International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

Secure-Medishare: A Comprehensive Secure Medical Data-Sharing System Using Blockchain, Watermarking, Steganography, And Optimized Hybrid Cryptography

Mrs. Arti Krushnarao Walzade
Ph.D. Scholar
Computer Science and Engineering
Dr. A. P. J. Abdul Kalam University, Indore MP.
artips15@gmail.com

Dr. Pradnya Ashish Vikhar
Research Supervisor
Computer Science and Engineering
Dr. A. P. J. Abdul Kalam University, Indore MP.
pradnyav123@gmail.com

***Abstract*—** Medical data plays a crucial role in healthcare, enabling accurate diagnosis, treatment planning, and research. However, the secure sharing of sensitive medical data and images remains a significant challenge. Existing techniques often fall short in terms of protecting data integrity, confidentiality, and authenticity. To address these limitations, this paper introduces Secure-Medishare, a novel secure medical data-sharing system that integrates blockchain technology, watermarking, steganography, and enhanced cryptography. The proposed Secure-Medishare system aims to provide robust security mechanisms for medical data sharing. Unlike centralized systems, which are susceptible to single points of failure and unauthorized access, Secure-Medishare utilizes blockchain technology to ensure decentralized and tamper-resistant storage and sharing of medical data. Secure-Medishare employs watermarking for data integrity and authentication and steganography for confidential transmission of metadata, ensuring authenticity, privacy, and confidentiality of medical data. Furthermore, an optimized hybrid cryptography technique is implemented to secure the transmission and storage of medical data, safeguarding confidentiality and privacy. Secure-Medishare offers several advantages over existing techniques. It provides enhanced security and privacy protection, efficient data sharing and retrieval, and improved trust among healthcare providers. The system ensures the integrity and authenticity of medical data, preventing unauthorized modifications or tampering. Additionally, the decentralized nature of blockchain technology reduces the risk of data breaches and single points of failure. Experimental results show that Secure-Medishare generates hashes quickly, taking only 65 milliseconds for 100 blocks. Optimized hybrid cryptography used in Secure-Medishare also outperforms other cryptography combinations, with encryption and decryption times of 5.635 seconds for 96-bit data. These findings highlight the efficiency and effectiveness of Secure-Medishare for secure medical data and image sharing. The experimental evaluation confirms that Secure-Medishare is a reliable and robust solution for secure medical data sharing in healthcare environments.

Keywords- Medical Data, Secure-Medishare, Watermarking, Steganography , Cryptography.

I. INTRODUCTION

Medical data is of paramount importance in the field of healthcare as it assumes a pivotal role in facilitating precise diagnosis, comprehensive treatment planning, and groundbreaking research [1]. This invaluable resource encompasses a diverse range of information, comprising not only textual data but also medical images. The amalgamation of these different data types empowers medical professionals to gain a comprehensive understanding of patients' health conditions, enabling them to make informed decisions and deliver personalized care.

Textual data forms a crucial component of medical records, capturing vital patient information such as medical history, symptoms, laboratory test results, and treatment details. This wealth of textual data provides healthcare practitioners with a comprehensive overview of a patient's health journey, allowing them to track progress, identify patterns, and make evidence-based decisions. Electronic health records (EHRs) have revolutionized the storage and accessibility of textual medical data, making it easier for healthcare providers to access and share patient information securely, leading to improved coordination of care and patient safety [2].

In addition to textual data, medical imaging plays a pivotal role in healthcare by providing visual representations of internal structures and organs [3]. Technologies such as X-rays, magnetic resonance imaging (MRI), computed tomography (CT), and ultrasound generate detailed images that aid in the diagnosis and treatment of various medical

conditions. These images offer healthcare professionals valuable insights into a patient's anatomy, detecting abnormalities, assessing the progression of diseases, and monitoring the effectiveness of treatments. Medical imaging data facilitates collaboration among healthcare providers and serves as a valuable educational tool for training future medical professionals.

The integration of textual data and medical images has proven to be particularly valuable in healthcare research and development. By combining large datasets from diverse sources, researchers can analyze medical information on a broader scale, identify correlations, and uncover new insights. This data-driven approach enhances our understanding of diseases, their causes, and potential treatment strategies, leading to the discovery of innovative therapies and improved patient outcomes. Furthermore, the aggregation of medical data from numerous sources enables the development of predictive models and artificial intelligence algorithms, which have the potential to revolutionize healthcare by facilitating early detection, personalized medicine, and more efficient healthcare delivery. As the healthcare sector embraces digital technology, the sharing of medical data among healthcare providers has become indispensable for collaborative patient care. However, ensuring the security and privacy of sensitive medical data remains a significant challenge. While the sharing of medical data is crucial for collaborative research, patient care coordination, and advancements in healthcare, many existing techniques used for this purpose have inherent limitations when it comes to safeguarding data integrity, confidentiality, and authenticity [4], [5]. These limitations pose significant challenges and raise concerns regarding the protection of sensitive patient information.

Data integrity refers to the assurance that medical data remains accurate, complete, and unaltered throughout its lifecycle. Unfortunately, existing techniques for medical data sharing often lack robust mechanisms to ensure data integrity. Without proper safeguards, data can be susceptible to unauthorized modifications, corruption, or tampering, leading to inaccurate diagnoses, compromised patient safety, and potentially harmful treatment decisions. Maintaining data integrity is crucial for preserving the reliability and trustworthiness of medical information.

Confidentiality is another critical aspect of medical data sharing. Patient health records contain highly sensitive information, including personal identifiers, medical conditions, treatment history, and genetic data. Protecting the confidentiality of this data is essential to maintain patient privacy and comply with legal and ethical standards. However, existing techniques may fall short of adequately safeguarding data confidentiality, potentially exposing sensitive information to unauthorized access, data breaches, or inadvertent disclosures. The consequences of breaches in confidentiality can be severe, leading to potential harm, discrimination, or misuse of personal health information.

Authenticity refers to the assurance that medical data originates from a trusted and verified source. Ensuring the authenticity of shared medical data is crucial to prevent unauthorized access, data manipulation, or the inclusion of fraudulent information. Existing techniques may lack robust mechanisms for verifying the identity and source of the data, making it difficult to establish trust in the shared information. Without adequate authentication measures, there is a risk of compromised data integrity and trustworthiness, impacting the reliability of medical decisions and research outcomes.

Medical data comprises a wide range of information, including patient records, test results, medical images, and clinical notes. These data points are essential for healthcare professionals to make informed decisions about patient care. On the other hand, medical images, such as X-rays, MRI scans, and CT scans, provide visual representations of a patient's condition, aiding in accurate diagnoses and treatment planning.

The need for secure medical data sharing arises from the collaborative nature of healthcare. Effective patient care often involves multiple healthcare providers working together, including primary care physicians, specialists, nurses, and pharmacists. Access to comprehensive and up-to-date medical data is crucial for these professionals to provide quality care and make informed decisions.

Existing techniques for medical data sharing rely on centralized systems, where data is stored and managed in a single location. However, these centralized systems present vulnerabilities, such as single points of failure and unauthorized access risks. Moreover, traditional encryption algorithms used to protect medical data may not be sufficient to withstand advanced attacks, such as data tampering or unauthorized data extraction. These limitations highlight the need for an advanced and comprehensive security system like SECURE-MEDISHARE.

The proposed SECURE-MEDISHARE system addresses the limitations of existing techniques by providing robust security mechanisms for medical data sharing. It integrates blockchain technology, watermarking, steganography, and optimized hybrid cryptography to ensure the integrity, confidentiality, and authenticity of medical data.

The SECURE-MEDISHARE system utilizes blockchain technology to enable decentralized and tamper-resistant storage and sharing of medical data. By leveraging the distributed ledger capabilities of blockchain, it eliminates the risks associated with centralized systems, reducing the likelihood of data breaches and unauthorized access.

Watermarking and steganography techniques are employed in the SECURE-MEDISHARE system to enhance data integrity, authentication, and medical image description hiding. Watermarking is used to embed the owner's name or other identifying information within medical images. This embedded watermark serves as a digital signature, enabling the verification of image ownership and ensuring data

integrity. On the other hand, steganography is utilized to hide medical image descriptions within the images themselves. This technique enables the confidential transmission of important metadata or patient-specific information, enhancing privacy and preventing unauthorized access to sensitive medical details. Together, these techniques contribute to the comprehensive security measures of the SECURE-MEDISHARE system, protecting the authenticity and confidentiality of medical data during sharing and storage. This additional layer of data integrity and authentication enhances the trustworthiness of medical images and prevents unauthorized modifications or tampering.

An optimized hybrid cryptography technique is implemented in SECURE-MEDISHARE to secure the transmission and storage of medical data. These algorithms provide robust encryption and decryption capabilities, safeguarding the confidentiality and privacy of sensitive medical information.

The proposed SECURE-MEDISHARE system offers several advantages over existing techniques. It provides enhanced security and privacy protection, efficient data sharing and retrieval, and improved trust among healthcare providers. The decentralized nature of blockchain technology reduces the risk of data breaches and single points of failure, ensuring the availability and integrity of medical data.

The contributions of this paper are as follows:

- Integration of blockchain, watermarking, steganography, and optimized hybrid cryptography in the SECURE-MEDISHARE system.
- Development of a decentralized and tamper-resistant storage and sharing mechanism for medical data.
- Enhancement of data integrity and authentication through the use of watermarking and steganography techniques.
- Implementation of advanced cryptography algorithms for secure transmission and storage of medical data.

This study aims to propose and evaluate the SECURE-MEDISHARE system as a comprehensive and secure solution for medical data sharing in healthcare environments. The paper is organized as follows: Section 2 provides a detailed review of existing medical data-sharing techniques and their limitations. Section 3 presents the methodology and design of the SECURE-MEDISHARE system. Section 4 describes the experimental setup and presents the results and analysis. Finally, Section 5 concludes the paper.

2 Related works:

This section focuses on security in the context of medical image sharing. The selected studies address various aspects of secure and privacy-preserving image retrieval, patient-centric image management, EHR sharing, and medical image encryption. These papers highlight the importance of protecting patient privacy, ensuring data security, and improving the efficiency of healthcare information exchange.

Chen et al. [6] propose a novel approach for the secure sharing of EHRs using blockchain technology and searchable encryption. They highlight that while blockchain ensures secure and transparent record keeping, it falls short in providing the same level of privacy protection as searchable encryption. The authors emphasize the significance of secure sharing mechanisms for EHRs to enhance collaboration among healthcare providers and improve patient outcomes. They introduce searchable encryption as a means to securely store and query data while safeguarding patient privacy and enabling efficient information sharing. The authors present the system architecture for blockchain-based searchable encryption (BBSE) of EHRs, designed to maintain privacy, confidentiality, and integrity while enabling authorized parties to securely search and retrieve specific data elements. Experimental results validate the feasibility and effectiveness of the BBSE system, demonstrating secure storage and retrieval of EHRs with minimal overhead. A security analysis confirms its resilience against common attacks and threats. The authors conclude by discussing limitations and future directions, highlighting the need to address potential vulnerabilities in the underlying blockchain infrastructure and suggesting the integration of artificial intelligence and machine learning techniques for improved data retrieval efficiency and accuracy.

Patel et al. [7] proposed a blockchain-based framework for the secure sharing of medical imaging data. The author emphasizes the importance of secure data sharing in healthcare, particularly in the field of medical imaging, and introduces blockchain consensus as a solution to address privacy, integrity, and accessibility challenges. The framework utilizes blockchain's decentralized and transparent nature, incorporating data encryption, decentralized storage, access control mechanisms, and smart contracts to ensure secure and auditable sharing while preserving patient privacy. The author discusses the benefits, including improved data integrity, reduced reliance on centralized intermediaries, and enhanced interoperability among healthcare providers. A case study validates the framework's effectiveness, demonstrating enhanced security, privacy, and data integrity in medical imaging data sharing.

Wang et al. [8] proposed a framework for secure and privacy-preserving sharing of EHRs using cloud technology and consortium blockchain. They emphasize the importance of security and privacy in EHR sharing and introduce consortium blockchain as a solution. The framework utilizes cloud storage, blockchain-based data sharing, and access control mechanisms to enable secure sharing while preserving patient privacy. The author discusses the benefits, such as enhanced data security and improved interoperability, and validates the framework's effectiveness through experimental results. In conclusion, the author contributes a practical solution for secure EHR sharing and highlights the potential benefits of consortium blockchain in healthcare.

Shen et al. [9] contribute to the field by proposing a blockchain-based approach for privacy-preserving image retrieval in medical Internet of Things (IoT) systems. The authors aim to address the privacy concerns associated with medical image sharing and retrieval. By leveraging blockchain technology, the authors propose a secure and decentralized system that allows for efficient and private retrieval of medical images. The blockchain-based approach ensures that sensitive medical data remains encrypted and only accessible to authorized parties, mitigating the risk of unauthorized access or data breaches. The proposed solution has the potential to enhance the privacy and security of medical image sharing in IoT-based healthcare systems.

Jabarulla and Lee [10] present a blockchain-based distributed patient-centric image management system to improve the efficiency and security of image management in healthcare settings. By utilizing blockchain technology, the authors aim to address the challenges of centralized image storage and the potential risk of data breaches. The proposed system allows for the secure and efficient sharing, retrieval, and management of patient-centric images. With a patient-centric approach, the system prioritizes the privacy and ownership of the images, empowering patients to have greater control over their medical data. The utilization of blockchain technology ensures data integrity, traceability, and immutability, providing a robust framework for patient-centric image management.

Fernandes et al. [11] introduce a scalable architecture for sharing EHRs using the Hyperledger Blockchain. The authors focus on developing a secure and scalable framework for EHR sharing, leveraging blockchain technology. The proposed architecture addresses the challenges associated with centralized storage, data privacy, and security. By utilizing the Hyperledger Blockchain, the authors ensure secure and auditable sharing of EHRs while maintaining patient privacy. The scalable nature of the framework allows for the efficient sharing of EHRs among authorized healthcare providers, improving data accessibility and interoperability. The proposed solution has the potential to enhance the efficiency, security, and privacy of EHR sharing in healthcare systems.

Huang et al. [12] propose MedBloc, a blockchain-based secure EHR system for sharing and accessing medical data. The authors emphasize the importance of security and privacy in EHR sharing and present a practical solution to address these concerns. By leveraging blockchain technology, MedBloc ensures secure and tamper-proof storage, retrieval, and sharing of medical data. The system utilizes encryption techniques and smart contracts to protect the privacy and integrity of EHRs. The proposed solution reduces reliance on centralized intermediaries and provides a decentralized and auditable platform for secure EHR sharing. The MedBloc system has the potential to enhance data security, privacy, and interoperability in healthcare settings.

Nguyen et al. [13] explore the use of blockchain for secure EHR sharing in mobile cloud-based e-health systems. The authors address the security challenges associated with EHR sharing and propose a blockchain-based approach for enhanced data protection. By integrating blockchain technology into mobile cloud-based e-health systems, the authors ensure secure and trustworthy sharing of EHRs. The blockchain provides a decentralized and transparent platform for data sharing, reducing the risk of data breaches and unauthorized access. The proposed solution enhances data security, privacy, and integrity while maintaining the flexibility and accessibility of mobile cloud-based e-health systems. The research highlights the potential of blockchain technology in improving the security and privacy of EHR sharing in mobile healthcare environments.

Shen, Guo, and Yang [14] introduce MedChain, an efficient healthcare data-sharing platform built on blockchain technology. The authors focus on improving the efficiency and reliability of healthcare data sharing while maintaining data security and privacy. The MedChain platform utilizes blockchain's decentralization, immutability, and transparency to establish a secure and trustworthy environment for healthcare data sharing. By leveraging smart contracts and cryptographic techniques, the authors ensure privacy, access control, and data integrity within the MedChain platform. The system allows authorized healthcare providers to securely share and access patient data, streamlining the exchange of medical information and facilitating collaborative decision-making. The use of blockchain technology ensures that data remains tamper-proof and auditable, reducing the risk of unauthorized modifications or data manipulation. MedChain contributes to the field by offering an efficient and secure solution for healthcare data sharing, addressing the challenges of data security, privacy, and interoperability in the healthcare domain.

Tang, Tong, and Ouyang [15] propose a medical image-sharing system based on blockchain and smart contracts of credit scores. The authors present a novel approach that utilizes blockchain technology to establish credit scores for participants, facilitating secure and controlled medical image sharing. The system assigns credit scores to healthcare providers based on their reputation and contributions to the network, ensuring that only trustworthy entities can access and share medical images. Smart contracts enforce access control and govern the sharing process, allowing for fine-grained permission management and traceability. By leveraging blockchain and smart contracts, the proposed system enhances trust, transparency, and accountability in medical image sharing, ultimately improving the quality and efficiency of healthcare services.

Masood et al. [16] present a lightweight chaos-based medical image encryption system using random shuffling and XOR operations. The authors propose an encryption technique to enhance the security of medical image data during transmission and storage. The system utilizes chaos theory and random shuffling algorithms to scramble the pixel values of medical images, making them resistant to unauthorized access and decryption. The XOR operation further strengthens the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

encryption process, introducing an additional layer of complexity. The lightweight nature of the proposed system ensures efficient encryption and decryption processes without compromising the quality and integrity of medical images. The research offers a practical solution to safeguard sensitive medical image data, protecting patient privacy and preventing unauthorized access or tampering.

The review of existing techniques for secure medical data sharing highlights that while they offer certain advantages, they also have significant limitations. These limitations include vulnerability to single points of failure, susceptibility to unauthorized access, and lack of robustness in protecting data integrity, confidentiality, and authenticity. To address these challenges, a novel system named SECURE-MEDISHARE is proposed. This system employs a comprehensive approach integrating blockchain technology, watermarking, steganography, and enhanced cryptography to provide efficient and robust security mechanisms for medical data sharing. SECURE-MEDISHARE offers several advantages, including enhanced security and privacy protection, efficient data sharing and retrieval, and improved trust among healthcare providers. The proposed system ensures the integrity and authenticity of medical data, preventing unauthorized modifications or tampering. Additionally, the decentralized nature of blockchain technology reduces the risk of data breaches and single points of failure.

3 Methodology:

This Medical data, including electronic health records (EHRs) and medical images, plays a crucial role in healthcare by enabling accurate diagnosis, treatment planning, and medical research. Efficient sharing of this sensitive medical data among healthcare providers is essential for ensuring quality care delivery and collaborative decision-making. However, the secure sharing of medical records and images remains a significant challenge due to the inherent risks associated with data integrity, confidentiality, and authenticity.

Existing techniques for medical data sharing often fall short of providing comprehensive security measures. Centralized systems, which are commonly employed, are susceptible to single points of failure and unauthorized access, compromising the confidentiality and privacy of sensitive medical data. Additionally, traditional encryption techniques may not fully address the complex requirements of medical data, such as ensuring data integrity and authenticity, while minimizing the impact on data quality and usability.

To address these limitations, this section introduces SECURE-MEDISHARE, a novel secure medical data-sharing system that integrates blockchain technology, watermarking, steganography, and enhanced cryptography. The proposed SECURE-MEDISHARE system aims to provide robust security mechanisms for the sharing of medical records and images in a healthcare environment.

SECURE-MEDISHARE utilizes blockchain technology to ensure decentralized and tamper-resistant storage and sharing of medical data. Unlike centralized systems, the decentralized nature of blockchain reduces the risk of single points of failure and unauthorized access. Watermarking techniques are employed to ensure data integrity and authentication, while steganography is used for confidential transmission of metadata, ensuring authenticity, privacy, and confidentiality of medical data. An optimized hybrid cryptography technique is implemented to secure the transmission and storage of medical data, safeguarding its confidentiality and privacy.

The SECURE-MEDISHARE system offers several advantages over existing techniques. It provides enhanced security and privacy protection, efficient data sharing and retrieval, and improved trust among healthcare providers. The system ensures the integrity and authenticity of medical data, preventing unauthorized modifications or tampering. Furthermore, the decentralized nature of blockchain technology reduces the risk of data breaches and single points of failure.

This section presents the methodology employed in the development of the SECURE-MEDISHARE system. It describes the design and implementation of the system. The methodology encompasses the system architecture design, blockchain integration, watermarking techniques, steganography techniques, and enhanced cryptography algorithms.

3.1 System architecture of SECURE-MEDISHARE system:

In the SECURE-MEDISHARE system, the system architecture is designed to overcome the limitations and challenges of traditional medical data sharing. The architecture consists of several components, including the Admin, Blockchain, Authentication Server, Certification Authority, Doctor, Nurse, Pharmacist, and Patient. These components work together to ensure secure registration, authentication, encryption, and sharing of medical data.

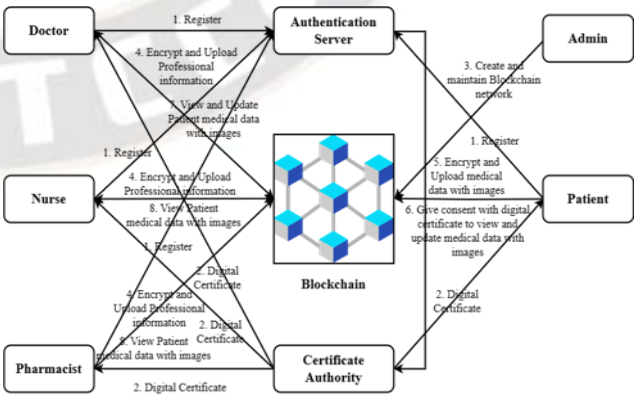

Figure 1: Architecture of the SECURE-MEDISHARE system

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

The concept depicted in Figure 1 entails a comprehensive system within a healthcare setting that focuses on registering and authenticating doctors, nurses, pharmacists, and patients. It involves the issuance of digital certificates for authentication, the establishment and management of a blockchain network, as well as the implementation of secure encryption and uploading mechanisms for professional information and medical data.

1. The registration process with the Authentication Server involves doctors, nurses, pharmacists, and patients providing necessary personal and professional information, including name, contact details, and credentials. The authentication server verifies this information to ensure the authenticity of the users.
2. After successful registration, a certification authority issues digital certificates to the registered healthcare professionals and patients. These digital certificates serve as proof of identity and are used for encryption and decryption purposes within the system.
3. A network administrator is responsible for creating and maintaining a blockchain network, which functions as a decentralized and distributed ledger. This secure platform records transaction and information across multiple nodes, ensuring the secure storage and management of encrypted professional information and medical data, including medical images.
4. Doctors, nurses, and pharmacists can encrypt and upload their professional information, such as qualifications, certifications, and work history, onto the blockchain network. This information is securely stored within the blockchain and can only be accessed by authorized parties.
5. Patients can encrypt and upload their medical data, which typically includes their medical history, diagnoses, treatments, medical images, and other relevant health information, onto the blockchain network. This encryption ensures the privacy and security of the medical data.
6. Patients have control over their medical data through the use of a digital certificate and an expiry date. They can specify their consent preferences regarding who can view and update their medical data. With their digital certificate, patients can provide explicit consent for specific healthcare providers, such as doctors, nurses, and pharmacists, to access their medical data within a specified timeframe.
7. Doctors registered in the system can view and update the medical data of their patients securely. This allows them to access and modify their patients' medical records to provide appropriate healthcare services.
8. Nurses and pharmacists registered in the system can only view patient medical data and do not have the authority to make changes. This ensures that nurses and pharmacists have access to relevant patient information for quality care while maintaining the integrity of the medical data.

These steps collectively establish a secure and transparent system for registering and authenticating healthcare professionals and patients, issuing digital certificates, creating and maintaining a blockchain network for secure storage of professional information and medical data, and enabling authorized access to patient medical data based on roles and permissions. This system ensures the privacy, security, and integrity of healthcare data while facilitating necessary access to quality healthcare services.

3.2 Registration:

This section describes a registration and authentication process in the SECURE-MEDISHARE system. The process utilizes an authentication server to verify the identities of doctors, nurses, pharmacists, and patients. During the registration process, users provide personal information such as their name, email ID, and role (doctor, nurse, pharmacist, or patient), which is then validated by the authentication server to ensure authenticity.

As part of the registration process, users are assigned a unique HMAC-SHA3 hash value based on their email ID. This hash value serves as a secure identifier for users to log in and access the system at a later time.

To provide more clarity, the registration process typically involves the following steps:

- **Collection of Personal Information:** During registration, doctors, nurses, pharmacists, and patients provide their personal information, including their name and email ID. They also specify their role within the system, indicating whether they are registering as a doctor, nurse, pharmacist, or patient.
- **Verification by the Authentication Server:** The authentication server verifies the provided personal information to ensure its validity and authenticity. This verification process may include checking the email ID against existing records, validating the email ID format, and confirming that the specified roles are recognized in the system.
- **Generation of HMAC-SHA3 Hash:** Once the personal information is verified, the authentication server generates an HMAC-SHA3 hash value based on the registered email ID. HMAC-SHA3 is a cryptographic hash function that produces a fixed-size hash value, which is crucial for secure authentication.
- **Provision of Hash Value:** The generated hash value is then provided to the users (doctors, nurses, pharmacists, and patients) as part of the registration process. This hash value is typically conveyed through a secure channel, such as email, to ensure its confidentiality and integrity.
- **Login and Access:** During subsequent logins, users enter their email ID and the hash value generated during

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

registration. The system verifies the entered hash value against the stored hash value associated with the corresponding email ID in its records. If the hash values match, users are granted access to the system, enabling them to log in and utilize the system based on their designated roles.

Algorithm 1 depicts generating a hash for Plaintext using the HMAC-SHA3 algorithm.

Algorithm 1: HMAC-SHA3 hash generation

Input : plaintext: The plaintext message.
key: The secret key used for HMAC.
Output : hmac: The HMAC-SHA3 hash value.
Step 1 : If the length of the key is greater than the block size of the hash function, compute the hash of the key: key = SHA3(key).
Step 2 : If the length of the key is less than the block size, pad it with zeros to match the block size.
Step 3 : Create two variables, inner_key and outer_key, by XORing the key with specific constants:
- inner_key = key XOR 0x3636...36 (the constant is repeated to match the block size).
- outer_key = key XOR 0x5C5C...5C (the constant is repeated to match the block size).

Step 4 : Concatenate inner_key with the plaintext to form the inner message: inner_msg = inner_key || plaintext.
Step 5 : Compute the hash of the inner message: inner_hash = SHA3(inner_msg).
Step 6 : Concatenate outer_key with the inner_hash to form the outer message: outer_msg = outer_key || inner_hash.
Step 7 : Compute the final HMAC-SHA3 hash by computing the hash of the outer message: hmac = SHA3(outer_msg).
Step 8 : Return the hmac as the output.

The HMAC-SHA3 algorithm is used to generate a hash value, known as the HMAC-SHA3 hash, for a given plaintext message and a secret key. This algorithm provides a way to verify the integrity and authenticity of the message by combining the key with the message and applying a cryptographic hash function.

In Step 1, the algorithm checks if the length of the key is greater than the block size of the hash function. If it is, the key is hashed using the SHA3 hash function. This ensures that the key is within the acceptable size limit for further processing. Step 2 handles the case where the key is shorter than the block size of the hash function. In such cases, the algorithm pads the key with zeros until it matches the block size. This ensures that the key has the correct length required for subsequent operations.

In Step 3, two variables called inner_key and outer_key are created by performing a bitwise XOR operation on the key with specific constants. The inner_key is obtained by XORing the key with the constant 0x3636...36, where the constant is repeated to match the block size. Similarly, the outer_key is obtained by XORing the key with the constant 0x5C5C...5C. These constant values are chosen to create different key values for the inner and outer parts of the HMAC calculation.

Step 4 involves concatenating the inner_key with the plaintext message to form the inner message. Concatenation simply means joining the two values together. The inner message serves as the input for the subsequent hash operation. In Step 5, the algorithm computes the hash of the inner message using the SHA3 hash function. This generates a hash value, called inner_hash, which represents the cryptographic fingerprint of the inner message.

Step 6 involves concatenating the outer_key with the inner_hash obtained from the previous step. This forms the outer message, which combines the outer key and the inner hash into a single value. In Step 7, the algorithm computes the final HMAC-SHA3 hash value by applying the SHA3 hash function to the outer message. This generates the HMAC-SHA3 hash, which provides a secure and unique representation of the original message and the secret key. Finally, in Step 8, the computed HMAC-SHA3 hash value is returned as the output of the algorithm.

The HMAC-SHA3 algorithm offers several advantages over HMAC-SHA256 and HMAC-SHA1. Here are some of the key advantages:

- **Enhanced Security:** SHA3 is the latest member of the Secure Hash Algorithm family, designed as a successor to SHA2 (which includes SHA256). SHA3 has undergone extensive analysis and scrutiny by the cryptographic community, which helps ensure its security. By using HMAC-SHA3, one can benefit from the enhanced security provided by SHA3, making it less susceptible to potential attacks compared to older hash functions like SHA256 and SHA1.
- **Resistance to Known Attacks:** Over time, vulnerabilities and attacks against cryptographic algorithms can be discovered. SHA1, in particular, is considered weak due to known collision vulnerabilities. SHA256, although still widely used, may be susceptible to certain attacks, such as length extension attacks. SHA3, on the other hand, is designed to resist a wide range of cryptographic attacks and provides a stronger security foundation.
- **Improved Performance:** While performance can vary depending on the specific implementation and hardware, SHA3 can offer improved performance in certain scenarios. The SHA3 family of algorithms has been optimized for hardware implementations and can leverage parallelism more effectively, potentially resulting in faster hash computation compared to SHA256 and SHA1. However, the actual performance benefits may depend on the specific use case and implementation details.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

- **Future-Proofing:** As technology evolves and cryptographic requirements change, it is essential to use algorithms that are designed to withstand future attacks. SHA3 is a more modern hash function and offers better long-term security. By adopting HMAC-SHA3, one can future-proof systems against potential vulnerabilities that may arise in older algorithms like SHA256 and SHA1.

3.3 Medical Data Encryption and Decryption:

Upon successful registration, a certification authority (CA) generates digital certificates for the users who have registered. These digital certificates serve as evidence of identity within the system. They include the user's email ID as well as optimized hybrid cryptographic keys (RSA++ public key, RSA++ private key, and BC++ secret key) that are used for encryption and decryption purposes. The RSA++ public key and BC++ secret key are utilized to encrypt messages intended for the user, while the RSA++ private key and BC++ secret key are employed for decrypting messages and performing other authorized actions within the system.

The hybrid encryption technique leverages the strengths of both symmetric and asymmetric encryption, achieving a balance between speed and security. To illustrate the implementation of this optimized hybrid cryptography technique, Figure 2 depicts the architecture of optimized hybrid cryptography for patient medical text data.

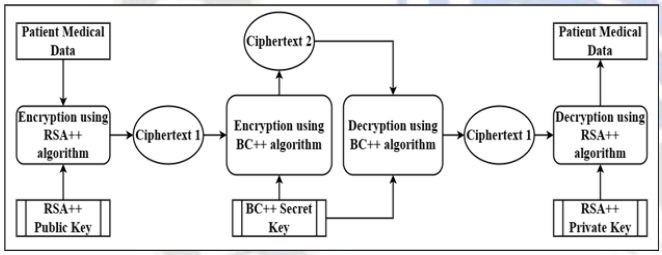

Figure 2: Optimized Hybrid Cryptography for patient medical text data

This optimized hybrid cryptography has 2 processes namely, hybrid encryption and hybrid decryption. Hybrid encryption is a cryptographic technique that combines the strengths of multiple encryption algorithms to provide enhanced security for sensitive data. In the given scenario, the patient's medical data is being encrypted using a hybrid encryption approach involving two algorithms: RSA++ and BC++.

Let's break down the process of hybrid encryption step by step:

RSA++ Encryption:

- The patient's medical data, represented as plaintext, is encrypted using the RSA++ algorithm.
- RSA++ is an advanced version of the RSA algorithm, which is used for secure communication and data encryption.
- RSA++ utilizes a pair of keys: a public key and a private key. In this case, the RSA++ public key is used for encryption.
- The plaintext data is encrypted with the RSA++ public key, resulting in Ciphertext1.

BC++ Encryption:

- Ciphertext1, obtained from the previous step, is further encrypted using the BC++ algorithm.
- BC++ is another encryption algorithm that provides additional security layers to the data.
- BC++ employs a secret key for encryption and decryption. In this case, the BC++ secret key is used for encryption.
- Ciphertext1 is encrypted with the BC++ secret key, producing Ciphertext2.

Uploading Ciphertext2 to the Blockchain:

- Ciphertext2, the final encrypted data, is uploaded to a blockchain.
- A blockchain is a distributed and decentralized ledger that ensures the integrity and immutability of data through a consensus mechanism.
- By uploading Ciphertext2 to the blockchain, the data becomes publicly accessible while remaining encrypted and secure.

Hybrid decryption is the process of reversing the hybrid encryption steps to retrieve the original patient's medical data. Here's how it works:

Downloading Ciphertext2 from the Blockchain:

- Ciphertext2, previously uploaded to the blockchain, is downloaded from the blockchain network.

BC++ Decryption:

- Ciphertext2, obtained from the blockchain, is decrypted using the BC++ algorithm and the corresponding BC++ secret key.
- The BC++ secret key is used to decrypt Ciphertext2, resulting in Ciphertext1.

RSA++ Decryption:

- Ciphertext1, obtained from the previous step, is decrypted using the RSA++ algorithm and the corresponding RSA++ private key.
- The RSA++ private key is the counterpart of the public key used for encryption.
- Ciphertext1 is decrypted with the RSA++ private key, and the original patient medical data, which was the initial plaintext, is obtained.

By combining the encryption capabilities of RSA++ and BC++ algorithms, hybrid encryption ensures that the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

patient's medical data remains secure and protected. The RSA++ algorithm provides asymmetric encryption, allowing the use of a public key for encryption and a private key for decryption. The BC++ algorithm adds a layer of encryption using a secret key, providing an extra level of security. The use of blockchain ensures the integrity and availability of encrypted data, allowing secure sharing and storage.

3.3.1 RSA++ algorithm:

RSA (Rivest-Shamir-Adleman) is an asymmetric cryptographic algorithm widely used for secure communication and data encryption. It involves the use of a pair of keys: a public key for encryption and a private key for decryption.

Disadvantages of the RSA algorithm:

- **Performance**: RSA encryption and decryption operations involve large modular exponentiations, which can be computationally expensive, especially for large key sizes.
- **Key Size**: As computational power increases, the recommended key sizes for RSA need to be larger to maintain security. Larger key sizes require more computational resources and can slow down cryptographic operations.
- **Security Concerns**: RSA is susceptible to certain attacks, such as factoring large numbers and attacks based on the mathematical properties of the algorithm. For example, advancements in quantum computing threaten the security of RSA by potentially breaking its underlying factorization problem.

To address these disadvantages and provides enhancements over the traditional RSA algorithm, the SECURE-MEDISHARE system proposed the RSA++ algorithm. RSA++ is an enhanced version of the traditional RSA algorithm that incorporates various improvements to address the disadvantages and security concerns. It includes techniques such as Optimal Asymmetric Encryption Padding (OAEP) padding, optimized modular exponentiation algorithms, strong probable prime generation, larger key sizes, and randomization techniques. It operates on plaintext input and produces EncryptedPlaintext after encryption and Plaintext after decryption.

The algorithm begins with the generation of the public and private keys. Two large prime numbers, p, and q, are selected using a strong probable prime generation algorithm. The modulus N is calculated as the product of p and q, and the totient function φ(N) is computed as (p - 1) multiplied by (q - 1). An encryption exponent, e, is chosen such that it satisfies the conditions $1 < e < \varphi(N)$ and has a greatest common divisor (GCD) of 1 with φ(N). The modular multiplicative inverse of e modulo φ(N), represented as d, is computed using modular arithmetic. The public key is formed by pairing e with N, and the private key is formed by pairing d with N.

For encryption, the algorithm extracts e and N from the public key. The plaintext is first subjected to OAEP for secure encryption. This padding system adds randomness, error-detection, and hashing to the plaintext. The padded plaintext, converted into a numeric value (e.g., using UTF-8 encoding), is then randomized using techniques such as adding a random salt to introduce entropy and protect against attacks. The randomized numeric value is raised to the power of e modulo N to obtain the EncryptedPlaintext.

During decryption, the private key is used, and d and N are extracted from it. The EncryptedPlaintext is raised to the power of d modulo N to retrieve the numeric value, T. The padding applied during encryption using the OAEP system is removed, revealing the original numeric value. Finally, the numeric value is converted back to plaintext form (e.g., decoding from numeric to text using UTF-8 encoding) to obtain the original Plaintext. Algorithm 2 outlines the asymmetric key cryptography based on the RSA++ algorithm.

Algorithm 2: RSA++ algorithm

Input	**:**	Plaintext
Output	**:**	EncryptedPlaintext (after encryption), Plaintext (after decryption)
		/* RSA++ Public Key and Private Key generation */
Step 1	**:**	Choose two large prime numbers, p, and q, using the strong probable prime generation algorithm.
Step 2	**:**	Calculate N = p * q
Step 3	**:**	Calculate φ(N) = (p - 1) * (q - 1)
Step 4	**:**	Choose an encryption exponent, e, such that 1 < e < φ(N) and GCD(e, φ(N)) = 1
Step 5	**:**	Calculate the modular multiplicative inverse, d, of e modulo φ(N), i.e., d = e^(-1) mod φ(N)
Step 6	**:**	Public Key = Pair of e and N
Step 7	**:**	Private Key = Pair of d and N
		/* Encryption */
Step 8	**:**	Extract e and N from the Public Key
Step 9	**:**	Apply OAEP (Optimal Asymmetric Encryption Padding) to the Plaintext for secure encryption.
Step 10	**:**	T = Convert the padded Plaintext to a numeric value (e.g., using UTF-8 encoding)
Step 11	**:**	Apply Randomization Techniques, such as adding a random salt, to add entropy and protect against attacks.
Step 12	**:**	EncryptedPlaintext = T^e mod N
		/* Decryption */
Step 13	**:**	Extract d and N from the Private Key
Step 14	**:**	T = EncryptedPlaintext^d mod N
Step 15	**:**	Remove the padding applied during encryption using the OAEP system.
Step 16	**:**	Plaintext = Convert T to the original plaintext (e.g., decoding from numeric to text using UTF-8 encoding)

Enhancements in RSA++ algorithm:

- **OAEP Padding**: RSA++ incorporates the OAEP padding system, which is a widely accepted and secure padding

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

technique. OAEP adds randomness, hashing, and error-detection to the plaintext before encryption, making it more resistant to chosen-ciphertext attacks.

- **Miller-Rabin Test**: The strong probable prime generation algorithm employed in RSA++ uses the Miller-Rabin primality test. This probabilistic test ensures the generated primes are highly likely to be prime, adding a layer of security to the key generation process.
- **Larger Key Size**: RSA++ recommends the use of larger key sizes compared to traditional RSA. Larger key sizes, such as 1024 bits or more, provide increased security against advancements in computational power and attacks using brute force or factorization algorithms.
- **Randomization Techniques**: RSA++ applies randomization techniques during encryption, such as adding a random salt or nonce, to introduce entropy and protect against attacks based on predictable patterns. Randomization makes it harder for an attacker to analyze and exploit patterns in the ciphertext.

The RSA++ algorithm incorporates OAEP padding, optimized modular exponentiation techniques, randomization, and strong prime generation to enhance the security and performance of the traditional RSA algorithm. It provides a stronger and more efficient asymmetric key cryptography solution for secure communication and data encryption.

3.3.2 BC++ algorithm:

Blowfish cryptography is a symmetric key block cipher algorithm designed by Bruce Schneier in 1993. It operates on 64-bit blocks of data and supports key sizes ranging from 32 bits to 448 bits. Blowfish uses a Feistel network structure and applies a series of substitutions and permutations to encrypt and decrypt data.

While Blowfish has been widely used and studied, there are some disadvantages associated with it:

- **Vulnerability to brute-force attacks:** The key space for Blowfish is large but not as large as newer encryption algorithms. Advances in computing power make it more susceptible to exhaustive key search attacks.
- **Limited block size:** Blowfish operates on 64-bit blocks, which can be a limitation when processing large amounts of data. This can lead to potential vulnerabilities in certain scenarios, such as when encrypting long messages.
- **Lack of parallelism:** Blowfish does not lend itself well to parallel processing, which can limit its performance in modern computing environments.

The BC++ algorithm, or the Optimized Blowfish Cryptography algorithm, is designed to address these disadvantages and provide improved security and performance. Algorithm 3 outlines the symmetric key cryptography based on the BC++ algorithm.

Algorithm 3: BC++ algorithm

Input	**:**	Input_Plain_text, Key
Output	**:**	Output_Cipher_text, Output_Decrypted_Plain_text
		/* Encryption */
Step 1	**:**	Derive a 512-bit key from the provided Key using PBKDF2 with a salt and iteration count.
Step 2	**:**	Initialize the key_schedule using the derived key.
Step 3	**:**	Generate a random Initialization Vector (IV).
Step 4	**:**	Divide the Input_Plain_text into blocks of 128 bits (16 bytes) using padding if necessary.
Step 5	**:**	For each block: • XOR the block with the IV. • Perform the Blowfish encryption using the key_schedule and the CBC mode of operation. • Set the IV to the resulting ciphertext for the next block.
Step 6	**:**	Assemble the Cipher_text blocks and return the final result as Output_Cipher_text.
		/* Decryption */
Step 7	**:**	Derive the 512-bit key from the provided Key using PBKDF2 with the same salt and iteration count.
Step 8	**:**	Initialize the key_schedule using the derived key.
Step 9	**:**	Divide the Output_Cipher_text into blocks of 128 bits (16 bytes).
Step 10	**:**	For each block: • Save a copy of the ciphertext as the current IV. • Perform the Blowfish decryption using the key_schedule and the CBC mode of operation. • XOR the resulting plaintext with the previous ciphertext (IV) to retrieve the original plaintext. • Set the IV to the saved ciphertext for the next block.
Step 11	**:**	Assemble the Decrypted_Plain_text blocks and return the final result as Output_Decrypted_Plain_text.

The BC++ algorithm takes an input plaintext (Input_Plain_text) and a key (Key) as input and produces the encrypted ciphertext (Output_Cipher_text) and the decrypted plaintext (Output_Decrypted_Plain_text) as output.

For encryption, the BC++ algorithm follows a series of steps. First, it derives a 512-bit key from the provided key using the PBKDF2 algorithm. This key derivation process incorporates salt and iteration count to increase the strength of the key and protect against brute-force attacks. The derived key is then used to initialize the key_schedule, which is a set of subkeys required for the encryption and decryption operations.

To add an extra layer of security, the BC++ algorithm generates a random Initialization Vector (IV). This IV is a unique value that introduces randomness into the encryption process. The input plaintext is divided into blocks of 128 bits (16 bytes) and, if needed, padding is added to ensure that all blocks have the same size.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

For each block of plaintext, the BC++ algorithm performs the following operations. First, it XORs the block with the IV. This XOR operation introduces randomness and prevents patterns in the plaintext from being preserved in the ciphertext. Next, it applies the Blowfish encryption algorithm using the key_schedule and the CBC (Cipher Block Chaining) mode of operation. The CBC mode uses feedback from the previous ciphertext block to encrypt the current block. The resulting ciphertext block becomes the new IV for the next block.

After encrypting all the blocks, the BC++ algorithm assembles the encrypted blocks to form the final ciphertext (Output_Cipher_text), which is the encrypted representation of the input plaintext.

For decryption, the BC++ algorithm reverses the encryption process. It first derives the same 512-bit key from the provided key using PBKDF2 with the same salt and iteration count. The key_schedule is initialized with this derived key. The encrypted ciphertext (Output_Cipher_text) is divided into blocks of 128 bits.

For each ciphertext block, the BC++ algorithm saves a copy of the ciphertext as the current IV. It then applies the Blowfish decryption algorithm using the key_schedule and the CBC mode of operation. The resulting plaintext block is XORed with the previous ciphertext (IV) to retrieve the original plaintext. The saved ciphertext becomes the IV for the next block, ensuring the correct decryption sequence.

Finally, the BC++ algorithm assembles the decrypted plaintext blocks to form the final output (Output_Decrypted_Plain_text), which is the original plaintext obtained from the input ciphertext.

The BC++ algorithm offers several advantages over basic Blowfish. It increases the key strength by deriving a 512-bit key using PBKDF2, making it more resistant to brute-force attacks. The use of random IVs and the CBC mode of operation adds a layer of security to prevent patterns and correlations in the plaintext from being revealed. Furthermore, BC++ optimizes the encryption and decryption processes, resulting in improved performance compared to basic Blowfish.

Overall, the BC++ algorithm provides enhanced security and performance compared to the original Blowfish algorithm. By addressing vulnerabilities and limitations, BC++ strengthens the encryption process and ensures the confidentiality of data.

3.4 Medical Image Encryption and Decryption:

Medical image encryption and decryption involve the application of optimized hybrid cryptography techniques to protect the confidentiality and integrity of sensitive medical images. Here's an overview of the hybrid encryption and decryption process of optimized hybrid cryptography:

3.4.1 Hybrid Encryption Process:

- **Input:** The process begins with a patient medical image, which serves as the input for encryption.
- **Text Watermarking:** A text watermark is added to the medical image. This watermark has a patient name.
- **RSA++ Encryption:** The image description is encrypted using the RSA++ algorithm. RSA++ is an enhanced version of the RSA encryption algorithm, which utilizes public-key cryptography. The encryption process is performed using the RSA++ public key, generating Ciphertext1.
- **Steganography:** Steganography is employed to hide Ciphertext1 within the medical image that now contains the added text watermark. Steganography involves embedding information (in this case, Ciphertext1) within the pixels of an image without perceptible changes to the image. This step produces a stegano image, which looks like the original medical image but contains the encrypted Ciphertext1. In this case, LSB (Least Significant Bit) substitution steganography technique was used.
- **BC++ Encryption:** The stegano image, containing the hidden Ciphertext1, is then encrypted using the BC++ algorithm. BC++ is a symmetric encryption algorithm that utilizes a secret key. This encryption step generates Ciphertext2.
- **Upload to Blockchain:** Finally, Ciphertext2, resulting from the BC++ encryption, is uploaded to a blockchain.

Figure 3 depicts the proposed architecture of the hybrid encryption process for patient medical images.

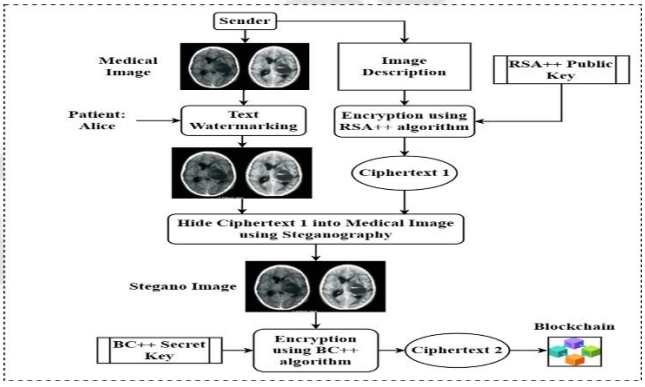

Figure 3: Medical Image Hybrid Encryption Process

3.4.2 Hybrid Decryption Process:

- **Download from Blockchain:** The first step in the decryption process involves downloading Ciphertext2 from the blockchain.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

- **BC++ Decryption:** Ciphertext2 is decrypted using the BC++ secret key, which reverses the BC++ encryption process. This decryption operation retrieves the stegano image that contains the hidden Ciphertext1.
- **Steganography Extraction:** Steganography is employed again to extract Ciphertext1 from the stegano image. By applying the LSB substitution steganography technique, Ciphertext1 is retrieved while maintaining the integrity of the medical image with the added watermark.
- **RSA++ Decryption:** Finally, Ciphertext1, representing the encrypted image description, is decrypted using the RSA++ private key. The RSA++ private key corresponds to the public key used in the encryption process. Decrypting Ciphertext1 with the RSA++ private key yields the original image description, allowing the authorized recipient to access the information contained within the medical image.

Figure 4 depicts the proposed architecture of the hybrid decryption process for patient medical images.

4 Experimental Results and Discussions:

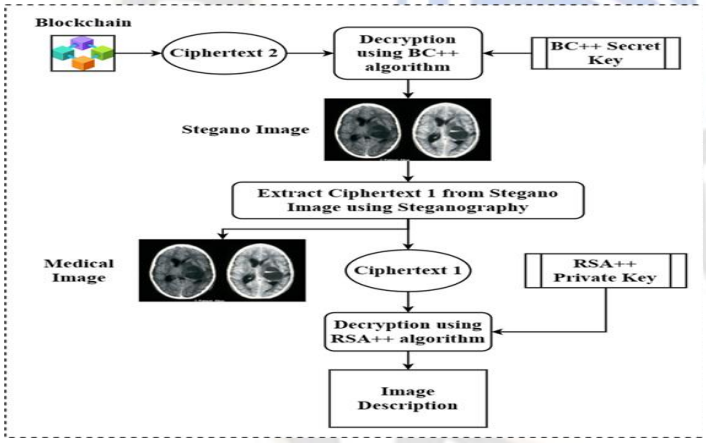

Figure 4: Medical Image Decryption Process

3.4.3 Advantages of Optimized Hybrid Cryptography:

The advantages of using this optimized hybrid cryptography technique for the secure sharing of medical images are as follows:

- **Enhanced Security:** Hybrid encryption combines multiple encryption techniques, each with its strengths, to provide enhanced security. The use of both asymmetric encryption (RSA++) and symmetric encryption (BC++) ensures a higher level of protection for the medical image and its description. This makes it more difficult for unauthorized individuals to access and decipher sensitive information.
- **Text Watermarking:** The addition of a text watermark to the medical image provides an extra layer of information security. The watermark can patient's name, which can help verify the authenticity and integrity of the image. It acts as an additional security measure against tampering or unauthorized modifications.
- **Steganography-Based Hiding:** The use of steganography to hide the encrypted data within the medical image offers another level of security. Steganography ensures that the presence of hidden information is imperceptible to human observers, making it difficult for attackers to detect or manipulate the encrypted data within the image. This covert embedding helps to maintain the confidentiality of sensitive information.
- **Blockchain-based Storage:** Uploading the BC++-encrypted data to a blockchain provides secure storage and immutability. The blockchain acts as a decentralized and tamper-resistant public ledger, ensuring that the encrypted information remains secure and unaltered. By leveraging blockchain technology, the integrity and authenticity of the encrypted data can be verified by authorized parties.
- **Key Management:** The hybrid encryption approach allows for the use of different keys for encryption and decryption. The RSA++ algorithm utilizes public-key cryptography, enabling the secure distribution of the public key while keeping the private key secret. This separation of keys enhances the security of the encryption process and ensures that only authorized individuals with the private key can decrypt and access sensitive information.
- **Selective Access:** The use of encryption and decryption processes enables selective access to the medical image and its description. Only authorized recipients with the appropriate private key can decrypt and retrieve the original image description. This control over access ensures that only authorized individuals can view and interpret sensitive medical information.

Overall, the enhanced hybrid cryptography approach with text watermarking, RSA++ encryption, steganography, BC++ encryption, and blockchain-based storage offers robust security measures for the secure sharing of medical images. It helps protect the confidentiality, integrity, and authenticity of sensitive information, preventing unauthorized access and ensuring that only authorized recipients can access the decrypted data.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

3.5 Use of blockchain in SECURE-MEDISHARE system:

In the SECURE-MEDISHARE system, blockchain technology is utilized to address the limitations of existing techniques and provide robust security mechanisms for medical data sharing. Blockchain is a decentralized and distributed ledger that records and verifies transactions across multiple computers or nodes in a network. Here's why blockchain is needed and how it is used in the SECURE-MEDISHARE system:

- **Decentralized and Tamper-Resistant Storage:** Blockchain technology is needed to ensure decentralized storage of medical data. By distributing the data across multiple nodes, blockchain eliminates the reliance on a central authority, reducing the risk of single points of failure and unauthorized access. This decentralized approach enhances the tamper-resistance of the stored medical data.
- **Data Integrity and Authentication:** Blockchain is used in the SECURE-MEDISHARE system to maintain the integrity and authenticity of medical data. Each transaction or update to the data is recorded in a block, forming an immutable chain. This makes it easier to verify the integrity of the data and detect any unauthorized modifications. The blockchain provides a tamper-resistant and auditable log of all transactions, ensuring the accuracy and authenticity of the shared medical data.
- **Confidentiality and Privacy:** Blockchain technology is utilized to protect the confidentiality and privacy of medical data in the SECURE-MEDISHARE system. The decentralized nature of blockchain ensures that sensitive medical data is not stored in a single location, minimizing the risk of data breaches. Additionally, access controls and encryption techniques can be implemented within the blockchain to further safeguard the privacy of the shared data.
- **Efficient Data Sharing and Retrieval:** Blockchain enables efficient sharing and retrieval of medical data among authorized parties. The distributed nature of the blockchain allows for faster and more reliable access to the data, reducing the reliance on traditional centralized systems that may suffer from latency or downtime issues. Authorized users can securely access and retrieve the required medical data from the blockchain, promoting efficient collaboration and decision-making in healthcare environments.
- **Improved Trust and Transparency**: The utilization of blockchain in the SECURE-MEDISHARE system enhances trust and transparency among healthcare providers. The transparency of the blockchain ensures that all transactions and updates to the medical data can be audited and traced back to their source. This transparency, combined with the immutability of the blockchain, fosters trust among the participating entities, reducing the need for intermediaries and establishing a more reliable and secure data-sharing ecosystem.

Algorithm 4 discusses the use of blockchain in the SECURE-MEDISHARE system.

Algorithm 4: Use of blockchain in SECURE-MEDISHARE system		
Input	**:**	Ciphertext2 (Encrypted Medical Data) (Professional details of Doctor, Nurse, and Pharmacist, Medical data, and medical images of the patient), blockchain
Output	**:**	An event that announces that a new user detail has been added to the blockchain
Step 1	**:**	PreviousBlockHash = 0
Step 2	**:**	SizeoftheBlockchain = 0
Step 3	**:**	For each block B in the blockchain
Step 4	**:**	SizeoftheBlockchain++
Step 5	**:**	PreviousBlockHash = Hash of B
Step 6	**:**	End For
Step 7	**:**	Blocknumber = SizeoftheBlockchain + 1
Step 8	**:**	Timestamp = Get the current time
Step 9	**:**	Nonce = Generate a random number
Step 10	**:**	Hash = Generate hash for Blocknumber, Timestamp, Nonce, and Ciphertext2 // Algorithm 1
Step 11	**:**	If SizeoftheBlockchain == 0 Then
Step 12	**:**	GenesisBlock = Block (Blocknumber, Timestamp, Nonce, Hash, PreviousBlockHash)
Step 13	**:**	Upload GenesisBlock to Blockchain
Step 14	**:**	Else
Step 15	**:**	SucceedingBlock = Block (Blocknumber, Timestamp, Nonce, Hash, PreviousBlockHash)
Step 16	**:**	Upload SucceedingBlock to Blockchain
Step 17	**:**	End If

Algorithm 4 outlines the steps for using blockchain in the SECURE-MEDISHARE system to add a new user's details, including encrypted medical data and professional information, to the blockchain. The algorithm begins by initializing variables and iterating through existing blocks in the blockchain to determine their size and track the previous block's hash. It then calculates the block number, retrieves the current timestamp, and generates a random number as the nonce. The algorithm proceeds to generate a hash for the block using the block number, timestamp, nonce, and encrypted medical data. If the blockchain is empty, a GenesisBlock is created with the necessary details and uploaded. Otherwise, a SucceedingBlock is created and uploaded. The algorithm concludes after adding the new user's details, and the output is an event signaling the addition of the details to the blockchain.

Overall, blockchain technology is necessary for the SECURE-MEDISHARE system to provide decentralized and tamper-resistant storage, ensure data integrity and authentication, protect confidentiality and privacy, enable efficient data sharing and retrieval, and foster improved trust and transparency among healthcare providers.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

3.6 Access control for patient medical data and images:

Access control for patient medical data and images refers to the process of regulating and managing the access, viewing, and updating of patient medical data and images by healthcare professionals based on consent preferences and access permissions. It ensures that only authorized individuals can access the medical images, and their actions are in line with the patient's consent and privacy preferences.

The need for access control in patient medical data and images arises from several reasons:

- **Privacy Protection:** Patient medical images contain sensitive and confidential information. Access control ensures that only authorized healthcare professionals can view and update these images, reducing the risk of unauthorized access or exposure to private medical data.
- **Consent Management:** Patients have the right to control who can access their medical images and under what conditions. Access control allows patients to specify their consent preferences through their digital certificate, granting or denying access to specific healthcare professionals. It ensures that patient autonomy and privacy choices are respected.
- **Data Security:** Medical images are valuable assets and require protection from unauthorized access, tampering, or misuse. Access control mechanisms help prevent unauthorized modifications, ensuring data integrity and protecting against potential security breaches.

Advantages of access control for patient medical data and images include:

- **Enhanced Privacy:** Access control mechanisms ensure that patient medical images are accessed only by authorized individuals, minimizing the risk of privacy breaches and unauthorized disclosures.
- **Consent Management:** Access control allows patients to exercise control over their medical data, enabling them to define who can view and update their images. This promotes patient-centered care and respects their preferences.
- **Improved Data Security:** By restricting access to medical images, access control measures help protect against data breaches, unauthorized modifications, and other security threats. This contributes to maintaining the confidentiality and integrity of patient data.
- **Auditing and Accountability:** Access control systems typically include auditing capabilities, allowing healthcare organizations to track and monitor access and changes to patient medical images. This aids in detecting any suspicious activities and enables accountability and traceability of data access.

Algorithm 5 outlines the Access Control for Patient medical data and images.

Algorithm 5: Access control for patient medical data and images		
Input	**:**	Patient's digital certificate containing consent preferences and access permissions with an expiry date Doctor, nurse, and pharmacist credentials for authentication Patient medical data and images
Output	**:**	Authorized access to view and/or update patient medical data and images based on consent preferences and access permissions
Step 1	**:**	Patient specifies consent preferences using their digital certificate, indicating who can view and update their medical data and images, along with an expiry date.
Step 2	**:**	Authenticate the credentials of doctors, nurses, and pharmacists to ensure their registration and appropriate access permissions.
Step 3	**:**	Verify the validity and compatibility of the patient's digital certificate with their consent preferences.
Step 4	**:**	If the doctor's credentials are authenticated and the digital certificate grants access permissions, allow the doctor to view and update the patient's medical data and images until the expiry date.
Step 5	**:**	If the nurse's credentials are authenticated and the digital certificate grants access permissions, permit the nurse to view the patient's medical data and images without updating them until the expiry date.
Step 6	**:**	If the pharmacist's credentials are authenticated and the digital certificate grants access permissions, enable the pharmacist to view the patient's medical data and images without updating them until the expiry date.
Step 7	**:**	Deny access to patient medical data and images if user credentials are not authenticated or if the digital certificate does not indicate access permissions.
Step 8	**:**	Log any attempted changes to the patient's medical data and images for auditing purposes.
Step 9	**:**	Continuously monitor access and updates to ensure compliance with consent preferences and access permissions.
Step 10	**:**	Terminate the algorithm when authorized users complete their tasks or log out of the system.

Overall, access control for patient medical data and images plays a crucial role in safeguarding patient privacy, maintaining data integrity, and ensuring compliance with regulations, ultimately contributing to the overall quality and security of healthcare services.

4 Experimental Results and Discussions:

In this section, the efficiency of the SECURE-MEDISHARE system was evaluated. An experiment was conducted using a Java-based blockchain. The experiment involved a data string that encompassed various elements,

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

including Ethereum-style smart contracts. To assess the performance of the SECURE-MEDISHARE system, the focus was on two primary evaluation criteria: the time taken for hash generation and the time taken for encryption and decryption.

The time difference (in milliseconds) between pre and post-hash generation in the blockchain network, denoted as Hash Generation Time (HGT), is defined as follows in Eq. (1):

$$HGT = AH - BH \quad (1)$$

Here, BH represents the current time in milliseconds before hash generation, and AH represents the current time in milliseconds after hash generation. Table 1 presents a comparison of the hash generation time across different algorithms used in the blockchain network, including the Algorithm by Shynu et al. [17], the Algorithm by Abunadi et al. for BSF-EHR [18], the Algorithm by Abunadi et al. for BBPM [19], and the SECURE-MEDISHARE system.

Table 1: Hash generation time comparison

Number of Blocks	Shynu et al. [17]	Abunadi et al. [18]	BBPM [19]	SECURE-MEDISHARE
10	22	20	16	18
25	38	36	32	21
50	60	42	37	48
75	90	78	71	60
100	130	118	110	65
Average	68	58.8	53.2	42.4

Upon examining Table 1, the SECURE-MEDISHARE system consistently demonstrates shorter hash generation times compared to the other algorithms for each number of blocks. This indicates that the SECURE-MEDISHARE system is more efficient in generating hashes, resulting in faster processing and computational performance. Efficiency in hash generation time is crucial in blockchain applications as it impacts the overall performance and responsiveness of the system. By having shorter hash generation times, the SECURE-MEDISHARE system can achieve faster transaction processing and enhance the overall efficiency of the system. Therefore, based on Table 1, the SECURE-MEDISHARE system is considered the best choice among the compared algorithms due to its consistently shorter hash generation times, indicating superior efficiency and faster processing capabilities. Figure 5 shows the pictorial diagram of the hash generation time comparison.

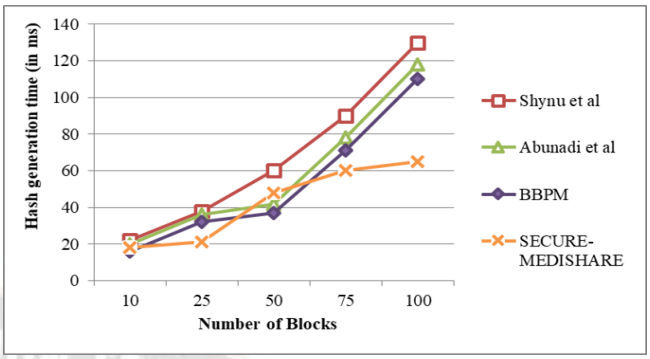

Figure 5: Hash generation time comparison

Moreover, Table 2 compares the times required for encryption and decryption using two-hybrid cryptography combinations, AES - RSA and RSA - ECC, as documented in reference [20], with the proposed optimized hybrid cryptography technique (RSA++ - BC++).

Table 2: Comparative analysis of encryption and decryption times (in seconds) for the Optimized hybrid cryptography technique with two existing hybrid cryptography combinations

Data size (bits)	AES - RSA	RSA - ECC	Optimized Hybrid Cryptography
48	10.2493	8.3266	2.23
64	14.7360	14.8445	2.674
80	21.8297	17.1274	5.718
96	34.0297	21.6842	5.635

Looking at Table 2, we can observe the encryption and decryption times for each technique at different data sizes. The Optimized Hybrid Cryptography consistently outperforms both AES-RSA and RSA-ECC in terms of encryption and decryption times.

For example, at a data size of 48 bits, the Optimized Hybrid Cryptography technique takes only 2.23 seconds for both encryption and decryption, while AES-RSA and RSA-ECC take 10.2493 and 8.3266 seconds, respectively. Similarly, at larger data sizes such as 96 bits, the Optimized Hybrid Cryptography technique takes 5.635 seconds for both encryption and decryption, whereas AES-RSA and RSA-ECC take 34.0297 and 21.6842 seconds, respectively.

From these results, we can conclude that the Optimized Hybrid Cryptography technique offers significantly faster encryption and decryption times compared to the other two combinations. It provides a more efficient and optimized approach to cryptography, making it the best choice among the options presented in Table 2. Figure 6 visually depicts the comparison of these techniques based on both encryption and decryption time.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

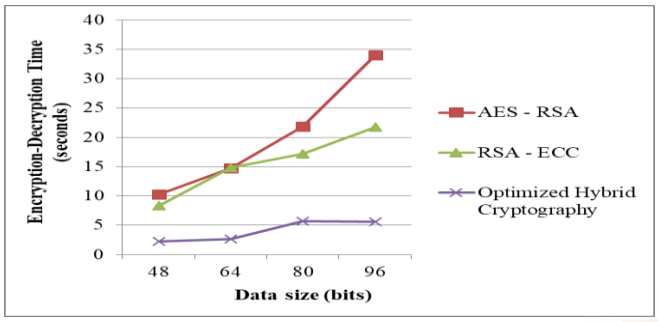

Figure 6: Comparison of data size versus encryption-decryption time for optimized hybrid cryptography technique with two existing hybrid cryptography combinations

Overall, the SECURE-MEDISHARE system stands out for its faster hash generation times compared to other algorithms, and this is primarily attributed to its utilization of the Hmac-SHA3 hash-generating technique. By employing the Hmac-SHA3 technique, the SECURE-MEDISHARE system can generate hashes more quickly and efficiently, contributing to improved performance and reduced processing times.

In addition to the fast hash generation, the proposed optimized hybrid cryptography technique showcased in the system also demonstrates efficient encryption and decryption times across various data sizes.

The optimized hybrid cryptography technique combines different encryption algorithms and strategies to achieve enhanced performance. By carefully selecting and combining the most effective encryption components, the technique maximizes efficiency and reduces processing overhead. This optimized approach ensures that encryption and decryption can be performed swiftly and with minimal computational resources.

The efficiency of the optimized hybrid cryptography technique is particularly noteworthy when considering different data sizes. Table 2 showcases how the technique consistently outperforms other combinations, such as AES-RSA and RSA-ECC, in terms of encryption and decryption times. Regardless of the data size, the optimized hybrid cryptography technique excels in providing faster processing times, indicating its effectiveness and superiority in real-world applications.

5 Conclusion:

In conclusion, the secure sharing of sensitive medical data poses a significant challenge in healthcare. Existing techniques often fall short in terms of protecting data integrity, confidentiality, and authenticity. To overcome these limitations, this paper introduces SECURE-MEDISHARE, a novel system that integrates blockchain technology, watermarking, steganography, and optimized hybrid cryptography to provide robust security mechanisms for medical data sharing. Unlike centralized systems, SECURE-MEDISHARE leverages blockchain technology to ensure decentralized and tamper-resistant storage and sharing of medical data. It employs watermarking for data integrity and authentication and steganography for confidential transmission of metadata, ensuring authenticity, privacy, and confidentiality. Enhanced cryptography algorithms further secure the transmission and storage of medical data, safeguarding confidentiality and privacy. SECURE-MEDISHARE offers enhanced security and privacy protection, efficient data sharing and retrieval, and improved trust among healthcare providers. It ensures the integrity and authenticity of medical data, mitigating the risk of unauthorized modifications. The decentralized nature of blockchain technology reduces the possibility of data breaches and single points of failure. These experimental results demonstrate the efficiency and effectiveness of the SECURE-MEDISHARE system in providing secure medical data and image sharing. Overall, SECURE-MEDISHARE provides a reliable and robust solution for secure medical data sharing in healthcare environments, addressing the challenges associated with data security, integrity, and privacy. In the future, SECURE-MEDISHARE has the potential to make significant contributions to secure data sharing across diverse industries, addressing the challenges associated with safeguarding sensitive information in various contexts. Some potential areas where SECURE-MEDISHARE can be applied include:

- **Financial Data Sharing:** SECURE-MEDISHARE can be utilized to enhance the security and privacy of sensitive financial data, facilitating secure data sharing among financial institutions, regulatory bodies, and customers.
- **Supply Chain Management:** SECURE-MEDISHARE can play a crucial role in securing data sharing and ensuring transparency in the complex network of supply chains, enabling the secure and trustworthy exchange of information among stakeholders.
- **IoT Data Sharing:** By integrating SECURE-MEDISHARE with IoT systems, secure and reliable sharing of data collected from IoT devices can be achieved, ensuring the integrity, confidentiality, and authentication of IoT-generated data.
- **Intellectual Property Protection:** SECURE-MEDISHARE can contribute to protecting intellectual property rights by enabling the secure sharing and distribution of digital assets, such as multimedia files, designs, and sensitive documents.
- **Government Data Sharing:** SECURE-MEDISHARE can be employed to address the challenges associated with secure data sharing among government agencies, ensuring the confidentiality, integrity, and authenticity of sensitive government data.

Extending the application of SECURE-MEDISHARE to these diverse industries and contexts holds the potential to

revolutionize secure data-sharing practices and mitigate the risks associated with unauthorized access, tampering, and data breaches.

References:

[1] Willemink, M. J., Koszek, W. A., Hardell, C., Wu, J., Fleischmann, D., Harvey, H., ... & Lungren, M. P. (2020). Preparing medical imaging data for machine learning. Radiology, 295(1), 4-15.
[2] Shahnaz, A., Qamar, U., & Khalid, A. (2019). Using blockchain for electronic health records. IEEE Access, 7, 147782-147795.
[3] Zhou, S. K., Greenspan, H., Davatzikos, C., Duncan, J. S., Van Ginneken, B., Madabhushi, A., ... & Summers, R. M. (2021). A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. Proceedings of the IEEE, 109(5), 820-838.
[4] Hasan, M. K., Islam, S., Sulaiman, R., Khan, S., Hashim, A. H. A., Habib, S., ... & Hassan, M. A. (2021). Lightweight encryption technique to enhance medical image security on the Internet of medical things applications. IEEE Access, 9, 47731-47742.
[5] Pirbhulal, S., Samuel, O. W., Wu, W., Sangaiah, A. K., & Li, G. (2019). A joint resource-aware and medical data security framework for wearable healthcare systems. Future Generation Computer Systems, 95, 382-391.
[6] Chen, L., Lee, W. K., Chang, C. C., Choo, K. K. R., & Zhang, N. (2019). Blockchain-based searchable encryption for electronic health record sharing. Future generation computer systems, 95, 420-429.
[7] Patel, V. (2019). A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. Health informatics journal, 25(4), 1398-1411.
[8] Wang, Y., Zhang, A., Zhang, P., & Wang, H. (2019). Cloud-assisted EHR sharing with security and privacy preservation via consortium blockchain. Ieee Access, 7, 136704-136719.
[9] Shen, M., Deng, Y., Zhu, L., Du, X., & Guizani, N. (2019). Privacy-preserving image retrieval for medical IoT systems: A blockchain-based approach. IEEE Network, 33(5), 27-33.
[10] Jabarulla, M. Y., & Lee, H. N. (2020). Blockchain-based distributed patient-centric image management system. Applied Sciences, 11(1), 196.
[11] Fernandes, A., Rocha, V., da Conceição, A. F., & Horita, F. (2020, March). Scalable Architecture for sharing EHR using the Hyperledger Blockchain. In 2020 IEEE International Conference on Software Architecture Companion (ICSA-C) (pp. 130-138). IEEE.
[12] Huang, J., Qi, Y. W., Asghar, M. R., Meads, A., & Tu, Y. C. (2019, August). MedBloc: A blockchain-based secure EHR system for sharing and accessing medical data. In 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (pp. 594-601). IEEE.
[13] Nguyen, D. C., Pathirana, P. N., Ding, M., & Seneviratne, A. (2019). Blockchain for secure ehrs sharing of mobile cloud-based e-health systems. IEEE Access, 7, 66792-66806.
[14] Shen, B., Guo, J., & Yang, Y. (2019). MedChain: Efficient healthcare data sharing via blockchain. Applied sciences, 9(6), 1207.
[15] Tang, H., Tong, N., & Ouyang, J. (2018, August). Medical images sharing system based on blockchain and smart contract of credit scores. In 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN) (pp. 240-241). IEEE.
[16] Masood, F., Driss, M., Boulila, W., Ahmad, J., Rehman, S. U., Jan, S. U., ... & Buchanan, W. J. (2022). A lightweight chaos-based medical image encryption system using random shuffling and XOR operations. Wireless personal communications, 127(2), 1405-1432.
[17] P. G. Shynu, V. G. Menon, R. L. Kumar, S. Kadry, and Y. Nam, "Blockchain-based secure healthcare application for diabetic-cardio disease prediction in fog computing," IEEE Access, vol. 9, pp. 45706–45720, 2021.
[18] I. Abunadi and R. L. Kumar, "BSF-EHR: blockchain security framework for electronic health records of patients," Sensors, vol. 21, no. 8, Article ID 2865, 2021.
[19] Abunadi, I., & Kumar, R. L. (2021). Blockchain and business process management in health care, especially for covid-19 cases. Security and Communication Networks, 2021.
[20] Subedar, Z., & Araballi, A. (2020). Hybrid cryptography: Performance analysis of various cryptographic combinations for secure communication. International Journal of Mathematical Sciences and Computing (IJMSC), 6(4), 35-41.