-	0.0072	0	10800	N
	Zhang et al. [13]	-	0.0078	0	13500	Y
steganography and compression	Soni et al. [14]	11.79	0.3200	0	216	N
	Banerjee and Singh [15]	22	3.6200	0	-	N
	Ours	8.21	0.5500	0	8874	Y

[1] The ECG signal adopted in the Table VI is a 10s signal.

meets the requirement of imperceptibility while increasing the embedding capacity.

Furthermore, Rahman et al.'s scheme [10], Soni et al.'s scheme [14] and Banerjee and Singh's scheme [15] do not fully consider the security of steganography, especially forward secrecy. Although Zhang et al.'s [13] scheme achieves forward secrecy by introducing a key agreement scheme, the usage of key agreement scheme reduces the performance of the steganography. Moreover, how to design a lightweight key agreement scheme to generate three unrelated shared keys is also a tricky issue. In our proposed scheme, two independent secrets S_1 and S_2 are generated to encrypt the user's private information and determine the embedding locations. Since these two secrets are varied in each steganography, our scheme ensures the change of encryption key and embedding locations in each steganography. So, in our scheme, the adversary cannot retrieve the user's private information from the intercepted message, that is, our scheme can resist illegitimate retrieval with/without the steganography algorithm. Furthermore, in our design, the shared key is updated via the secret S_2 after each steganography. Therefore, even if the adversary compromises the current shared key and the steganography algorithm, he/she cannot correctly retrieve the user's private information from previously intercepted stego ECG signals via the compromised shared keys, indicating that our scheme can provide forward secrecy. In general, compared with other related works [10], [14], [15], our scheme enhances the security of steganography by providing more security properties, especially forward secrecy.

Based on the above analysis, compared with other related schemes [10], [13], [14], [15], our scheme enhances the security of steganography and achieves a balance between imperceptibility, embedding capacity, availability, and compression.

D. Potential Application Scenarios

With the arrival of old-age society, the traditional medical system is facing great pressure, which promotes the rapid development of e-health networks. The e-health networks can provide remote health monitoring, allowing people to enjoy convenient medical services at home. In the e-health networks, there are lots of sensitive data being transmitted over insecure channels such as user private information and personal health data. Our proposed health data propagation scheme can provide protection for the transmitted health data. Specifically, adopting our scheme, the user's sensitive data is encrypted and embedded in the user's ECG signals and then these stego ECG signals are compressed and sent to the specified server through the Internet. After receiving the compressed stego ECG signals, the doctor can obtain the user's private information by using our proposed scheme.

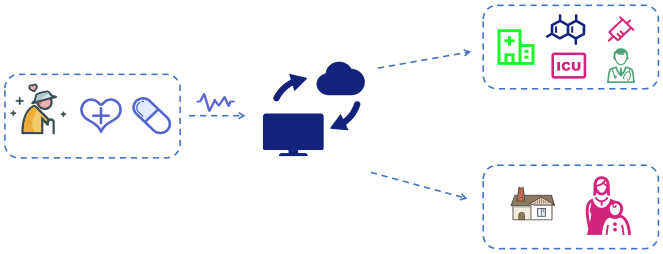

Fig. 7. An application scenario in e-health networks.

An application scenario of adopting our scheme for secure health data transmission in e-health networks is shown in Fig. 7. In this scenario, the user at home can send his/her health data securely to the remote doctor using e-health networks for diagnosis. Adopting our secure health data propagation scheme, the user's health data acquired by various medical sensors will be encrypted and then embedded in ECG signals. After that, the stego ECG signals are compressed and then transmitted to the medical server. Upon receiving the compressed stego ECG signals, the medical server first decompresses the received message and then performs the retrieval and decryption operations to get the user's private information. Then, physicians can provide a diagnosis based on these health data. Moreover, the user's family members can also access the user's health data through their mobile devices. Since our scheme achieves secure health data transmission in e-health networks, the users do not need to worry about privacy leakage while enjoying the remote e-health services.

VII. CONCLUSION

In this paper, we proposed an efficient and secure health data propagation scheme for e-health networks. In our design, the steganography approach was employed to protect the health data transmitted in the e-health networks. To enhance the security of steganography, two independent secrets were generated via BCH codes to change the embedding locations and encryption keys during each steganography process. Based on this design, our scheme achieves forward secrecy, thus providing effective protection of previously transmitted health data in the case that the current shared key and the

steganography algorithm are both compromised. The security analysis proves that our proposed scheme can not only provide forward secrecy but also resist illegitimate retrieval and impersonation attacks. In addition, a two-stage compression method was designed in our scheme to reduce the communication overhead in e-health networks. The experimental results show that our proposed scheme achieves an elaborate balance between imperceptibility, embedding capacity, and compression. Compared with other state-of-the-art schemes, our proposed health data propagation scheme is more suitable for e-health networks.

REFERENCES

[1] A. Jindal and M. Liu, "Networked computing in wireless sensor networks for structural health monitoring," *IEEE/ACM Trans. Netw.*, vol. 20, no. 4, pp. 1203–1216, Aug. 2012.

[2] X. Li, M. Wang, H. Wang, Y. Yu, and C. Qian, "Toward secure and efficient communication for the Internet of Things," *IEEE/ACM Trans. Netw.*, vol. 27, no. 2, pp. 621–634, Apr. 2019.

[3] N. Sun, J. Zhang, P. Rimba, S. Gao, L. Y. Zhang, and Y. Xiang, "Data-driven cybersecurity incident prediction: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1744–1772, 2nd Quart., 2019.

[4] P. Schöttle and R. Böhme, "Game theory and adaptive steganography," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 760–773, Apr. 2016.

[5] I. Odinaka, P.-H. Lai, A. D. Kaplan, J. A. O'Sullivan, E. J. Sirevaag, and J. W. Rohrbaugh, "ECG biometric recognition: A comparative analysis," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 6, pp. 1812–1824, Dec. 2012.

[6] C.-Y. Yang and W.-F. Wang, "Effective electrocardiogram steganography based on coefficient alignment," *J. Med. Syst.*, vol. 40, no. 3, pp. 1–15, Mar. 2016.

[7] S. Edward Jero, P. Ramu, and S. Ramakrishnan, "ECG steganography using curvelet transform," *Biomed. Signal Process. Control*, vol. 22, pp. 161–169, Sep. 2015.

[8] A. Ibaida and I. Khalil, "Wavelet-based ECG steganography for protecting patient confidential information in point-of-care systems," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 12, pp. 3322–3330, Dec. 2013.

[9] E. J. S. J. Raj, P. Ramu and R. Swaminathan, "Imperceptibility—Robustness tradeoff studies for ECG steganography using continuous ant colony optimization," *Expert Syst. Appl.*, vol. 49, pp. 123–135, May 2016.

[10] M. S. Rahman, I. Khalil, and X. Yi, "Reversible biosignal steganography approach for authenticating biosignals using extended binary Golay code," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 1, pp. 35–46, Jan. 2021.

[11] N. Soni, I. Saini, and B. Singh, "A morphologically robust chaotic map based approach to embed patient's confidential data securely in non-QRS regions of ECG signal," *Australas. Phys. Eng. Sci. Med.*, vol. 42, no. 1, pp. 111–135, Mar. 2019.

[12] A. Abuadbba and I. Khalil, "Walsh–Hadamard-based 3-D steganography for protecting sensitive information in point-of-care," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 9, pp. 2186–2195, Sep. 2017.

[13] L. Zhang, Z. Wei, W. Ren, X. Zheng, K. R. Choo, and N. N. Xiong, "SIP: An efficient and secure information propagation scheme in e-health networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1502–1516, Apr. 2021.

[14] N. Soni, I. Saini, and B. Singh, "AFD and chaotic map-based integrated approach for ECG compression, steganography and encryption in e-healthcare paradigm," *IET Signal Process.*, vol. 15, no. 5, pp. 337–351, Jul. 2021.

[15] S. Banerjee and G. K. Singh, "Quality aware compression of multi-lead electrocardiogram signal using 2-mode tucker decomposition and steganography," *Biomed. Signal Process. Control*, vol. 64, Feb. 2021, Art. no. 102230.

[16] J.-J. Wei, C.-J. Chang, N.-K. Chou, and G.-J. Jan, "ECG data compression using truncated singular value decomposition," *IEEE Trans. Inf. Technol. Biomed.*, vol. 5, no. 4, pp. 290–299, Apr. 2001.

[17] A. M. Rufai, G. Anbarjafari, and H. Demirel, "Lossy image compression using singular value decomposition and wavelet difference reduction," *Digit. Signal Process.*, vol. 24, pp. 117–123, Jan. 2014.

[18] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[19] Q. Do, B. Martini, and K.-K.-R. Choo, "The role of the adversary model in applied security research," *Comput. Secur.*, vol. 81, pp. 156–181, Mar. 2019.

[20] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, May 2001.

[21] A. L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. 1–12, Jun. 2000, doi: 10.1161/01.cir.101.23.e215.

[22] S. K. Mukhopadhyay, M. O. Ahmad, and M. N. S. Swamy, "Compression of steganographed PPG signal with guaranteed reconstruction quality based on optimum truncation of singular values and ASCII character encoding," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 7, pp. 2081–2090, Jul. 2019.

[23] S. Alam, R. Gupta, and K. D. Sharma, "On-board signal quality assessment guided compression of photoplethysmogram for personal health monitoring," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[24] H. Sajedi, "Applications of data hiding techniques in medical and healthcare systems: A survey," *Netw. Model. Anal. Health Informat. Bioinf.*, vol. 7, no. 1, p. 6, Dec. 2018.

Liping Zhang received the Ph.D. degree in information security from the Huazhong University of Science and Technology in 2009. She is currently an Associate Professor in information and network security with the China University of Geosciences. Her research interests include network security, key management and distribution, and privacy protection. She has published over 30 research articles, most of which are refereed international journal articles, including IEEE/ACM/IET journal articles. She is the principal grant holder of three externally funded research projects.

Wenshuo Han received the B.Sc. degree in information security from the China University of Geosciences, Wuhan, China, in 2021, where he is currently pursuing the M.Sc. degree in information security (computer science). His research interests include communications security and network security.

Shukai Chen received the B.Sc. degree in network engineering from PLA Army Engineering University in 2020. He is currently pursuing the M.Sc. degree in electronic engineering (computer science) with the China University of Geosciences. His research interests include ECG authentication, communications security, and network security.

Kim-Kwang Raymond Choo (Senior Member, IEEE) received the Ph.D. degree in information security from the Queensland University of Technology, Brisbane, QLD, Australia, in 2006. He currently holds the Cloud Technology Endowed Professorship at The University of Texas at San Antonio, San Antonio, TX, USA. He is the founding co-Editor-in-Chief of ACM Distributed Ledger Technologies: Research and Practice, and the founding Chair of IEEE Technology and Engineering Management Society Technical Committee on Blockchain and Distributed Ledger Technologies.