Applied Intelligence
https://doi.org/10.1007/s10489-023-04550-3

Medical image security and authenticity via dual encryption

Kishore Babu Nampalle[1] · **Shriansh Manhas**[2] · **Balasubramanian Raman**[1]

Accepted: 26 February 2023

Abstract
Since medical images include sensitive patient information, security is the top priority during transmission. In addition to protecting patient data from potential criminals, security helps to confirm the field staff's identity. However, many medical institutions still need to adopt advanced security measures. In this paper, a new dual encryption method is proposed that implements blowfish and signcryption in a certificateless generalized form. The proposed method has an advantage over other methods due to its computational cost-effectiveness and speed. The performance measurements used to assess a proposed strategy's effectiveness are PSNR, entropy, MSE, correlation coefficient (CC), and time taken. We obtain a high PSNR value of 57.72 and a low time requirement of 42 seconds on average. Combining blowfish and certificateless signcryption iTqnto one double encryption scheme that is computationally secure, fast, and easy to implement. It would help push hospitals toward a cost-effective image security environment.

Keywords Medical image processing · Blowfish algorithm · Certificateless signcryption · Image security

1 Introduction

Medical imaging is a vital diagnostic tool for physicians and patients. Images produced by medical imaging systems can detect signs of disease earlier and with much higher accuracy than those seen with the naked eye [1]. Images also help doctors determine the most effective treatment options for their patients, saving time and money. Increasing complexity in the healthcare sector has necessitated healthcare institutions to protect valuable, real-time medical imaging data [2]. A rise in healthcare tourism and favorable tax laws contribute to the demand for healthcare facilities, especially imaging centers [3]. However, traditional medical imaging and artificial intelligence systems are limited in their ability to protect privacy [4, 5]. Currently, many hospitals are developing strategies to provide security and authenticity [6] for patient data, and managing healthcare information systems [7] has become a common practice.

One of the recent advances in healthcare is the double encryption of medical images. The technology uses two different encryption methods, so the security of transmitted images is many times higher than [8]. Modifying medical images poses a more significant threat than simply accessing [9] medical images, which can be dangerous for patients and misleading for doctors. Although today's modern encryption techniques are virtually unbreakable, the threat is that an attacker can access the key or parts of it. There is no way to determine if the data has been tampered with [10], and if verification of the sender's certificate is required, the data must be signed. Due to these shortcomings and increased medical theft, we used the newly developed Signcryption technology to develop a dual encryption scheme focusing on signing and encrypting data at the lowest possible computational cost.

Signcryption is a technique that combines encryption and signature in a single step [11]. It is signing and encryption with dual encryption in medical imaging, protecting sensitive medical data while simultaneously providing the highest quality imaging and image analysis. Signcryption offers better privacy and protection for patients than traditional medical imaging while saving time and money. This method is faster than regular signatures and encryption and

Kishore Babu Nampalle and Shriansh Manhas are contributed equally to this work.

✉ Kishore Babu Nampalle
kbabu89@cs.iitr.ac.in

Shriansh Manhas
201210043@nitdelhi.ac.in

Balasubramanian Raman
bala@cs.iitr.ac.in

[1] Indian Institute of Technology, Roorkee, India

[2] National Institute of Technology, New Delhi, India

Published online: 19 April 2023

saves computation costs. This technique has been modified over time as flaws were found, like forward secrecy, but it is now a robust and well-rounded method.

The following are the contributions of our proposed work.

1. Novel dual encryption technique to implement blowfish, which is computationally uncracked and relatively fast [12], and Signcryption, which performs verification and encryption together.
2. The proposed algorithm reduces the computational cost, getting an algorithm as fast as possible that can provide strong encryption and perform image signing.
3. Several tests are carried out to check for speed of implementation and the quality of the final image is obtained using metrics such as $PSNR$, entropy, and MSE.
4. The implemented algorithm performs extremely fast on all images, with dual encryption, in this case, more quickly than even a single sign and encryption. Image quality values get better as the size and quality of the base image increase.

2 Literature review

The review of previous works shows the feasibility of our novel approach, namely, security of dual encryption, signcryption viability, and other methods used to tackle image security problems. Merkle et al. have shown that multiple encryptions definitively increase security in the case of DES [13] at the cost of time. Dodis et al. furthered research by Zhang et al. to show that Independent multiple encryptions have a better resistance against chosen cipher-text attacks than regular single encryption [14]. However, some combinations of encryption may have lesser strength than assumed. Dai et al. showed that multiple encryptions in an ideal cipher model have better security against key recovery attacks [8], testing extensively for DES. Many proposed techniques [15] handle data confidentiality based on client and server systems for effective medical data management using query models [16]. A few more techniques based on IOT provide security by detecting malware using signaling games [17].

Barbossa et al. first realized the concept of certificateless signcryption [18]. Leveraging TLTS and the Henon Chaotic Map [19], Ali et al. devised an image encryption method. It provided a key exchange mechanism, a single solid signcryption technique employing ECC (Elliptical Curve Cryptography), and chaotic image encryption. The cost of managing public keys is significantly decreased since a certificate is not required for the public key, and there is no issue with a private key exposure. Zhou et al. modified the technique without bi-linear partings [20], dramatically increasing the scheme's speed while maintaining security. It is ideal for cloud security, which is used extensively in medical imaging. A chaotic logistic map is then used to build a key to decode the mixed image.

A modified salp swarm algorithm (SSA) [21] in conjunction with chaotic coupled map (CML) lattices was developed by Selvi et al. This algorithm uses a coupled map lattice to compress and encrypt the image. This CML first generates the number of encrypted images in the updated SSA population. Following initialization, the modified SSA with the whale optimization algorithm (WOA) is employed to reduce calculation time and boost entropy in the image encryption. In their investigation of highly secure medical pictures with subkeys, Shankar et al. found new and innovative medical image encryption. Kamal et al. [22] introduced a novel encryption method that uses image splitting, image scrambling, random permutation, and rotation, including the cases where a few subkeys were provided using chaotic logistic and tent maps [23]. Confusion and diffusion were used to investigate security using the chaotic function. Results were obtained using the Grasshopper optimization technique on PSNR.

Shafique et al. used a cubic logistic map, Discrete Wavelet Transform (DWT), Furthermore, the bit-plane extraction technique [24] encrypts the medical images at the bit level instead of the pixel level. The Proposed approach can decode pixel values of precise images, making it lossless. Avudaiappan et al. used dual encryption [25] using oppositional-based learning to simultaneously increase the image PSNR and security while simultaneously signing and verifying the image.

3 Methodology

3.1 Theoretical contribution

The keys are vulnerable to cryptographic attacks, so we use a novel approach, dual encryption, for additional security. The original image is first rearranged into 64-bit blocks and then subjected to dual encryption, blowfish, and Signcryption. Signcryption is performed using the certificateless signcryption algorithm, significantly reducing computational costs. This strategy improves the security and the quality of the final decrypted image. We thus achieve the maximum possible speed in our algorithm. The Blowfish algorithm is used as the outer encryption scheme. With the Cipher Feedback Mode (CFB) method, we chain the entire image length in bits and then perform the blowfish on each of them individually. Due to CFB, applying for a random number gives additional security against a chosen repeating plain text attack. The creation of encryption keys is crucial because it is responsible for ensuring the legitimacy of

medical images. Random number generation is critical in the quality control of cryptographic primitives.

Due to the susceptibility of keys to cryptographic attacks, we propose a novel strategy, i.e. dual encryption with certificatelessness for added protection.

1. The algorithm can use an adaptive switch between encryption, signature, and signcryption modes. Therefore, it can realize confidentiality and authentication separately or simultaneously, and the total number of keys in the system is significantly reduced.
2. The private keys may be exposed during some periods, but they are not affected in other periods.
3. The scheme does not rely on costly bilinear pairings. Bilinear pairing is a useful tool in the design of cryptography schemes, but the computational cost of a pairing can be almost 20 times that of elliptic curve point multiplication. Therefore, the computational efficiency of our scheme is high

   $$bilinear > O(e^n).$$

4. The scheme supports unbounded time periods. In comparison,in the first key-insulated scheme, the total number of time periods must be given in advance.
5. The scheme supports random-access key update; that is, for any current time period i and any desired time period j,the private key can be updated from

   $$sk_i \; to \; sk_j$$

   in one step.
6. The scheme supports secure key update. We considered that an adversary might break into the user's storage while a key update occurs. In this scenario, a key update exposure from time period i to j is equivalent to key exposures in time periods i and j. Other time periods remain secure.

3.2 Usefulness of proposed algorithm

- Since any license does not cover the Blowfish algorithm, anybody may use it without restriction. It is also a secure, time-tested algorithm with reasonably fast implementation across many platforms.
- Blowfish provides a secure outer layer of encryption, protecting the inner signature and safeguarding the confidentiality of the image.
- Signcryption here is implemented in a certificateless form. It is essentially asymmetric, creating two keys.
- Signature easily tells if any tampering has been performed on the image during designcryption.
- Signcryption is already quicker than the conventional signature + encryption techniques. Making it certificateless makes it even faster, thus solving the issue of computational time.
- High PSNR is achievable due to the usage of optimal keys in blowfish and signcryption.
- Dual Encryption is viable in this case as signature and verification are being done in conjunction with Dual Encryption, with highly optimized speed and fast algorithms.

3.3 Practical contributions

Testing the proposed algorithm on commonly used high and low-resolution medical images, we find a high PSNR and low entropy count. Time taken by the algorithm increases linearly(observed) with the size of the image, which is a good case for a dual encryption algorithm, being competitive with state-of-the-art techniques and practical to run on most encryption devices.

3.4 Dual encryption

Here, we consider the importance of double encryption joining and the driving forces behind the blowfish and Signcryption algorithms. This framework is more secure than the present two-fold encryption in terms of security. There is little risk of plaintext attacks as encryption does not uncover any information on plain medical images in the database. Since the image and keys are encrypted, this provides an extra layer of security without compromising digital signature verification.

3.5 Blowfish algorithm

The key length for Blowfish's 64-bit symmetric block figure ranges from 32 to 448 bits (14 bytes). This method creates a 64-bit cipher block by encrypting 64 bits of plain text. On 32-bit processors, data encryption and decryption were sped up using table query, expansion, modulus, and bitwise selective operations. Every operation consists of XOR on 32-bit cipher blocks.

3.6 Blowfish subkeys

Both the encryption and decryption procedures need 18 sub keys, P[0] through P[17], and the same sub keys are employed in both operations. These sub keys must be configured before the encryption and decryption processes. Four S boxes are created, each of 42 bits. Each S-Box, however, has 256 parts:-

$S_{1\,0}$; ....to $S_{1\,255}$
$S_{2\,0}$; ....to $S_{2\,255}$
$S_{3\,0}$; ....to $S_{3\,255}$
$S_{4\,0}$; ....to $S_{4\,255}$

The first step is to split the key into 32-bit parts and XOR those sections with the P array's comparing sections. The key is rotated, starting from the beginning until all of the P array's components are XORed with key bits if the key is shorter than 576(32 x 18) bits.

3.7 Blowfish operation

3.7.1 Encryption

The image, as shown in Fig. 1, is first encoded into plaintext. The blowfish method divides the unique image bit stream into blocks of the appropriate length. The byte components of the array are encrypted 64 bits at a time from left to right using CBC (cipher block chaining) due to its security over ECB (Electronic code book).

3.7.2 F function

Key length is 448 bits. After the initial XOR operation, the F function 1 is used here, as shown in Fig. 3.

$$F() = (((S_{1,x} + S_{2,x}) \mod 2^{32}) \, XOR \, S_{3,z}) + S_{4,p} \mod 2^{32} \quad (1)$$

3.7.3 Decryption

Decryption process of Blowfish algorithm is same as encryption process but sub keys such as P1, P2..., P18 have been used in the reverse order, as shown in Fig. 2.

3.8 Signcryption

Signcryption is a relatively new signature and encryption technique aiming simultaneously to perform signature and encryption in one concrete step. Generalized Signcryption(GSC) significantly reduces the computational cost of the standard signature and then the encryption method. The proposed Signcryption algorithm consists of eight algorithms and has randomized forward secrecy. The contents of the past images will remain safe even if the sender reveals his private key in the present or future (Fig. 3).

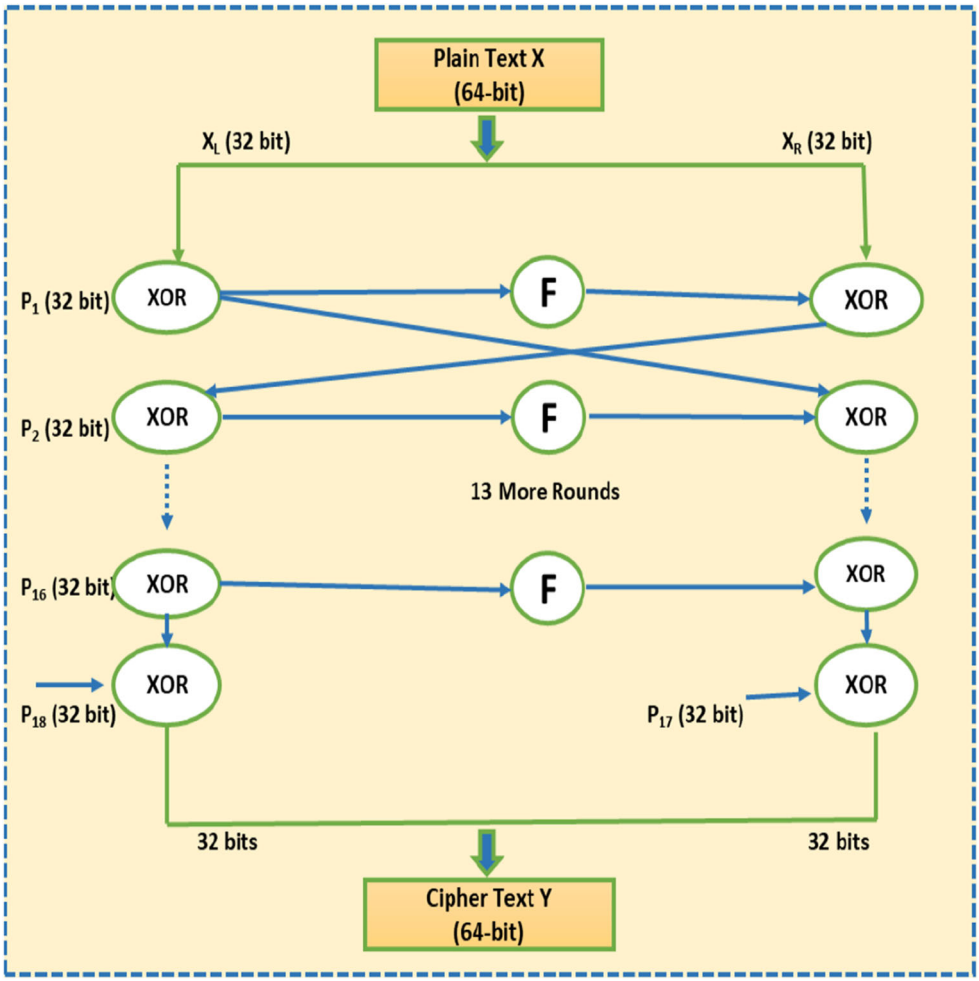


Fig. 1 Blowfish Encryption Algorithm

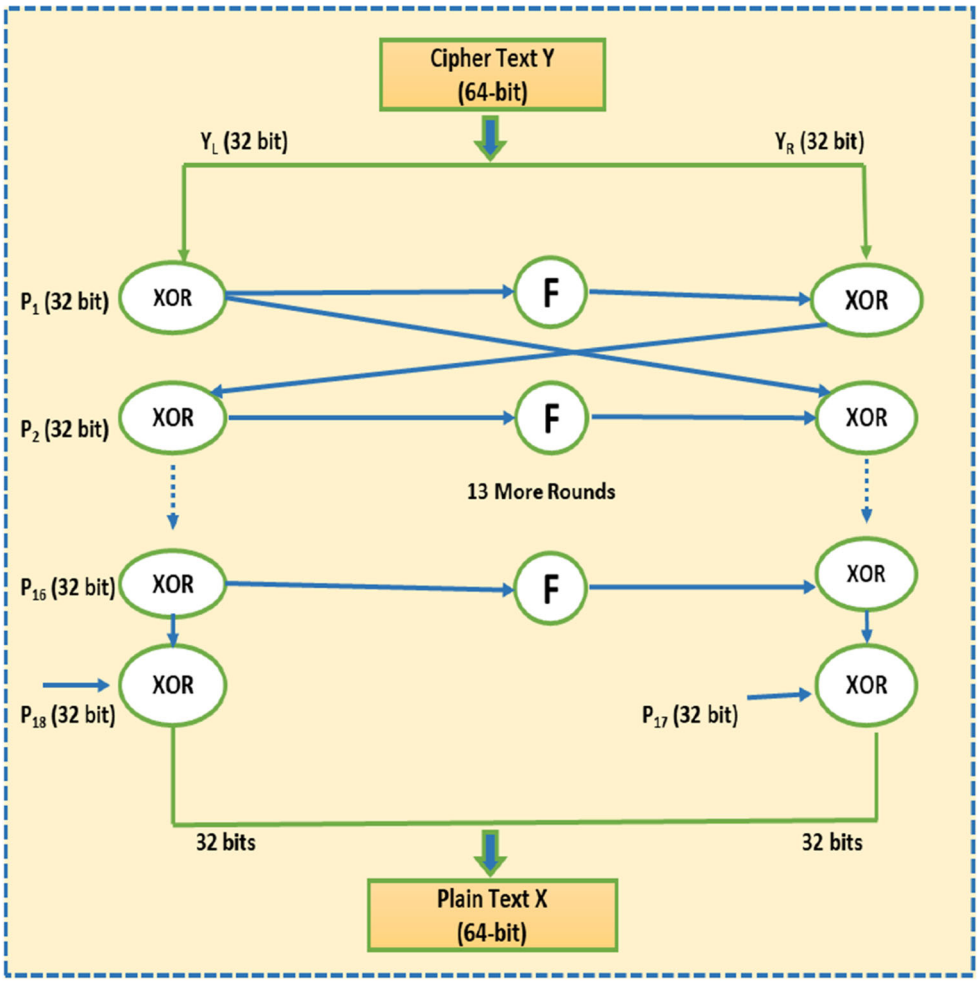


Fig. 2 Blowfish Decryption Algorithm

3.8.1 Certificateless signcryption

We employ a cryptosystem that is certificateless in which the user's private key comprises two components: a partial private key generated by a trustworthy third-party key generation center (KGC) and a user-defined secret value. In light of this, a user's public key is divided into two parts: the public key that corresponds to the secret value and the user's identification information. The cost of managing public keys is significantly decreased because a certificate

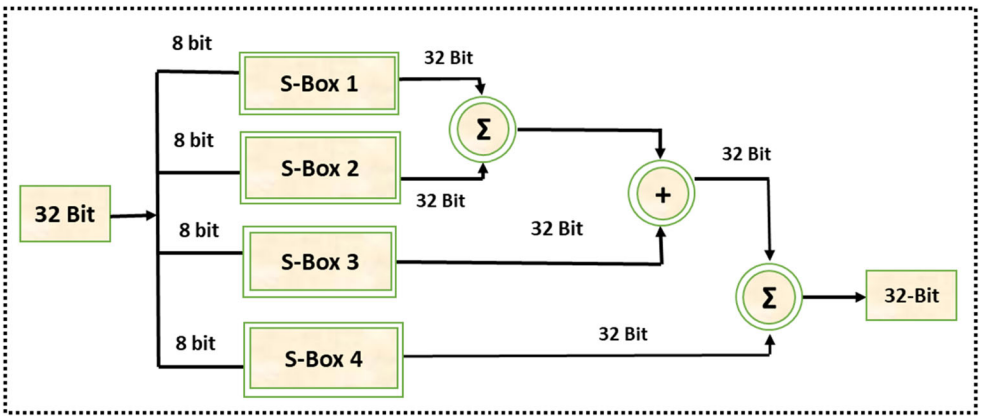


Fig. 3 Blowfish F function Algorithm

is not necessary for the public key. Private key escrow is not problematic because an independent KGC doesn't know the user's secret value.

3.8.2 Algorithm

The following steps of an Algorithm 1 make up the certificateless key-insulated GSC scheme.

3.8.3 Concrete implementation

Complete implementation details are explained below with the help of equations from 2 to the equation 33.

1. **Setup**. The KGC generates the large primes p and q given a security parameter of $1k$. On the finite field F_p, which is the generator of G_1, a strong elliptic curve:-→ $E(F_p)$. The user then chooses $s \epsilon Z^*_q$ at random as the main private key and computes:-

$$P_{pub} = s * p \tag{2}$$

above equation as the main public key. The user chooses seven hash functions:

$H_0, H_1, H_2, H_3, H_4, H_5 :: (0,1)^* \rightarrow Z^*_q$
and $H_6 : (0,1)^* \rightarrow (0,1)^L \cdot Z^*_q$

l represents the message's bit length (m), If the identification ID is null, then the function $f(ID)$ has the value 0; otherwise, it has the value 1. The system public parameters are :

$Params = p, q, G_1, P, E(F_p), P_{pub}, f,$
$H_0, H_1, H_2, H_3, H_4, H_5, H_6.$

2. **Partial-Private-Key-Gen**. Given the user's identification ID, KGC generates $r_{ID} \epsilon Z^*_q$ at random and computes

$$Y_{ID} = r_{ID} * P \tag{3}$$

$$y_{ID} = r_{ID} + s * h_{0,ID} \mod q \tag{4}$$

where $h_{0,ID} = H_0(ID, Y_{ID})$. KGC safely transmits y_{ID} to ID.

3. **User-Key-Gen**. The certification ID user selects $xID \epsilon Z^*_q$ at random to use as hidden value. The public key is determined by using:-

$$X_{ID} = x_{ID} * P \tag{5}$$

4. **Set-Initial-Key**. The certification ID user selects $u_{ID,0}, hk_{ID} \epsilon Z^*_q$ at random and computes

$$U_{ID,0} = u_{ID,0} * P \tag{6}$$

$$T_{ID} = hk_{ID} * P \tag{7}$$

$$h_{1,ID,0} = H_1(ID, Y_{ID}, T_{ID}, 0) \tag{8}$$

$$h_{2,ID} = H_2(ID, Y_{ID}, X_{ID}, T_{ID}) \tag{9}$$

$$h_{3,ID,0} = H_3(ID, Y_{ID}, U_{ID,0}, 0) \tag{10}$$

$$s_{ID,0} = y_{ID} + x_{ID} * h_{2,ID} + u_{ID,0} * h_{3,ID,0} + hk_{ID} + h_{1,ID,0}(\mod q) \tag{11}$$

In time frame 0 The user's period private key is $s_{ID,0}$, while the Assistant private key is hk_{ID}. The user broadcasts $(U_{ID,0}, Y_{ID}, T_{ID})$ and in time frame 0, transmits the Assistant key hk_{ID} and temporary variable $u_{ID,0}$ to the Assistant then finally deleting information from user.

5. **Key-Update-H.** The Assistant selects $u_{ID,t'} \epsilon Z^*_q$ from the user's identification ID, Y_{ID}, old time frame t, and new time frame t'. and computes.

$$U_{ID,t'} = u_{ID,t'} \cdot P \tag{12}$$

$$U_{ID,t} = u_{ID,t} \cdot P \tag{13}$$

$$T_{ID} = hk_{ID} \cdot P \tag{14}$$

$$h_{1,ID,t'} = H_1(ID, Y_{ID}, T_{ID}, t') \tag{15}$$

$$h_{1,ID,t} = H_1(ID, Y_{ID}, T_{ID}, t) \tag{16}$$

$$h_{3,ID,t'} = H_3(ID, Y_{ID}, U_{ID,t'}, t) \tag{17}$$

$$h_{3,ID,t} = H_3(ID, Y_{ID}, U_{ID,t}, t) \tag{18}$$

$$uk_{ID,t,t'} = u_{ID,t'} \cdot h_{3,ID,t'} - u_{ID,t} \cdot h_{3,ID,t} + hk_{ID} \cdot (h_{1,ID,t'} - h_{1,ID,t}) \, mod, q \tag{19}$$

The key value is updated to $uk_{ID,t,t'}, U_{ID,t'}$.
The assistant saves $u_{ID,t'}$ but deletes $u_{ID,t}$.

6. **Key-Update-U**. The user ID updates their period private key from t to t' time, utilizing the update key $(uk_{ID,t,t'}, U_{ID,t'})$ as

$$s_{ID,t'} = s_{ID,t} + uk_{ID,t,t'} \tag{20}$$

The user then transmits $U_{ID,t'}$.

7. **GSC**. Let the time interval be t, the sender's credentials be ID_s, the receiver's credentials be ID_r, and $m \epsilon (0,1)^L$.The sender picks $a_1, a_2 \epsilon Z^*_q$ at random and calculates

$$R_1 = a_1 * P, R_2 = a_2 * P \tag{21}$$

$$h_4 = H_4(m, Y_{ID_s}, R_1, R_2, ID_s, ID_r U_{t,ID_r}, X_{ID_r}, Y_{ID_r}, T_{ID_r}) \tag{22}$$

$$h_5 = H_5(m, U_{t,ID_r}, R_1, R_2, ID_r, X_{ID_r}, Y_{ID_r}, T_{ID_r}) \tag{23}$$

1. $Setup$. The KGC generates a global public parameter Params taking input as $1k$ security parameter and a main private key s.
2. $Partial - Private - Key - Gen$. The KGC also generates a partial private key for the user, D_{ID}, using the user's identification ID, the Params, and the main private key s. and it securely transmits the user their D_{ID}.
3. $User - Key - Gen$. Usually run by the user. It generates a secret value x_{ID} and the user's matching public key PK_{ID} given the Params and the user's identification ID.
4. $Set - Initial - Key$. Usually run by the user. Assistant private key hk_{ID} is generated for the Assistant and a period private key $S_{ID,0}$ is generated for the user in time frame 0 given a user's partial private key D_ID, the Params, their identification ID, and their secret value X_ID. The user then deletes the Assistant's private key from their own possession and delivers it to them through hk_{ID}.
5. $Key - Update - H$. Usually run in the Assistant device. It generates an update key with the following parameters: the previous time frame t and the current time frame t', the user's certification ID, the Params, and the Assistant private key hk_{ID}.
6. $Key - Update - U$. Usually user run. It generates user's own period private key $S_{ID,t}$. given user's certification ID, the Params, the update key $UK_{ID,t,t}$, and the user's own period private key $S_{ID,t'}$.
7. GSC. Usually sender driven IDs . This might even be an unauthorized third party. It generates a GSC ciphertext α provided a sender's credentials ID_s,, a receiver's credentials ID_r, the Params, a message m, a time frame t, and the sender's period private key $SID_{s,t}$. There are three ways to execute this algorithm.
   (a) $Encrypt\ mode$: GSC ciphertext α will be pure encryption if:- IDs is null and IDr is not.
   (b) $Signature\ mode$: GSC ciphertext α will be pure signature::- if ID_r is null and ID_s is not.
   (c) $Signcryption\ mode$: GSC ciphertext α is a signcryption if:- both ID_s and ID_r are not null
8. $Un - GSC$. Run usually by the receiver ID_r or another person in signature mode. This might even be an unauthorized third party. Given a the params, a ciphertext from GSC α, a time frame t, the senders certification ID_s, a receivers certification ID_r, and the receivers period private key $S_{IDr,t}$. In encryption or Signcryption mode, it recovers the message m; in signature mode, it returns true; in all other cases, it returns $\perp$, signifying a failed decryption or an invalid signature. There also are three ways to execute this algorithm.
   (a) $Decryption\ mode$: Decrypts the cipher if ID_s is null and ID_r is not
   (b) $Signature\ mode$: ciphertext from GSC α signature can be verified. if ID_r is null and ID_s is not.
   (c) $Un - Signcryption\ mode$: Performs reverse signcryption if neither ID_s nor ID_r is null.

Algorithm 1 Proposed Algorithm

$$u = f(ID_s) * s_{t,ID_s} * h_4 + a_1 * h_5 + a_2 (\mod q) \quad (24)$$

The user then calculates:

$$h_{0,ID_r} = H_0(ID_r, Y_{ID_r}) \quad (25)$$

$$h_{1,t,ID_r} = H_1(ID_r, Y_{ID_r}, T_{ID_r}, t) \quad (26)$$

$$h_{2,ID_r} = H_2(ID_r, Y_{ID_r}, X_{ID_r}, T_{ID_r}) \quad (27)$$

$$h_{3,t,ID_r} = H_3(ID_r, Y_{ID_r}, U_{t,ID_r}, t) \quad (28)$$

$$V = a_1 * (Y_{ID_r} + P_{pub} * h_{0,ID_r} + X_{ID_r} * h_{2,ID_r} + U_{t,ID_r} * h_{3,t,ID_r} + T_{ID_r} * h_{1,t,ID_r} \quad (29)$$

$$h_6 = f(ID_r) * H_6(ID_s, ID_r, U_{t,ID_s}, X_{ID_s}, Y_{ID_s}, T_{ID_s}, R_1, t, V) \quad (30)$$

$$c = (m \parallel u) \oplus h_6 \quad (31)$$

The equation finally outputs to :

$$(t, \alpha) = (t, (R_1, R_2, c, tag))$$

8. **Un-GSC.** Final cipher formed is:-

$$(t, \alpha) = (t, (R_1, R_2, c, tag))$$

the nature of the signcryption applied is given by the $tag\ \alpha = (R_1, R_2, c)$ is the ciphertext for signcryption. Receiver ID_r compute the following:

$$V = s_{ID_r} * R_1 \quad (32)$$

$$h_6 = H_6(ID_s, ID_r, U_{t,ID_s}, T_{ID_s}, X_{ID_s}, Y_{ID_s}, R_1, t, V) \quad (33)$$

Upon decryption correlates with eq (31) as:

$$(u \parallel m) = c \oplus h_6$$

Recalculate with respect to new decryption values (25-28)(22-23)

$$h_{0,ID_s} = H_0(ID_s, Y_{ID_s})$$
$$h_{1,t,ID_s} = H_1(ID_s, Y_{ID_s}, T_{ID_s}, t)$$
$$h_{2,ID_s} = H_2(ID_s, Y_{ID_s}, X_{ID_s}, T_{ID_s})$$
$$h_{3,t,ID_s} = H_3(ID_s, Y_{ID_s}, U_{t,ID_s}, t)$$
$$h_4 = H_4(m, R_1, R_2, Y_{ID_s}, ID_s, ID_r, U_{t,ID_r}, X_{ID_r}, Y_{ID_r}, T_{ID_r})$$
$$h_5 = H_5(m, ID_r, R_1, R_2, U_{t,ID_r}, X_{ID_r}, Y_{ID_r}, T_{ID_r})$$

3.8.4 Authentication

For authentication of the given message, if:

$$u * P = (Y_{ID_s} + P_{pub} * h_{0,ID_s} + X_{ID_s} * h_{2,ID_s} + U_{t,ID_s} * h_{3,t,ID_s} + T_{ID_s} * h_{1,t,ID_s} * h4 + R_1 * h_5 + R_2) \quad (34)$$

above given equation (34) holds true then the message is verified

4 Results analysis

4.1 Dataset

The dataset containing the images of brain tumors, chest, heart, skin, and skull, which is used to evaluate the performance of the proposed approach, has been extracted from the Kaggle website, and those images, as shown in Table 1 are available from the following links.

- https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection
- https://www.kaggle.com/datasets/mohamedhanyyy/chest-ctscan-images
- https://www.kaggle.com/datasets/adarshsng/heart-mri-image-dataset-left-atrial-segmentation
- https://www.kaggle.com/datasets/shubhamgoel27/dermnet
- https://www.kaggle.com/competitions/open-images-2019-instance-segmentation/data?select=test

Additional data averages have been calculated across multiple images of differnet categories, images which most closely reflect the average have been displayed.

4.2 Results

This proposed dual encryption scheme has been implemented using Python 3.9.7 with an i5 processor, 8GB of RAM, and a 4GB graphics card. Our process considers medical images collected from kaggle datasets for their security analysis. The final image obtained is then analyzed for performance. In the total procedure for encryption, the keys are generated using the certificateless signcryption technique. The effective evaluation of the proposed model is done using metrics such as Peak Signal to Noise Ratio (36), Entropy (38), CC (37), and Mean Squared Error (35).

$$RMSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2} \quad (35)$$

Table 1 Results analysis using input medical images and performance metrics.

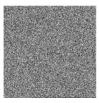
Original Image	Encrypt Image	Decrypt Image	PSNR	MSE	CC	Entropy	Time
			61.3310	0.04786	0.9999 8866	21.9886	118.637 sec
			57.8084	0.10770	0.9999 854	18.5797	42.5013 sec
			56.7161	0.13850	0.9999 8106	18.5235	61.1070 sec
			52.7287	0.34690	0.9999 7572	15.61622	10.4385 sec
			60.0265	0.06462	0.9999 9285	18.6408	37.6659 sec

Table 2 Comparison with other single encryption algorithms

	Blowfish	DES	TDES	AES
PSNR	44.782	42.543	42.643	45.2258
Entropy	12.683	11.453	14.414	12.512
Time Taken	7.02 sec	6.41 sec	15.31 sec	7.38 sec

$$PSNR = 20\log(255/RMSE) \tag{36}$$

$$CC = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2(y_i - \overline{y})^2}} \tag{37}$$

$$Entropy = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \tag{38}$$

With our proposed dual encryption scheme, entropy maximises, with the avg. value being:

18.6698105110762074

and thus the avg. PSNR is found out to be:-

57.722195665575656

The image maintains forward secrecy, and the recovered image is a distinct image similar to the original without any degradation in quality. This approach minimizes the computational exertion due to the application of certificateless signcryption and the required calculation time with the application of blowfish. Table 1 shows the performance results for the proposed model (dual encryption), including PSNR, MSE, CC, Entropy, and time taken. Here, the Optimal algorithm is utilized for encoding the image, allowing secure transmission. The image can be securely transmitted post-dual encryption. In the final stage, a dual decryption algorithm is used to recover the original image. Double security is acquired amid transmission. After receiving the transmission, the other user performs designcryption and decryption using the blowfish decryption algorithm. We then finally obtain a unique image. As per the data, the cerebral image offers PSNR estimates of proposed demonstrations, from Tables 2, 3, and 4, of 61.33 and MSE with the least value of 0.0478. Additionally, various parameters increase security in all test images compared to current methods. There is a trade-off between the image's highest available and highest entropy values. There is a trade-off between image size/quality and time taken. The better quality image takes greater time for signcryption, possibly due to the need for optimization of multiple blocks. However, such a problem is minimized on a dedicated machine. Meantime is not a good measure here due to outliers due to extremely large image size. Median time thus comes out to be 42.5013234615325 seconds.

Processing speeds with respect to other algorithms are favorable, and it outspeeds a single sign and encryption algorithm while staying competitive with different dual encryptions. Blowfish is by no means the only algorithm capable of being used with signcryption, but its speed edges out the competition. The bulk of the time is taken by signcrypion algorithm, and if it was not certificateless, it would be even slower.

5 Discussion

5.1 Possible attacks

5.1.1 Size constraint

While blowfish alone is vulnerable to birthday attacks (sweet32) on large files ($> 4GB$) due to its 64-bit block size, images rarely exceed 30 MB in size, making collisions and most known plaintext attacks unfeasible. Moreover, such an attack would be helpful only if signcryption is cracked.

5.1.2 Differential attack

The differential attack depends on guessing information about an image by making a slight change in the plain image and encrypting both images using the same algorithm. We compare both images to detect a correlation between the plain image and the encrypted image. For the algorithm to

Table 3 Comparison of results with other dual encryption algorithms

	Proposed	DES signc	Sign and Enc	AES signc
PSNR	57.7222	53.7813	52.543	57.2258
Entropy	18.669	17.912	18.414	18.312
Time Taken	42.501 sec	52.524 sec	61.313 sec	43.824 sec

Table 4 Average results of various test images from the algorithm

	Heart	Brain	Skin	Lungs	Spine
PSNR	60.4143	55.567	52.245	58.273	59.713
Entropy	18.349	17.997	17.478	18.184	18.373
Time Taken	44.121 sec	53.821 sec	57.826 sec	50.613 sec	53.544 sec

be immune to differential attacks, any slight change in the plain image should produce a different encrypted image. To assess the performance of an algorithm, the Number of Pixels Change Rate (NPCR), and the Unified Average Changing Intensity (UACI) used. The NPCR and UACI are calculated as follows in the equations from 39 to 41.

$$NPCR = \left(\frac{1}{MN}\right)\sum_{i=1}^{M}\sum_{i=1}^{N}(D(i,j)\cdot 100\%) \tag{39}$$

$$D(i,j) = \begin{cases} 0 & \text{if } E_1(i,j) = E_2(i,j) \\ 1 & \text{if } E_1(i,j) \neq E_2(i,j) \end{cases} \tag{40}$$

$$UACI = \left(\frac{1}{MN}\right)\sum_{i=1}^{M}\sum_{i=1}^{N}\frac{|E_1(i,j) - E_2(i,j)|}{255}\cdot 100\% \tag{41}$$

The symbols E1 and E2 refer to two encrypted images from the plain image and the modified image (made by changing one pixel in the plain image). The image width is M. Its height is N. Here. This study shows the proposed algorithm's effectiveness in resisting differential attacks by recording the NPCR and UACI values between the two encrypted images. All values in Table 5 are close to their ideal values. Table 6 shows a comparison between our algorithm and other image encryption algorithms. The results show that our proposed algorithm can resist differential attacks.

5.1.3 Meet in the middle attack

We consider an unauthorized user saying "M" to insert himself in communication between the patient and healthcare authority. He intercepts (C,s, G') sent by the patient. Then produces his secret key and uses it for the generation of cipher image (C'), signature (s'), and authentication parameter (G"). He alters patient sent parameters (C,s, G') by his generated tuple (C', s', G") and sends it to the healthcare authority. The authority works to generate secret key K^* to decrypt received C'. The hidden image cannot be revealed with this key. Additionally, the signcryption technique will not be able to validate the attacker's signature. Therefore, it is clear that the man-in-the-middle attack is not appropriate for the proposed scheme.

5.1.4 Poor implementation

As an adequately implemented signcryption scheme is secure due to the nature of ECC, the risk lies in the key escrow problem, which certificatelessness solves. Both encryption techniques are thus secure in a given scenario. Therefore, the only real threat is if someone carelessly stores images together while giving away the key. While this seems unlikely, poor implementation and human error give rise to many problems.

5.2 Speed

Blowfish is among the fastest encryption algorithms that are still computationally secure. A notable substitution for blowfish can be AES, which is slower algorithmically, but due to most modern systems having hardware AES acceleration, AES manages to run faster. It is thus a valid substitution, having even a larger block size. Certificateless signcryption is, however, the fastest way to perform signcryption and is significantly faster than a standard signature and then encryption.

Table 5 NPCR and UACI performances for image1 (skin), Image2 (Brain) and image 3 (Heart)

Test Image	NPCR	UACI
Image1	99.7024	33.501
Image2	99.6284	33.321
Image3	99.7443	33.381

Table 6 Comparison of results using the metrics NPCR and UACI performance comparison with other models

Method	NPCR	UACI
Proposed	99.6824	33.348
Ali and Ali [19]	99.6010	33.438
Kamal et al. [22]	99.7443	33.501

Blowfish algorithm costs $O(n)$ time complexity and $O(1)$ space complexity, N being number of partitions, as given below 42.

$$N = \frac{\text{Total image size}}{\text{Cipher block size}} \tag{42}$$

Certificateless Signcryption consists of multiple exponential time algorithms, which brings the overall complexity to $O(e^n)$. A different implementation of Signcryption has similar complexities with varying coefficients. Thus a speed-based comparison of a standard system is more beneficial.

5.3 Noise

Also, we have considered one case study, which includes noise. Figure 4 demonstrates the results of the proposed model using input images with noise and without noise.

6 Conclusion

This work creates a security model for medical images using a dual encryption process, including the blowfish and signcryption algorithms. Applying a certificateless-based signcryption approach optimizes the entire calculation process, making signcyption much faster. The suggested method's performance is assessed using PSNR, entropy, CC, and MSE. Entropy is maximized in the dual encryption approach we recommend, with a value of 18.66 and an average PSNR of 57.72. As a result, the image's confidentiality is maintained over time and the final image obtained is almost similar without degrading the image's quality in any way. Due to the use of the certificateless strategy and Blowfish algorithm, this solution significantly reduces the computing cost and the requisite calculation time.

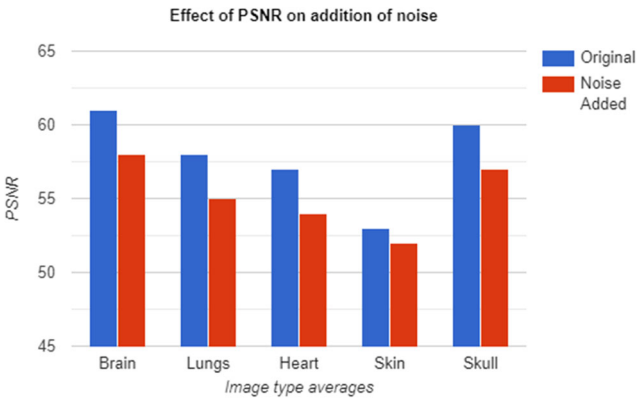


Fig. 4 Performance analysis of proposed model with and without noise

Acknowledgements The research was funded by the Ministry of Education, Government of India, using a grant from Indian Institute of Technology Roorkee (Grant No: OH-31-24-200-428)

Data Availability The datasets analysed during the current study are publicly available and can be accessed using https://www.kaggle.com/

Declarations

Ethics approval and consent to participate This article does not contain any of the authors' research involving humans or animals.

Conflict of Interests There were no potential conflicts of interests/competing interests revealed by the authors.

References

1. Kermany DS, Goldbaum M, Cai W, Valentim CC, Liang H, Baxter SL, McKeown A, Yang G, Wu X, Yan F, et al. (2018) Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 172(5):1122–1131
2. Siyal AA, Junejo AZ, Zawish M, Ahmed K, Khalil A, Soursou G (2019) Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives. Cryptography 3(1):3
3. Azam MA, Khan KB, Salahuddin S, Rehman E, Khan SA, Khan MA, Kadry S, Gandomi AH (2022) A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics. Comput Biol Med 144:105253
4. Razzak MI, Naz S, Zaib A (2018) Deep learning for medical image processing: overview, challenges and the future. Classification in BioApps 26:323–350
5. Kaissis GA, Makowski MR, Rùckert D, Braren RF (2020) Secure, privacy-preserving and federated machine learning in medical imaging. Nature Mach Intell 2(6):305–311
6. Thapa C, Camtepe S (2021) Precision health data: Requirements, challenges and existing techniques for data security and privacy. Comput Biol Med 129:104130
7. Lv Z, Piccialli F (2021) The security of medical data on internet based on differential privacy technology. ACM Trans Internet Technol 21(3):1–18
8. Dai Y, Lee J, Mennink B, Steinberger J (2014) The security of multiple encryption in the ideal cipher model. In: Annual Cryptology Conference, pp. 20–38. Springer
9. Jiao S, Lei T, Gao Y, Xie Z, Yuan X (2019) Known-plaintext attack and ciphertext-only attack for encrypted single-pixel imaging. IEEE Access 7:119557–119565
10. Noh J, Kwon Y, Son J, Cho S (2022) Blockchain-based one-time authentication for secure v2x communication against insiders and authority compromise attacks. IEEE Internet Things J
11. Zheng Y (1997) Digital signcryption or how to achieve cost (signature and encryption) cost (signature)+ cost (encryption). In: Annual International Cryptology Conference, pp. 165–179. Springer
12. Kofahi NA, Al-Somani T, Al-Zamil K (2003) Performance evaluation of three encryption/decryption algorithms. In: 2003 46th Midwest Symposium on Circuits and Systems, vol. 2, pp. 790–793. IEEE
13. Du J, Sheng L, Xu Y, Chen Q, Gu C, Li M, Zhang SX-A (2021) Printable off–on thermoswitchable fluorescent materials

for programmable thermally controlled full-color displays and multiple encryption. Adv Mater 33(20):2008055
14. Dodis Y, Katz J (2005) Chosen-ciphertext security of multiple encryption. In: Theory of Cryptography Conference, pp. 188–209. Springer
15. Wu Z, Xuan S, Xie J, Lin C, Lu C (2022) How to ensure the confidentiality of electronic medical records on the cloud: A technical perspective. Comput Biol Med 105726:147
16. Wu Z, Shen S, Lian X, Su X, Chen E (2020) A dummy-based user privacy protection approach for text information retrieval. Knowl-Based Syst 105679:195
17. Shen Y, Shen S, Wu Z, Zhou H, Yu S (2022) Signaling game-based availability assessment for edge computing-assisted iot systems with malware dissemination. J Inf Security and Applications 103140:66
18. Qu Y, Zeng J (2022) Certificateless proxy signcryption in the standard model for a uav network. IEEE Internet Things J
19. Ali TS, Ali R (2020) A novel medical image signcryption scheme using tlts and henon chaotic map. IEEE Access 8:71974–71992
20. Zhou C, Zhao Z, Zhou W, Mei Y (2017) Certificateless key-insulated generalized signcryption scheme without bilinear pairings. Security and Communication Networks 2017(3):1–17
21. Selvi CT, Amudha J, Sudhakar R (2021) A modified salp swarm algorithm (ssa) combined with a chaotic coupled map lattices (cml) approach for the secured encryption and compression of medical images during data transmission. Biomed Signal Process Control 66:102465
22. Kamal ST, Hosny KM, Elgindy TM, Darwish MM, Fouda MM (2021) A new image encryption algorithm for grey and color medical images. IEEE Access 9:37855–37865
23. Shankar K, Elhoseny M, Chelvi ED, Lakshmanaprabu S, Wu W (2018) An efficient optimal key based chaos function for medical image security. IEEE Access 6:77145–77154
24. Shafique A, Ahmed J, Rehman MU, Hazzazi MM (2021) Noise-resistant image encryption scheme for medical images in the chaos and wavelet domain. IEEE Access 9:59108–59130
25. Avudaiappan T, Balasubramanian R, Pandiyan SS, Saravanan M, Lakshmanaprabu S, Shankar K (2018) Medical image security using dual encryption with oppositional based optimization algorithm. J Med Syst 42(11):1–11

Kishore Babu Nampalle is a senior researcher and currently pursuing the Ph.D. in the Computer Science and Engineering department at Indian Institute of Technology Roorkee, India. He received the M.Tech. and B.Tech. Degrees in computer science and engineering from Indian Institute of Technology Roorkee and Sree Vidyanikethan Engineering College, India, respectively. His academic prowess has been recognized through a litany of prestigious awards and scholarships, including the Prathibha Award in 2006, Merit Scholarships and exceptional performance on the EAMCET and GATE tests. His research has garnered widespread recognition and support, including funding from esteemed institutions such as the Ministry of India, DORA IITR, and Microsoft Research. His contributions to the field have been acknowledged through numerous publications in reputable journals and conferences, cementing my position as a trailblazing researcher in the field. In addition to his research achievements, he possess extensive teaching experience and have had the privilege of mentoring numerous graduate and postgraduate students, imparting my knowledge and fostering their academic growth.

Mr. Shriansh Manhas is currently pursuing B. Tech in Computer Science and Engineering from the National Institute of Technology, Delhi. His areas of interests include Computer Networks, Computer Vision, Machine Learning and Object Recognition.

Balasubramanian Raman received the Ph.D. degree from the Indian Institute of Technology, Madras, India in 2001. He has served as a Guest Scientist at the International Centre for Theoretical Physics Trieste, Italy, a Post-Doctoral Fellow at the University of Missouri, Columbia, USA, a Post-Doctoral Associate at the State University of New Jersey, USA, and a Lecturer at the Birla Institute of Technology and Science, Pilani, India and a Visiting Professor at the University of Windsor, CANADA. He has been with the Indian Institute of Technology, Roorkee, India, since 2004, where he currently works as a professor in the Department of Computer Science and Engineering. He is the recipient of the BOYSCAST fellowship, awarded by the Department of Science and Technology, India. He has supervised 30 Ph.D. students (11 on-going), 146 master's Dissertations, and 30 undergraduate projects. So far, he has delivered 20 Industrial R&D projects (5 on-going) and published 142 International Journals, 73 Conference Proceedings, and 9 Book Chapters. His research areas include Machine Learning, Pattern Recognition, Image and Video Processing, Medical Imaging, Computer Vision, Activity Recognition, Affective Computing, Privacy-Preserving Computing, and Data Security.