National College of Ireland

Exploring Hybrid Encryption for Enhanced Security of Electronic Health Record in Cloud Environment

MSc Research Project
Cloud Computing

Rashmi Ratilal Vyavahare
Student ID: 22208038

School of Computing
National College of Ireland

Supervisor: Mr. J. K. Sharma

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Rashmi Ratilal Vyavahare
Student ID:	22208038
Programme:	Msc in Cloud Computing **Year:** 2023-2024
Module:	Research Project
Supervisor:	Mr. J.K. Sharma
Submission Due Date:	12th August, 2024
Project Title:	Exploring Hybrid Encryption for Enhanced Security of Electronic Health Record in Cloud Environment
Word Count:	8250 **Page Count:** 28

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Rashmi Ratilal Vyavahare

Date: 12th August, 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Exploring Hybrid Encryption for Enhanced Security of Electronic Health Record in Cloud Environment

Rashmi Ratilal Vyavahare
Student ID: 22208038

Abstract

The main objective of the research is to investigate how a hybrid cryptosystem model may be used to protect sensitive Healthcare data while it is being sent & stored on the cloud. The security, effectiveness, and usability of the hybrid cryptosystem model are compared to those of different encryption techniques in the research. This hybrid cryptosystem model is also more efficient as well as easy to use making it the most feasible option for health organizations looking to protect their sensitive personal data. To achieve the research objectives a systematic literature review was conducted. Data was gathered from existing secondary sources. The study further explores potential challenges and limitations that are associated with the implementation of the hybrid cryptosystem model in healthcare domain. This findings aims to contribute to the development of robust security frameworks for protecting personal sensitive healthcare information in cloud environments.

1 Introduction

Healthcare is a critical sector that generates a huge amount of sensitive patient's data every day. The adoption of cloud-based health information security (HIS) has made it possible to store and share this data in real-time, improving accessibility & efficiency in healthcare delivery. However, outsourcing health data to third-party cloud platforms has also raised significant concerns about privacy and data security. Protecting personal health records from unauthorized access or breaches is a top priority for healthcare organizations and patients. The enhancement of healthcare information security (HIS) in cloud environments is the primary objective of this study. Healthcare professionals, patients, nurses, lab workers, and other individuals are among HIS's clients. Although its ease, cloud computing compromises the privacy of its users. Defining privacy in cloud computing as the ability of an organization to restrict the information about itself that is revealed to the cloud. Public clouds offer cloud services through the Internet. Private clouds, which are created by organizations for their own use and are delivered through a private network are used to supply cloud services. Managing security, data access, service level are now more simple. Both public and private clouds can be found in hybrid clouds. This thesis proposes a special hybrid cryptosystem based on the Pretty Good Privacy Approach (PGP) for securing health records in cloud storage to address these issues. To ensure safe data access and privacy, the suggested strategy combines symmetric and asymmetric encryption techniques. Prior to storing health information on cloud servers, the system encrypts it to protect its confidentiality and shield it from illegal access. The system offers a user-friendly and economical solution, supporting granular user access, multimedia data files & simplified key administration. This project has the potential to increase patient confidence in the security of their health information by offering a secure and effective method for securing Electronic Health Record (EHR) on the cloud. The findings of this research can help create healthcare information security solutions that are more effective as well as efficient, maintaining the privacy, confidentiality and integrity of health data stored in the cloud.

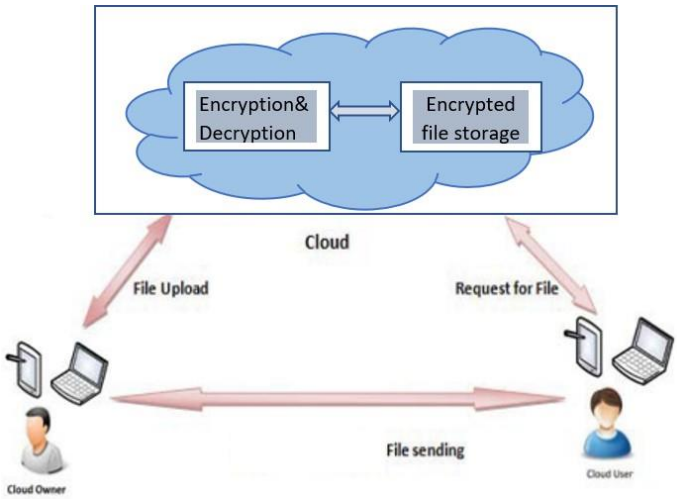


Figure 1 : Information Transfer in cloud.

1.1 Motivation

One of the biggest challenges in ensuring data security in HIS cloud computing (especially in public clouds) is that users cannot be sure that the cloud would not reveal their private information with unauthorized access. Cloud users are not aware of who or what physically owns their data since they are not aware of the places where their data is stored in the cloud, including the owners or managers of those sites. The user's security might be seriously impacted if electronic health record (EHR) is kept on an unreliable third-party cloud service. A cloud service provider may unfairly profit commercially from the usage of customers' data. Since the lack of confidence can only be partially remedied by more advanced technology, even if it were theoretically feasible, believing that users of HIS will not trust a public cloud to provide them with better levels of security (or even privacy) and integrity. We need a solution that enables HIS users to trust the cloud even if they have complete faith in its security measures. This research's original contribution will be achieving health information security in the cloud using a 'PGP-based' technique known as the hybrid cryptographic approach.

1.2 Objectives and Justification

The objectives of this thesis are to:

1. Investigate the current state of health information privacy in cloud computing and the challenges associated with securing electronic health record in the cloud.
2.Evaluate the effectiveness as well as efficiency of the hybrid cryptosystem based on the Pretty Good Privacy Approach (PGP) for enhancing health information privacy in cloud computing.
3. Compare the proposed approach with other encryption methods to assess its effectiveness and usability.
4. Investigate the impact of the proposed approach on system performance and scalability.

Justification: The Users who lack the necessary authorization may nevertheless be able to access health information in this situation. To boost electronic health record security, it is essential to propose a solid strategy that is convincing and practical. Additionally, the seller has total control over who receives compliance with the EHR. As a result only the authorized users will be able to access the EHR. Hybrid cryptography is a technology that combines symmetric encryption with asymmetric encryption. By using a range of other strategies to improve the encryption, it can combine the speed and strength of the two methods. This method

is used to ensure that cloud storage systems are secure. According to research, the method employs the AES and RSA algorithms with AES being used for information or text encryption and RSA being utilized for key encryption. This approach double encrypts data and keys using these two techniques, increasing security. improvements in processing performance for encryption and decryption for large datasets. The system will provide high levels of protection for user data, including written documents, pictures, voice recordings, and streaming video, among other things.

1.3 Importance of health information security in Cloud Computing

Due to the highly confidential and confidential character of health data, cloud computing security of electronic health record is of the highest significance. To preserve patient privacy and keep them confident in healthcare providers, health information must be safeguarded against illegal access, alteration or disclosure. Personal data included in health information includes medical history, diagnosis, treatments, and prescriptions. Even though cloud computing has numerous advantages for the healthcare sector, including cost reductions, scalability, and data accessibility, it also poses several security issues. These difficulties includes a need to protect the data from a variety of attacks, including malware, insider threats and data breaches, as well as the requirement to adhere to several laws and standards, including the General Data Protection Regulation (GDPR) & the Health Insurance Portability and Accountability Act (HIPAA).

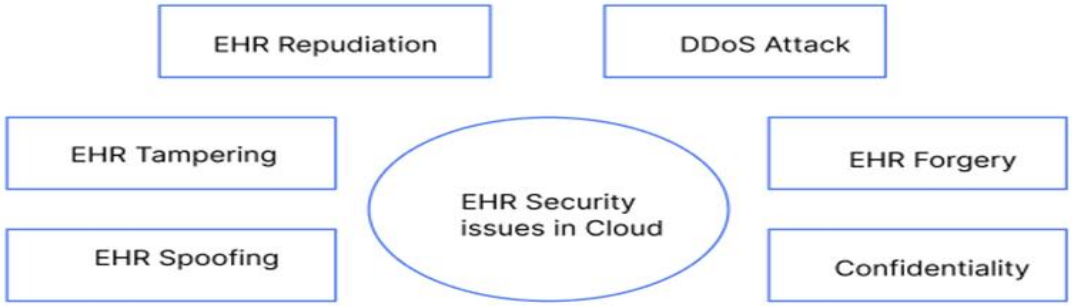


Figure 2: EHR Security issues in Cloud

1.4 Resolving health information security issues in cloud

Implementing strong security measures is necessary to guarantee the confidentiality, integrity and availability of sensitive health information in order to address health information security challenges in cloud computing. Encrypting the data with powerful encryption algorithms is one of the most important elements in reaching this objective. As a result even if the unauthorized users try to access the information, they'll not be able to read or understand it without the decryption key. Implementing access controls and permission systems that limit access to health information based on the user's job and the need to know principle is another crucial step. This can reduce the possibility of data breaches preventing unwanted access. It is crucial to make sure that cloud service providers follow compliance rules and industry standards like HIPAA, GDPR and CCPA, to mention a few. Finally, to reduce the danger of unintended or malicious data breaches brought on by human error businesses should educate their staff on security best practices & perform regular security awareness trainings. Healthcare businesses may guarantee the security as well as privacy of patient health information in cloud computing environments by putting these steps in place.

2 Related Work

2.1 Literature Review

This discussion provides an overview to protect the confidentiality of health information in a public cloud. Cloud computing poses security as well as confidentiality risks & addressing these concerns is crucial for secure data transportation. Various approaches have been used to improve healthcare data security from the consumer end to the server.
Farzana & Islam (2019) proposed a patient centric electronic health record management protocol based on symmetric key encryption. Attribute based access control was used to delegate data access to health professionals based on their attributes. The protocol was verified using AVISPA for security measures.

Pariselvam & Swarnamukhi (2019) employed an El Gamal homomorphic encryption strategy for distributed association rule mining in healthcare sector. Data privacy was protected, but then generating sufficiently large keys posed challenges.

Xu et al. (2019) proposed a patient health information exchange protocol using searchable encryption & multiple keyword searches. This system ensured privacy as well as reliability of search queries but managing the multiple keys for data owners was a difficulty.

Nadaf and Patil (2016) implemented a secure storage as well as retrieval system for health information using AES encryption. Key management was solved by emailing the secret key to users but the system lacked variation in encryption & decryption.

Lu et al. (2021) proposed the dynamic searchable symmetric encryption method for electronic medical services. Accuracy of searched data was checked using homomorphic MAC but the system lacked access control integration.

Mythri and Jayram (2017) used functional encryption for versatile & precise access control in managing large amounts of information. The system offering high information security but had computational & temporal complexity issues.

Personal Medical Records Exchange Mechanism Ensuring Data Privacy & Correctness of Queries: Obiri et al. (2022) presented a personal medical records exchange mechanism ensuring data privacy, correctness of queries & attribute-oriented cryptography. Scheduled key maintenance was not adequately addressed.

Performance Analysis of Encryption Algorithms in Healthcare Data Security: Thakur and Kumar (2011) analyzed the performance of DES, AES & Blowfish are encryption algorithms with Blowfish outperforming others. AES required more processing power.

Modular Encryption Standard for Network Access Control and Patient Privacy Protection: Shabbir et al. (2021) proposed the Modular Encryption Standard for preventing security flaws as well as data breaches. The system provided fine-grained network access and patient privacy protection but had limitations in encrypting image-based data.

Performance Analysis of RSA, ECC, AES Algorithms in Healthcare Data Security: Owolabi et al. (2017) analyzed the performance of RSA, ECC as well as AES algorithms with ECC showing the highest cipher complexity. Sharma et al. (2019) used the RSA key technique for secure health information storage but the approach had higher processing requirements.

Decentralized Hierarchical Attribute Based Encryption & Auditing Architecture: Liu et al. (2020) proposed decentralized hierarchical attribute-based encryption reducing computational complexity and storage costs but making collaboration challenging. Oh et al. (2014) introduced an auditing architecture for broker-based encryption but it lacked details on access control systems.

Encryption Techniques for Digital Photographs and Data Integrity Verification: Zhang and Ding (2015) described a method for encrypting digital photographs using the AES algorithm, but the approach was vulnerable to plain document attacks. Agarwala et al. (2017) introduced a data duplication mechanism called DICE for integrity verification but key generation and administration details were lacking.

Enhanced CP-ABE Solution for Secure Messaging in a Virtualized Environment: Lin and Jiang (2020) proposed an enhanced CP-ABE (Ciphertext-Policy Attribute-Based Encryption) solution with the specific search feature for secure messaging in a virtualized environment. They introduced the concept of a separate virtual machine to facilitate customer cloud interaction & reduce computational load on cloud storage. They also identified various challenges in maintaining the consistent ciphertext and attribute revocation.

Performance Comparison of Discrete Logarithm & RSA Techniques: John et al. (2015) conducted a comparison between the Discrete Logarithm & RSA techniques and found that the ElGamal algorithm performs slower during encryption as well as decryption. This highlights the importance of considering performance trade offs in selecting the encryption algorithms.

-Owolabi et al. (2017) analyzed the performance of RSA, ECC as well as AES algorithms considering the time & complexity. ECC demonstrated the highest cipher complexity making it a most secure option. It also required more time to encrypt the data.

- Zhang and Ding (2015) presented a method for encrypting and decrypting digital photographs using the AES algorithm. They transformed the images into binary matrices and applied the AES method to encrypt each matrix. However, they identified vulnerability to plain document attacks when using a slightly different decryption key which results in incorrect image decryption.

- Agarwala et al. (2017) introduced a reliable data duplication mechanism called DICE (Dual Integrity Convergent Encryption) that aimed to prevent copy faking & deletion attacks while ensuring integrity verification at the source & destination ends. The system verified authenticity of uploaded tags & performed an integrity check during user data retrieval process. However, the key generation & administration process were not thoroughly explained.

- Sharma et al. (2019) addressed the issue of owners of Sensitive Electronic Health Record retaining their privacy by using the RSA key technique. The attribute authority received passwords from patients & physicians to generate a unique key. The data encrypted using the domain based authentication scheme before it being sent to the cloud. This approach ensures anonymity and reduced the workload associated with this key management. It is required generous amount of processing power due to the use of a more sophisticated algorithm.

These additional studies provide insights into different encryption techniques their performance characteristics & the challenges involved in ensuring data privacy as well as integrity in healthcare systems.

3 Research Methodology

This research focuses on creating a system that accurately and effectively identifies security vulnerabilities in cloud storage of medical information. This section describes steps to ensure

a high degree of security for medical information storage on cloud platforms. When used in a cloud environment, its exceptional approach to data protection provides remote servers with the highest level of protection.

Scheme	Advantage	Disadvantage	Reference
symmetric key-based design	Enhanced access control management.	Least efficient and complex computation	Farzana and Islam (2019)
asymmetric cryptography encryption	Reduces the workload of key management.	Using Complex algorithm	Pariselvam and Swarnamukhi (2019)
El Gamal homomorphic encryption	Provides security to data by better encryption technique	Poor key management	Domadiya and Rao (2022)
CP-ABE	shorter ciphertext and a faster running time	Issues with consistent ciphertext and attribute revocation	Lin and Jiang (2020)
searchable encryption technique	Provides strong security to data	Poor key management	Xu et al. (2019)
AES encryption	improved key management	followed similar encryption for every block	Nadaf and Patil (2016)
DSSE	Provide Efficiency and security	failed to reduce computational complexity and communication complexity	Lu et al. (2021)
Feature-based encryption	Reduced overhead of key generation	increased complexity and computational cost	Mythri and Jayram (2017)
Attribute-oriented cryptography system	Fine grained access control and strong security	Lack of information on key maintenance	Obiri et al. (2022)
Performance analysis of DES, AES, and Blowfish	Blowfish outperformed other techniques	The processing power of AES lower than other algorithms	Thakur and Kumar (2011)
Comparison of ElGamal and RSA	Both algorithm Provide strong security	The performance of ElGamal is lower than RSA	John et al. (2015)

Table 1: Summary of the literature review

In the thesis that is being presented, I propose how the model should operate in this part. Time complexity must also be taken into consideration while evaluating the cryptographic algorithm's whole evaluation model. Designing its evaluation settings is therefore also an essential function. The AES and RSA algorithms serve as the foundation for the evaluation model in the suggested thesis. Later the model is assessed based on the timing and phase complexity of its execution. The conversion of plaintext into cipher text initiates the model's

execution.The presence of assaults in this thesis implementation is one of the most crucial things to observe. The server model is subject to vulnerabilities and may experience specific assaults that may be initiated by the attackers since key sharing occurs on an insecure platform. On the other hand, unauthorized access to the generated keys might be obtained by intruders via a man in the middle attack or even an eavesdropping assault. It is crucial to prioritize sending the keys to the authorized user when distributing these keys. Only after sending the legitimate user's login information can this assurance be provided.

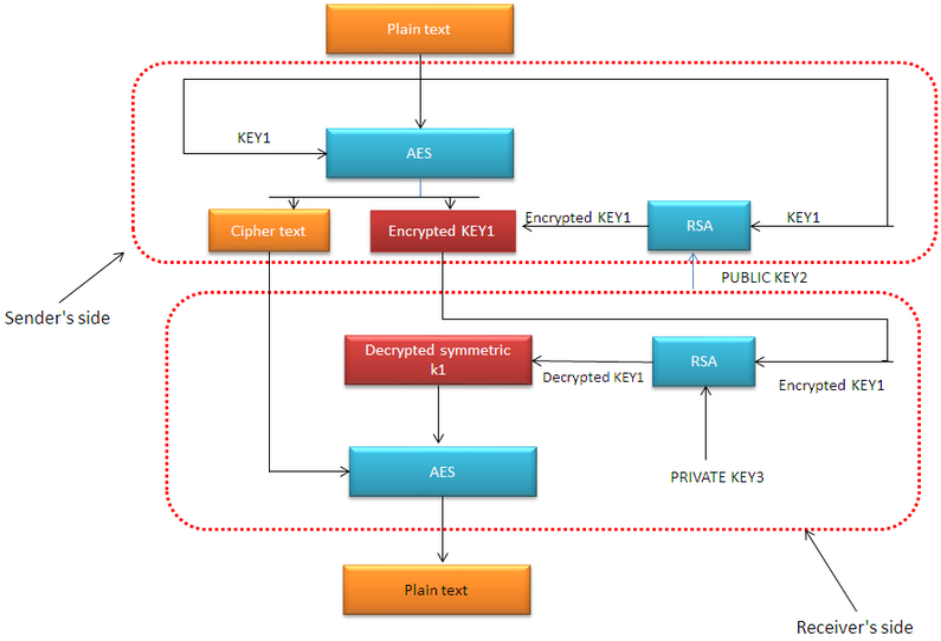


Figure 3: Workflow of AES and RSA.

3.1 The Methodology of Encryption

Encryption is fundamental method used to safeguard the data by transforming the message it into an encoded format that can't easily read by any unauthorized users. This process involve using the algorithm to convert the original message - plaintext into a ciphertext that can only be read with a decryption key. This ensures that even if any of the unauthorized party intervene into the ciphertext as they cannot access or decrypt this original message without the exact decryption key.

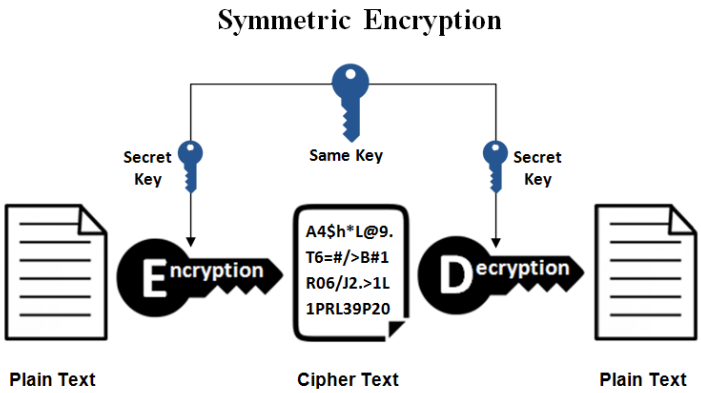


Figure 4: Symmetric Encryption.

Block ciphers like the AES and DES algorithms are recognized and often utilized. After the US standard recognized the DES standard in 1977, the protocol progressively spread throughout the world as the acknowledged norm. The DES implementation became old around

the middle of the 1990s & the key length was reduced to 56 bits. The AES algorithm took the role of the DES algorithm in the latter part of 2001. DES continues to be prevalent particularly in the banking industry despite the existence of several versions.

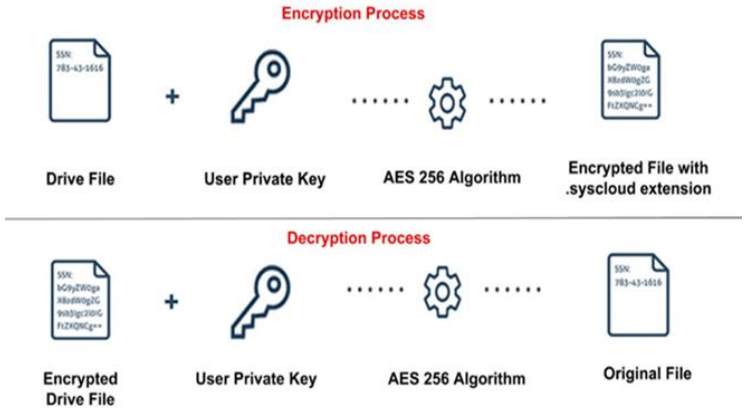


Figure 5: Encryption and Decryption.

Although there have been numerous encryption algorithms developed, many of them have not been able to achieve the level of effectiveness demonstrated by the Data Encryption Standard (DES). The high cost associated with developing new encryption algorithms has made it a challenging task. Despite these difficulties the Advanced Encryption Standard (AES) and DES were created using simpler designs with a weakness being their short key length which left them vulnerable to attacks. The preference for block ciphers over stream ciphers is due to their simpler design and widespread use. AES uses the substitution permutation network while DES uses the Feistel network.

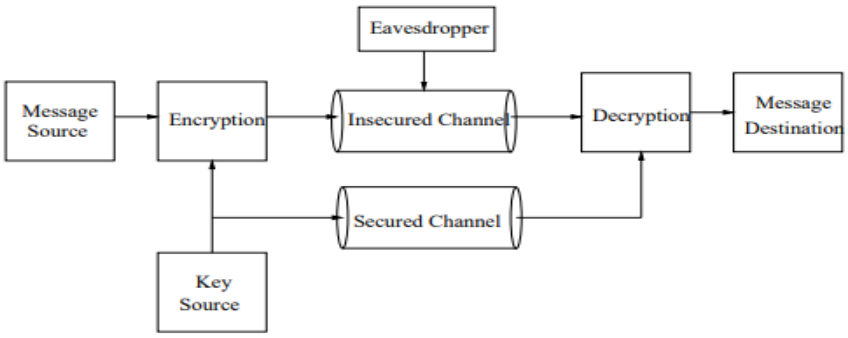


Figure 6: The Cryptosystem Model.

Implementing a traditional cryptosystem the input message is fed into one end of the communication channel and undergoes encryption using a generated key from a desired algorithm. The encrypted message is then transmitted to the recipient through an insecure network platform. This creates a risk of interception by hackers who may inject malicious code to compromise the confidentiality and integrity of the system. Since the communication medium is a public internet securing data transfer becomes a challenging task. The decryption process occurs through the secure channel & requires the presence of the keys generated during encryption.

3.2 Hybrid Approach

The concept of hybridization generally follows the working principle of attaching different algorithms to develop a common model that can have all the properties of related algorithms.

Also, through literature review and related work, we observed that multiple authors used different algorithms to create a common model.

Factors	DES	3DES	AES	Blowfish	RSA	ECC
Develop-ed	IBM in 1975	IBM in 1978	Vicent rijman, Joan Daemon 2001	Bruce Schneier 1993	Ron Rivest 1978	Neal Koblitz, Victor Miller 1985
Key length	56bits	168bits (k1, k2, k3) 112bits (k1 and k2)	128, 192, 256 bits	32 to 448bits	1024bits	160bits
Block size	64bits	64bits	128bits	64 bits	Min 512 bits	64bits
Security	Not secure enough	Not secure enough	Adequately secured	Least secure	Least secure	Adequately secured
Cipher type	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher	Asymmetric block cipher	Asymmetric discrete logarithm
Speed	Moderate	Slower	Faster	Faster	Slower	Faster
Rounds	16	48	10-128 bits key 12- 192 bits key 14- 256 bits key	16	1	16
Power Consumption	Low	Low	Low	Low	High	Low

Table 2: Comparison of multiple working of authors

However, all the author's main goals were to strengthen the model's overall security architecture and better guard against unpredictable attacks. An overview of the algorithms utilized to achieve the same goal is provided in the table above.

Utilizing cryptographic techniques that might safeguard the entire file and shield it from threats like eavesdropping will help to assure security via hybridization. This ensemble approach is regarded as the ideal practice to improve and raise the security of data on server websites since the hybrid algorithm implementation uses both symmetric key and asymmetric key strategies. With a key length equivalent to that of the RSA method of 160 bits to 1024 bits, AES-based encryption is the quickest algorithm in terms of computing and uses less power, as can be seen from the table above.

Despite this, the RSA technique offers significantly less security for data on-site servers when the same parametric conditions are satisfied. Substitution and permutation are two of an AES algorithm's primary operation units. The link between the produced key, the plaintext, and the cipher text is established using the substitution idea. Combining processes like linear and nonlinear algorithms enables the connection of relationships. Permutations, on the other hand, utilize every piece of the encrypted text and use a key to translate it back to plaintext. More

security on the server locations is further enhanced by this hybrid network of substitutes and permutations.

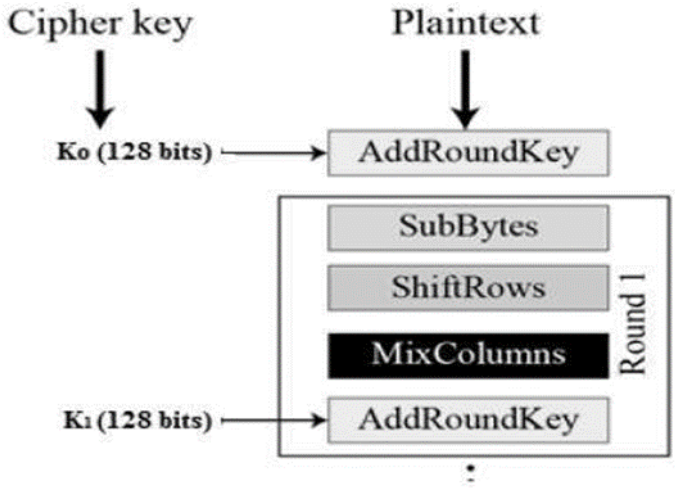


Figure 7: First round process of Hybrid AES algorithm.

3.3 Pretty Good Privacy Approach (PGP)

PGP provides a security and authentication feature for applications like electronic mail and data storage. PGP encrypts the symmetric key so that it may be used as a session key for emailing in the future. Public key cryptography is utilized to encrypt the symmetric key because the other party must receive it in secret. Because only the recipient's secret key, which the attacker is unaware of, can unlock the session key in this case, asymmetric encryption is the best option. This makes it impossible for any attacker to decode the key in time complexity. When developing the thesis, the algorithms used showed in figure 10:

3.4 Advanced Encryption Standard (AES)

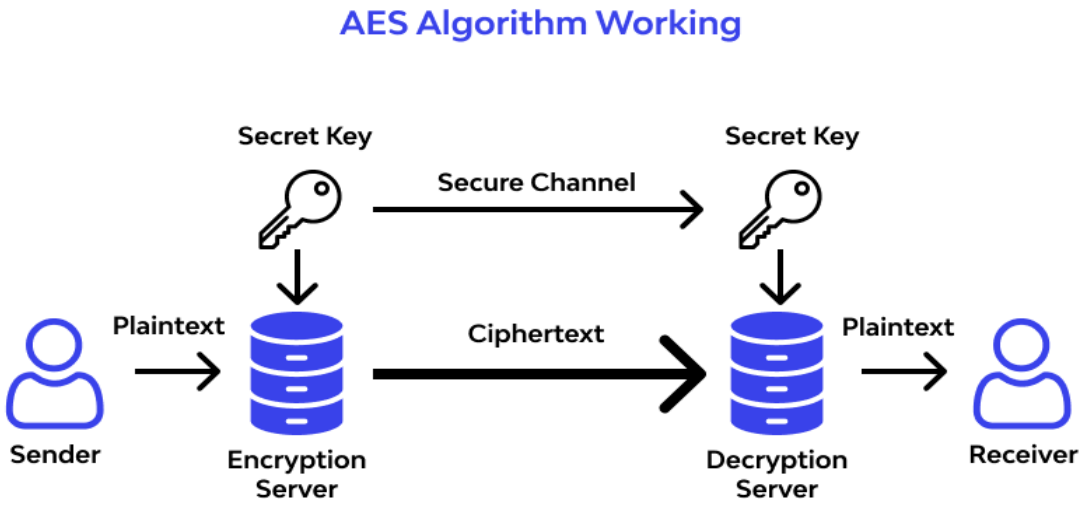


Figure 8: AES Algorithm Working

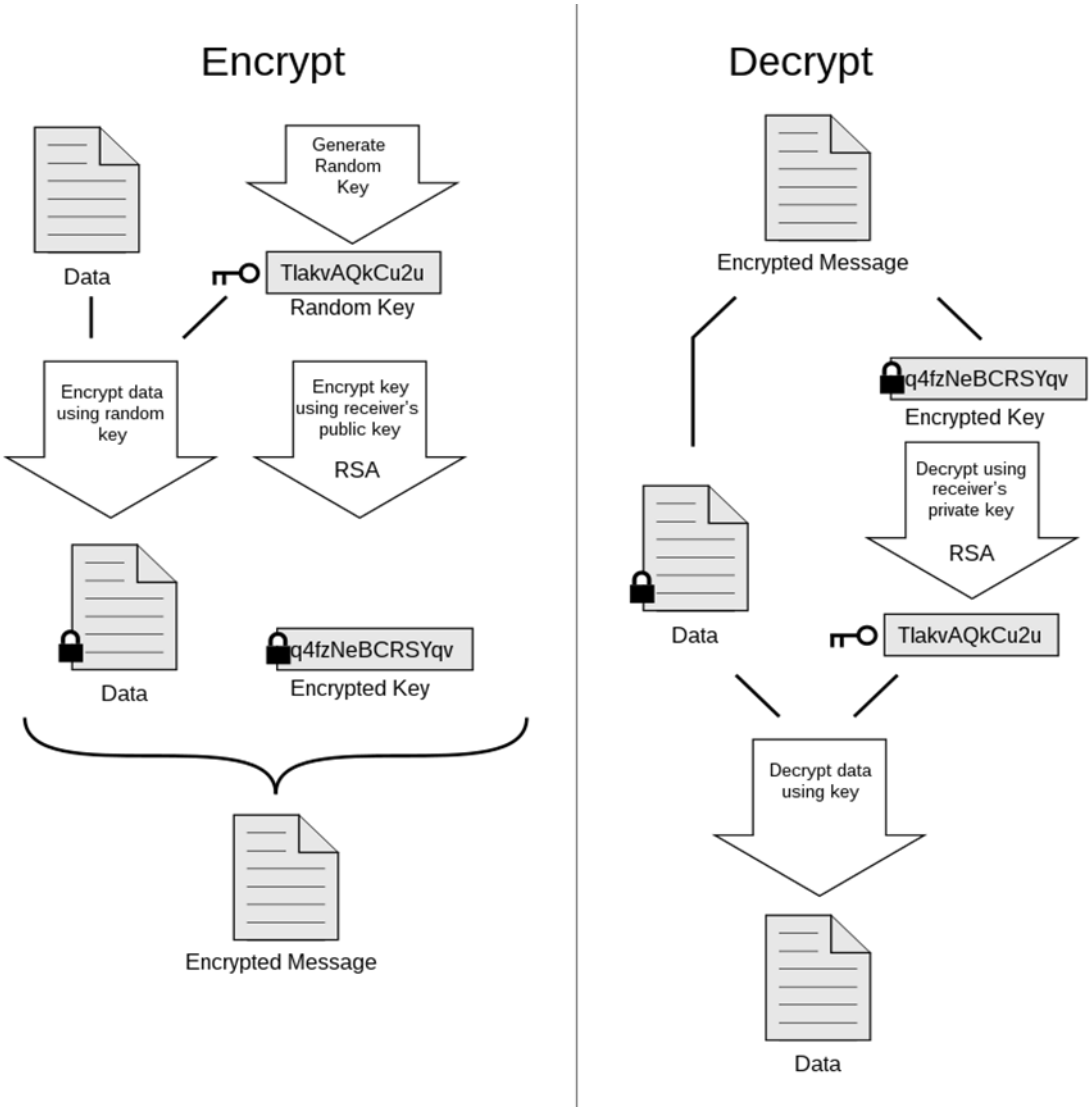


Figure 9: Working of AES and RSA using PGP.

All the encrypted files are sent and stored in the cloud after the Blowfish technique has been used. On the other hand, when the user downloads the file, the user needs to type in his credentials in order to decrypt the message so transferred. This decryption is done to retrieve the original data from the cloud and convert the cipher text into plain text.
The following are the primary features of an AES algorithm:

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Specifications and design specifications should be provided in full
- Software implementable in C and Java

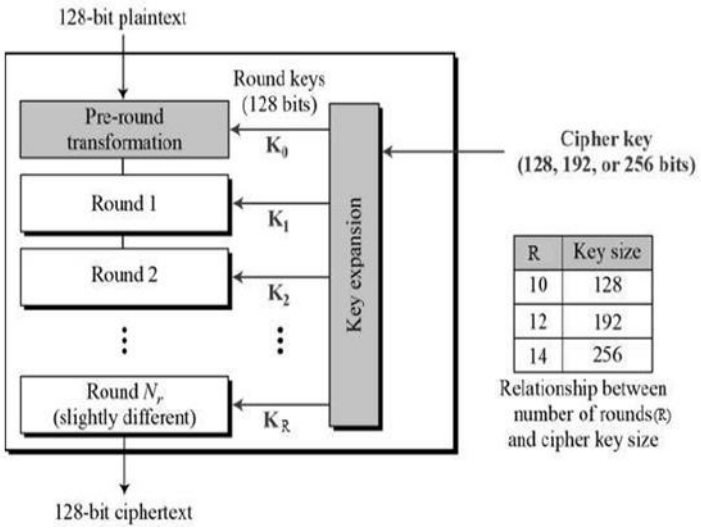


Figure 10: Schematic Representation of AES structure.

3.5 Improvement of AES Algorithm

The database of the hospital information system contains a significant number of medical records in various record sizes. such as prescription drugs, doctor's orders, health information, and other information. The data length of the healthcare order information is often around twenty bytes, making it quite compact. Medical record information can be hundreds of thousands of bytes long and includes things like the home page of an outpatient electronic healthcare record, the record of an outpatient medical examination, the landing pages of an inpatient electronic health record, the history of an admission & more. If there are more data items than a certain number, processing the data will take a lengthy time.

3.6 Rivest–Shamir–Adleman (RSA)

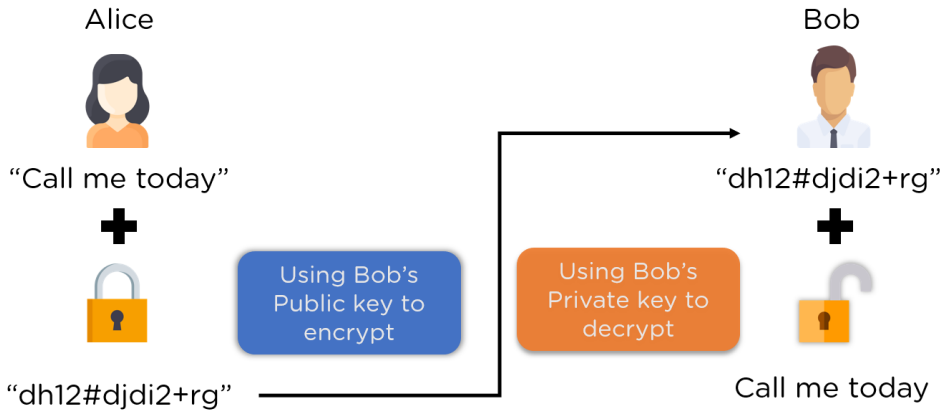


Figure 11: How RSA Works.

The cryptography algorithm RSA is asymmetric. Every other symmetric system operates on the same fundamental tenet. It employs a mechanism based on two keys: a public key and a private key. The private key of each person is kept secret, while their publicly access key is known to all and publicly accessible. Diffie and Hellman, who pioneered the method for the astonishing key transfer, provided the knowledge on the RSA public- key crypto algorithm. RSA uses a variable-size key and an encryption block that may be changed in size. While performing an RSA calculation, one participant uses a p key and the other a private key. The

information in the proposed work will be safeguarded by being encrypted using the RSA calculation so that only the relevant client may decrypt it.

3.7 Performance Evaluation

The performance of the system may be determined by looking at the accuracy metrics of the hybrid cryptography methods. Keeping encrypted files and their source files should be less complicated because multimedia files are compressed before being encrypted and the keys used for both approaches are examined.

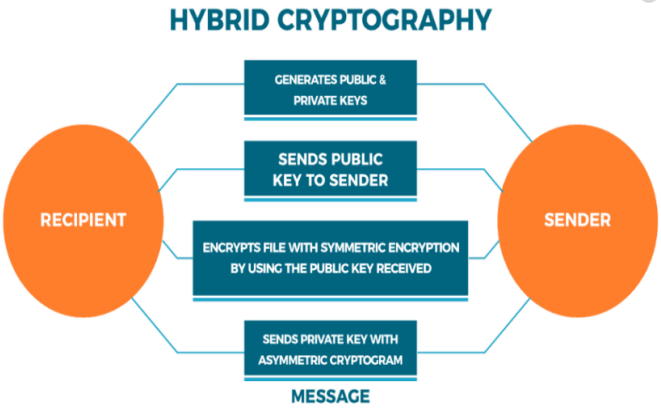


Figure 12: Working of Hybrid Cryptography.

4 Design Specification

The system is designed to function in the following ways:

Step 1: Initially, the admin needs to register to the web application with a pre-defined set of username and password.

Step 2: This login credentials are stored on the database which is triggered upon successful check in of the credentials with the database.

Step 3: Once the login is being made, the admin is further directed to the homepage of the application.

Step 4: The user can access local storage for data files to upload after logging onto the system.

Step 5: Before transmitting the chosen data to the recipient end, the user might then encrypt it. Users of the described approach can choose to combine RSA and AES.

Step 6: In addition, the user could access and examine the files they have submitted.

Step 7: The decryption key is transferred to the email details provided during registration or login when a recipient selects a file to download.

Step 8: This key can then be used by the user to download the decode or original file.

Execution Steps:

1. Registration
2. Login
3. Key generation
4. Key exchange
5. Uploading / downloading / encryption / decryption – AES Hybrid algorithm & RSA Hybrid algorithm
6. Data storage on server site
7. Logout

4.1 Environment Configuration

The Windows Operating System serves as the infrastructure needed to construct the model. The model was created in Visual Studio Code using C sharp .NET. Here, a hybrid crypto system is designed to be as efficient as possible while utilizing a less complicated architecture

Software Specifications

1. ASP.Net
2. HTML
3. CSS
4. Microsoft SQL server for backend
5. C#
6. Microsoft Visual Studio 2022

Hardware Specifications

1. Pentium core
2. RAM size: 16GB
3. Processor: i5,1.2Ghz
4. SSD:1 TB

4.2 Architecture

The utilization of a hybrid cryptosystem is made to protect files. There are two steps to it. As seen in the architecture below, the data was first split into two phases, namely the encryption and decryption components.

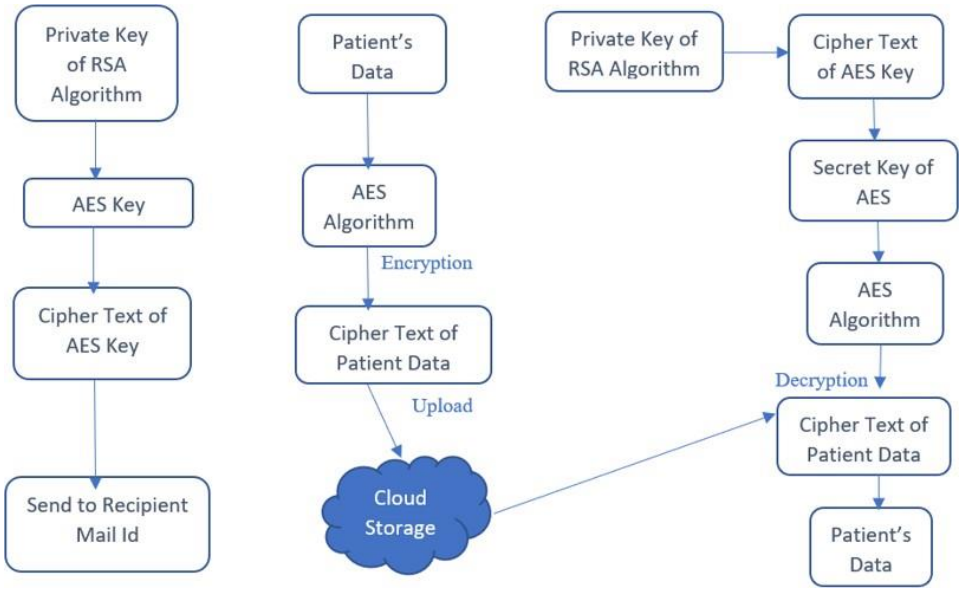


Figure 13: Hybrid Crypto System.

4.3 Encryption Phase

As depicted in Figure [14], the encryption procedure was carried out in a few steps. First, use the file system module to encode the downloaded file, and then divide it into two halves. Two distinct cryptographic methods, such as RSA and Advanced Encryption Standard algorithms (AES), are used to encrypt each component. After merging the sections, a single file is created and sent to the cloud once more fig [15].

4.4 Decryption Phase

The processes used in the decryption process are the exact reverse of those in the encryption process. First, the encrypted file must be downloaded, after which it is split into two halves and disseminated for decoding using the encrypted technique (AES and RSA) fig [15].

5 Implementation

Here, an effective PGP cryptographic system is created by designing it so that the optimal outcomes are attained utilizing a less sophisticated architecture. The following provides an explanation of the hardware and software requirements for development process.

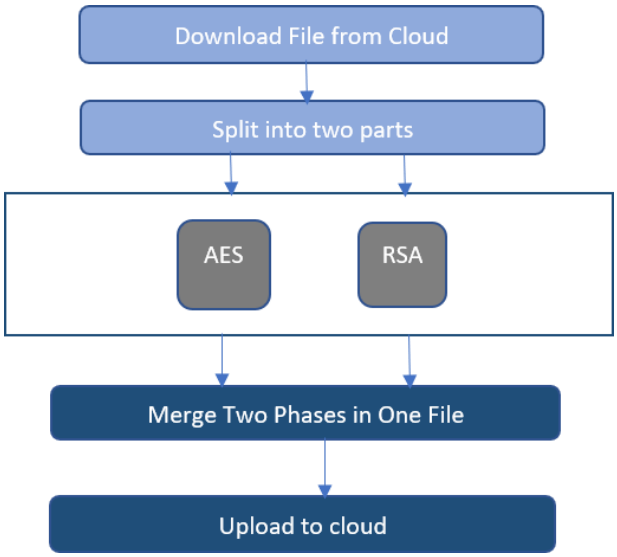


Figure 14: Proposed Encryption Process.

.

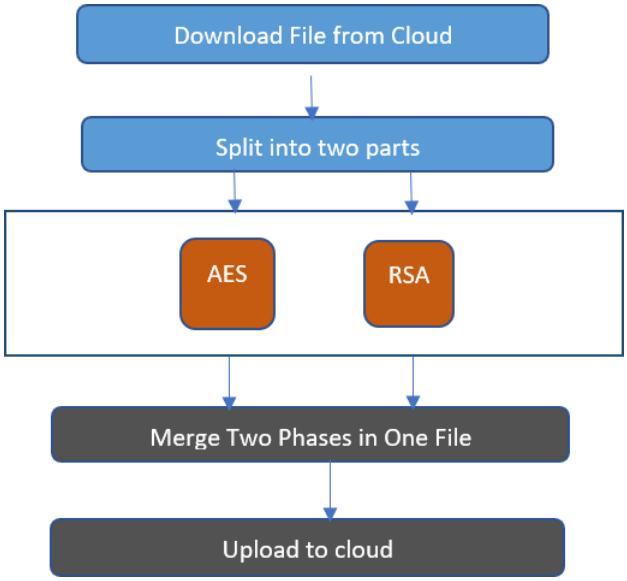


Figure 15: Proposed Decryption Process.

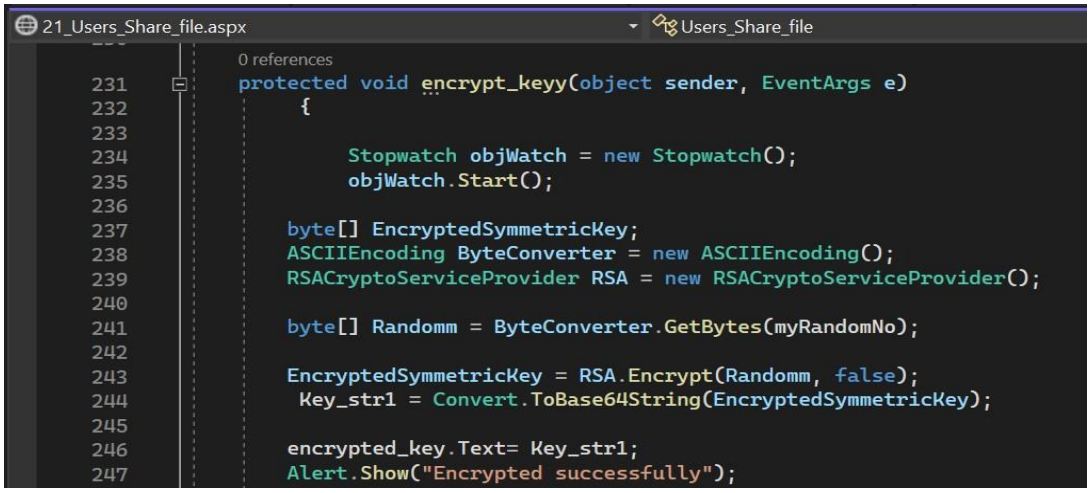
5.1 Implementation of PGP Technique

To incorporate this Pretty Good Privacy (PGP) approach a hybrid cryptosystem model combining symmetric & asymmetric algorithms was employed. This model leverages the robustness of both algorithms to enhance security of data transmission & storage. In the implementation an asymmetric algorithm like RSA is used to generate a secret key. This secret key plays a important role in the encryption & decryption process. For added security the secret key is encrypted using the RSA. This ensures that even if the secret key remains protected & inaccessible to unauthorized individuals.

The secret key encryption a symmetric algorithm such as AES is employed to encrypt the actual data. The symmetric algorithm is very well suited for efficiently encrypting & decrypting massive amount of data. By using a shared symmetric key the data can be securely encrypted & decrypted using the same key. To initiate the decryption process users can receive the encrypted secret key through an email which is a more secure channel. The user then decrypts the secret key using their private key which is securely stored on their device. Once this secret key is decrypted it can be used to decrypt data that was encrypted using the symmetric algorithm.

The combination of symmetric & asymmetric encryption in the PGP approach provides a secure and robust method for protecting sensitive data. The asymmetric encryption ensures confidentiality & integrity of the secret key while the symmetric encryption effectively encrypts and decrypts the original data. This hybrid approach offers a balance between security as well as computational efficiency. The secure distribution & management of encryption keys are essential components of the PGP approach. Strict measures should apply to ensure the secure delivery of the encrypted secret key to the intended recipient. The private key is used for decrypting the secret key as well as to access the data should be carefully prevented by unauthorized access.

By implementing PGP method with a combination of symmetric & asymmetric encryption algorithms the system ensures a high level of security for data transmitted as well as stored in the cloud computing environments. This approach provides confidentiality, integrity & authentication which offers users peace of mind when dealing with sensitive information.



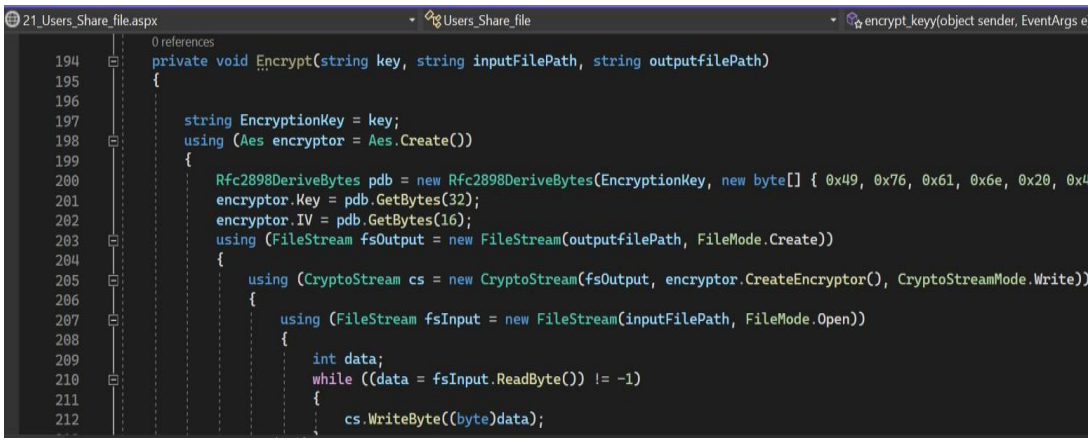


Figure 16: Code for key Encryption.

The screenshot below showcases the code implementation for securely sending encrypted files and data to the users email address. This functionality ensures that the sensitive personal information remains secure during transmission. The code snippet will demonstrate the integration of encryption techniques into the email sending process. It includes multiple steps for encrypting the files or the data using a specified encryption algorithm & generating the unique encryption key. Before it is sent to the designated recipient, encrypted data is attached to the email. By incorporating encryption into the email sending mechanism this application enhances the confidentiality & integrity of the transmitted data. This is particularly crucial while dealing with the sensitive information such as personal healthcare information or financial data where maintaining the data privacy is of most important.

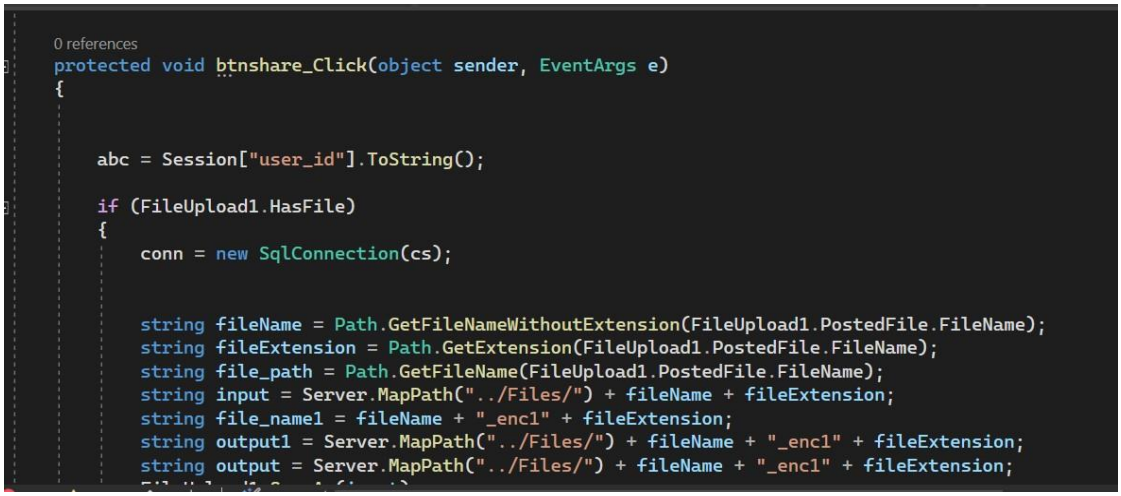


Figure 17: Code for Share encrypted file.

The web.config file acts as a central configuration point, enabling seamless integration between the application and the chosen database infrastructure. It is important to carefully manage and secure the connection details stored in the web.config file. Proper access controls and encryption techniques should be implemented to protect sensitive information, such as database credentials, from unauthorized access. Additionally, regular monitoring and updates to the

web.config file are necessary to ensure the application maintains a reliable and secure connection to both local and cloud databases.

5.2 Cloud Deployment

5.2.1 New web App creation on Azure

In this report, the web application was created following a structured process in Microsoft Azure. The process began by signing into the Azure portal and navigating to the App Services section. From there, a new app service was initiated by clicking Create and configuring the necessary details such as selecting the appropriate subscription, resource group as well as providing the name for the application. The runtime stack set to .NET & the preferred hosting region was chosen as Eash US. The service plan defines the pricing tier & allocated computing resources was also configured during this setup. After reviewing this configurations the following app service was created mentioned in Fig[18]. The deployment of the ASP.NET web application was then completed using publish profile through Visual Studio 2022.
Find my published web app here:
https://pgpencryption-euecg9hyc5csddb4.eastus-01.azurewebsites.net/

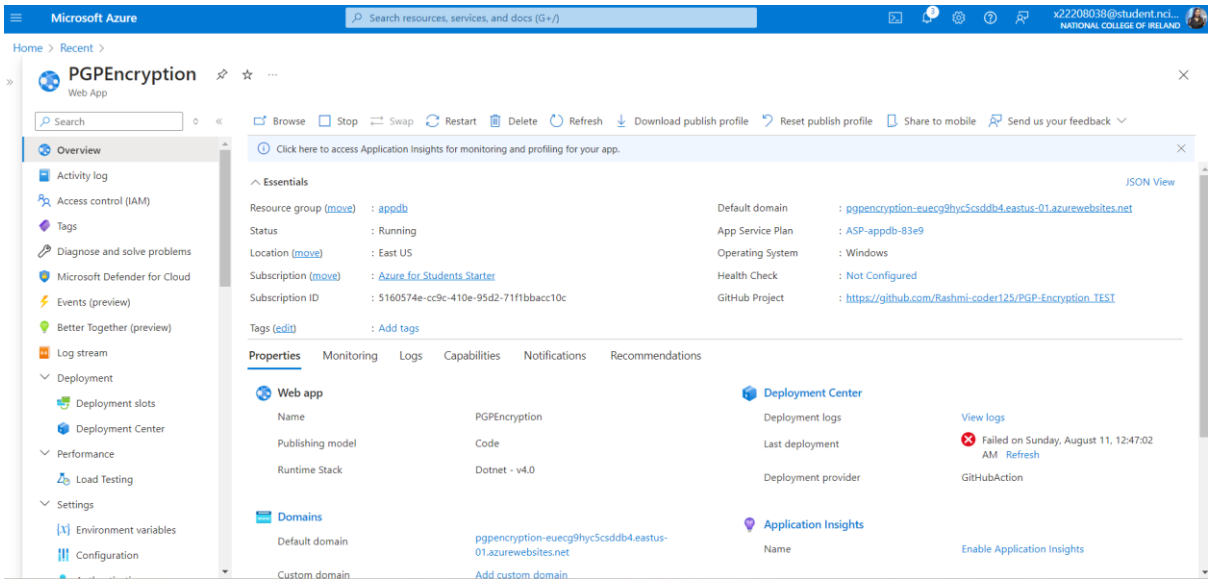


Figure 18: Web App Deployment in Cloud

5.2.2 Create Database on Azure cloud

While working with databases in a cloud platform like Azure, the connection details may vary from traditional on-premises setups.The process of creating a database involves specific configurations & settings that are tailored in the Azure the cloud environment. This includes defining the database type, selecting appropriate service tiers specifying resource allocations & configuring security measures. The connection details for accessing the database in the cloud may differ from traditional setups. Server Name,Connection Strings are defined in Azure SQL Database as shown in Fig [19].Users typically obtain connection strings or connection information that contains the necessary details to establish the connection to the database in Azure cloud. This information may include server names, credentials, ports & other relevant parameters.

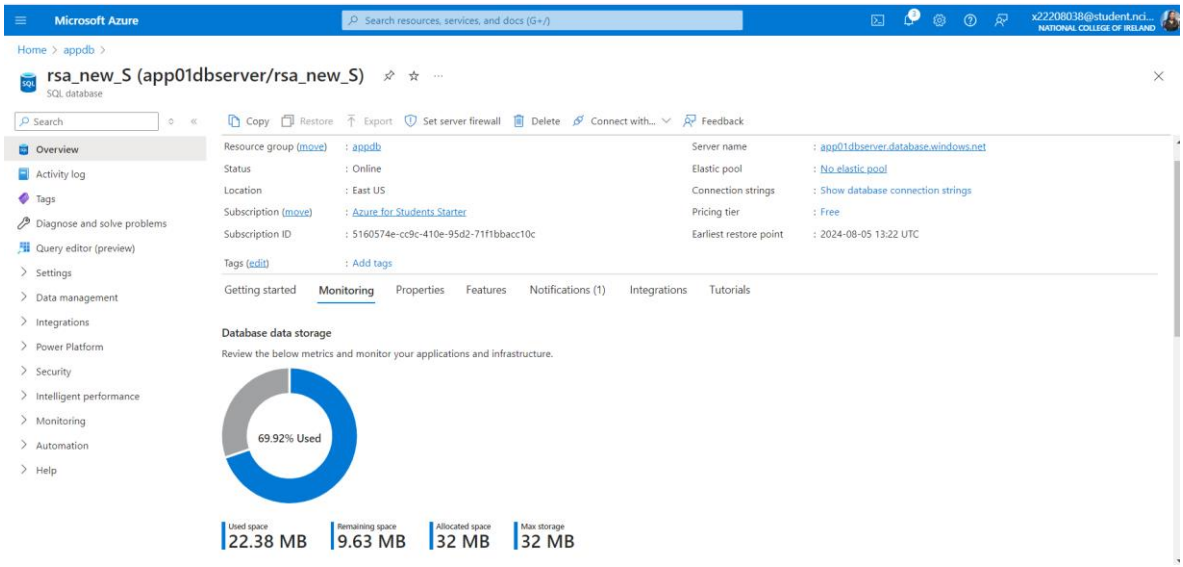


Figure 19: Database created for web app on Azure cloud.

5.2.3 Publish Updated details on Azure web app

The figure provided (Fig:20) demonstrates an option to publish changes from the local machine to the cloud environment. This option enables developers to easily transfer their local code changes to the cloud making the updated code immediately available for execution. By publishing changes to the cloud, developers can take advantage of the scalability, flexibility, accessibility offered by cloud environments.

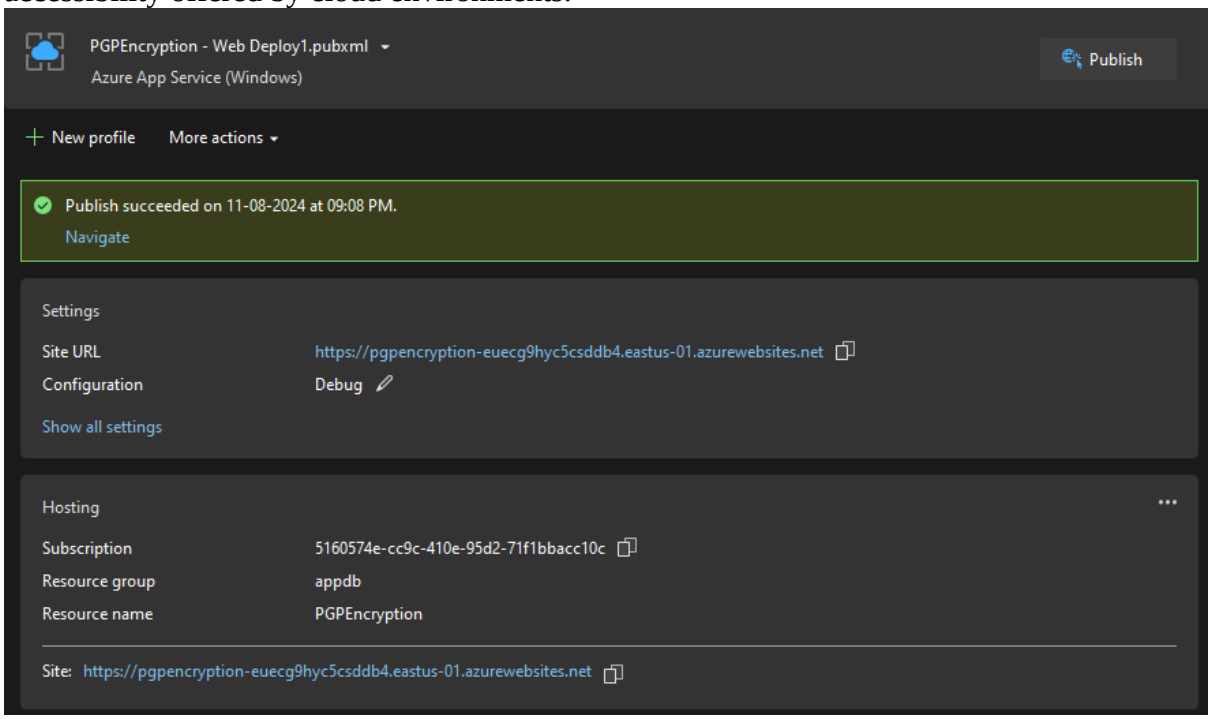


Figure 20: Publish Code Changes on Azure

5.3 Working of the web application

5.3.1 Login and Registration page

The provided screenshots is the login page & the admin portal of the web application. These showcases the interface designed for users to authenticate themselves & gain access to the

application as well as the dedicated area for administrators to manage and oversee various aspects of the application.

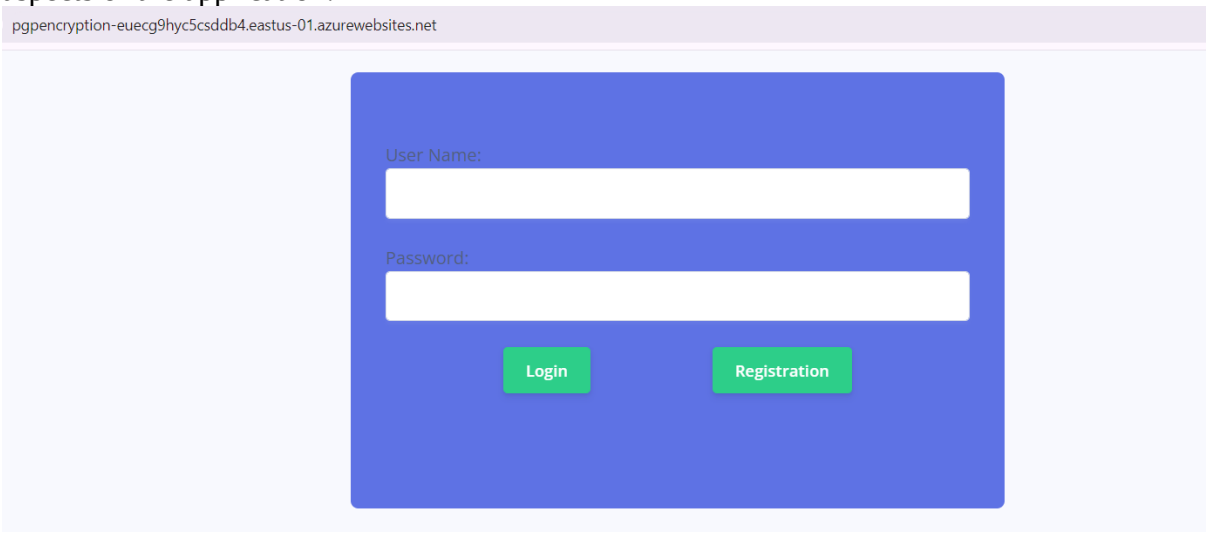


Figure 21: Login Page.

The provided screenshot displays the registration page of the web application which enables users to sign up by providing their full name, contact number, email ID as well as address. Upon registration, the application generates both a public and private key in the backend. As part of the registration process, the web application generates a public and private key in the backend. These keys are cryptographic tools that enhance the security and privacy of user data. The public key as the name suggests is accessible to the anyone who uses this app and can be used for encryption or verification purposes. It allows other users or entities to securely communicate with the registered user. On the other hand the private key remains confidential and is only known to the registered user. It is used for decrypting messages or accessing encrypted data.

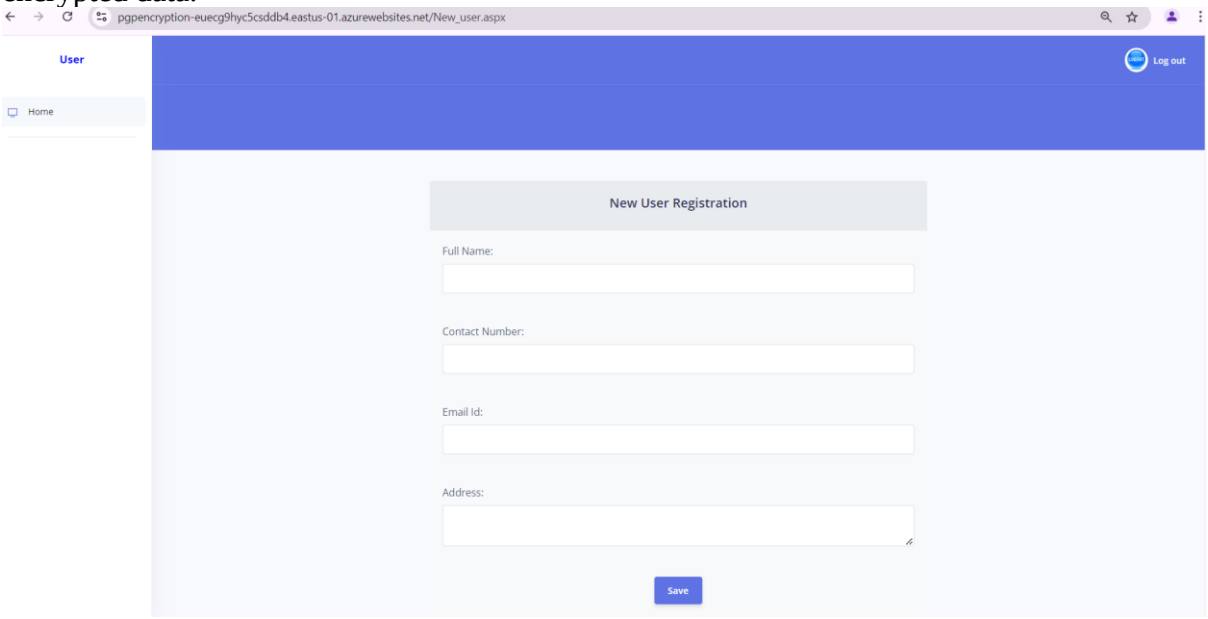


Figure 22: Registration Page.

5.3.2 Home Page

After login with given credentials the provided screenshots showcases various options available within the web app including Share Files, View Share Files, Forgot Password. The View Share Files feature enables users to access & view files stored within the web application. The Forgot Password functionality assists users who have forgotten their password by providing a feature to reset it. It involves a password recovery feature that verifies the user identity through email verification. The Logout option that allows the users to securely log out of their accounts. Logging out ensures that the user's session is terminated which prevents unauthorized access to their account & also protect their privacy.

5.3.3 Share Files

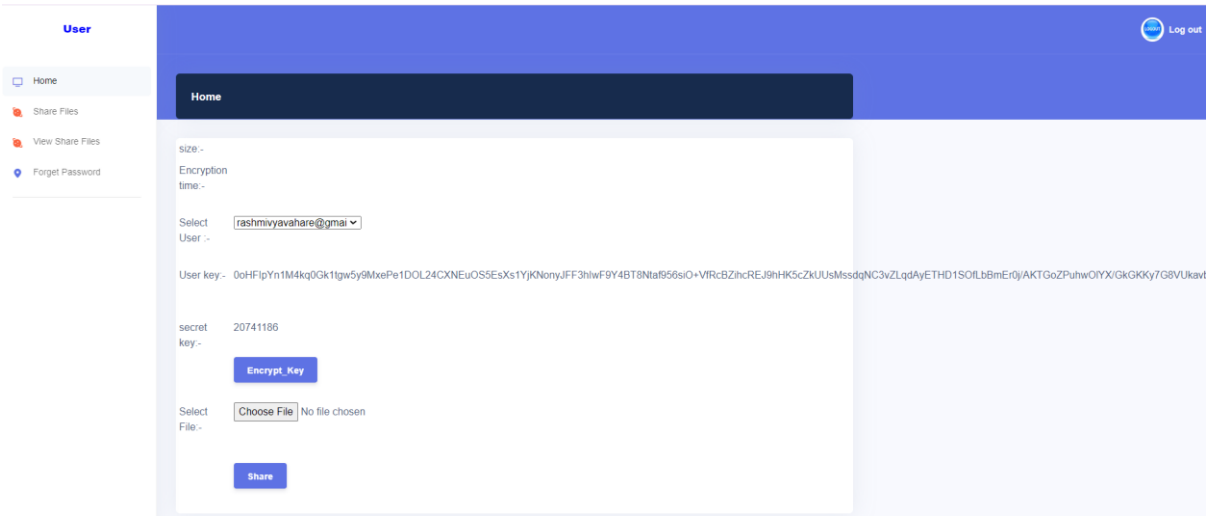


Figure 23: Share files Using PGP method

The Above Screenshot illustrates, the system employs the PGP (Pretty Good Privacy) method to enhance security. As part of this method, the secret key used for encryption is encrypted itself using the recipient's public key (users key). In the encryption process the system takes the secret key and utilizes the recipient's public key which is obtained through the PGP encryption. The recipient's public key is the component of public key cryptography that consists of a public key for encryption and a private key for decryption. Using the recipient's public key this system encrypts the secret key creating an additional layer of security. This ensures that only the intended recipient possessing the corresponding private key which can decrypt and access the original secret key. encrypting the secret key using the recipient's public key the PGP method provides a robust mechanism for protecting sensitive information. This enables secure communication & data transfer by protecting the encryption key itself that is reducing the risk of unauthorized access. The utilization of the PGP method adds up the extra layer of protection to the secret key ensuring that even if intercepted during transmission or storage, the encrypted secret key remains secure and only accessible to the intended recipient with the corresponding private key.

After receiving the encryption Secret key, users have the option to conveniently copy & paste it for further use. In the provided screenshot below, we can observe the logged in view using the receiver's credentials to access the shared file. The list displays list of files that have been shared by the first user to the receiver. Upon selecting the desired shared file the user proceeds to the next page for decrypting & downloading the file. On the decryption page the user is prompted to copy & paste the encrypted secret key that was received via email or another communication channel. By copying the encrypted secret key and entering it into the designated field the decryption process is initiated. The system then successfully decrypts the file making it accessible for download.

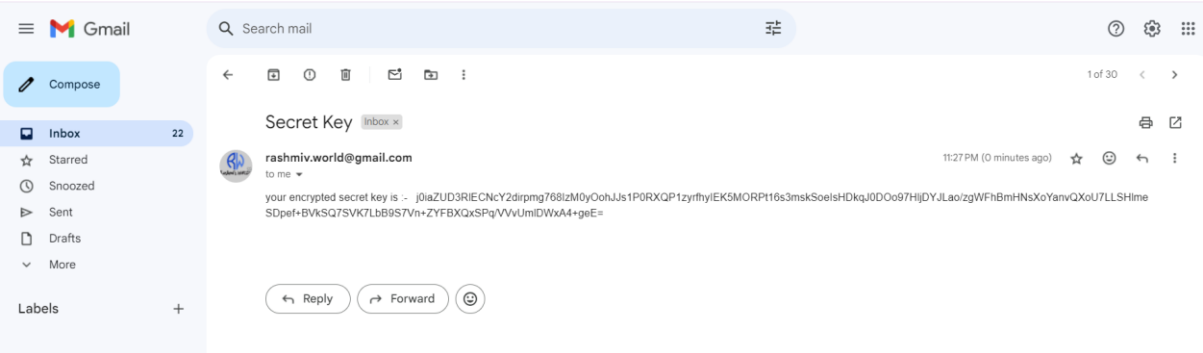


Figure 24: Received encrypted Secret Key.

Figure 25: Share Files Screen

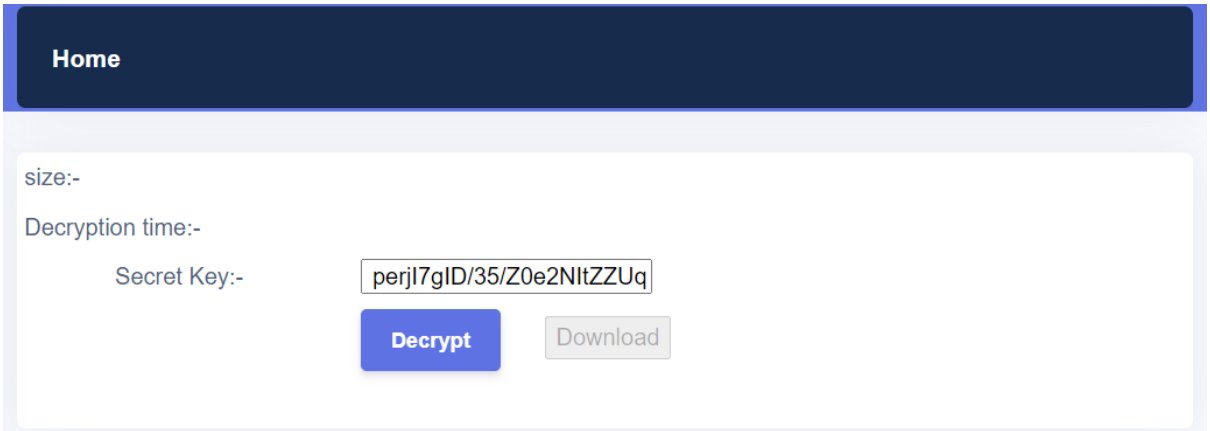


Figure 26: The encrypted secret key is copied and pasted into the designated box, and subsequently entered to proceed with the decryption process.

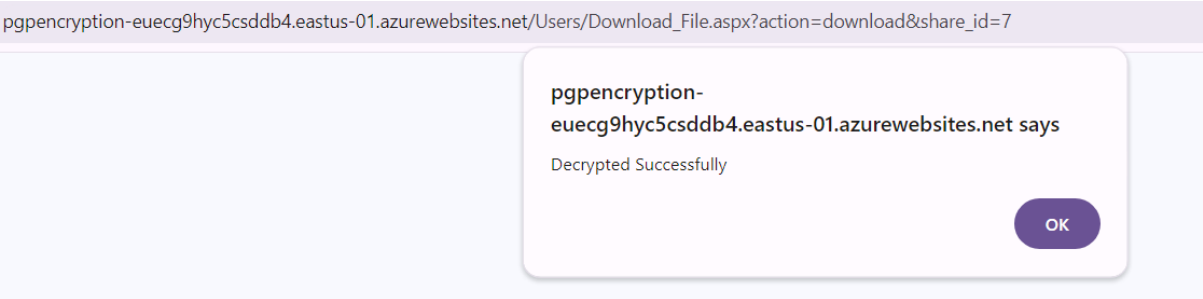


Figure 27: A pop-up message appears, indicating that the decryption process has been successfully completed.

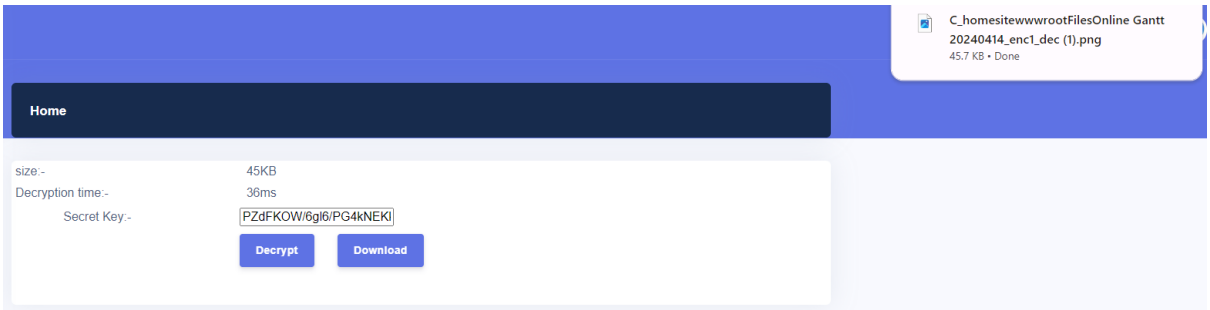


Figure 28: By clicking the Download button, the shared medical files are successfully obtained and made available for retrieval.

6 Evaluation

Cryptography is the field of study that involves using the mathematical principles to secure data through encryption & decryption processes. It provides the reliable method to protect sensitive information or transmit it securely across vulnerable networks.

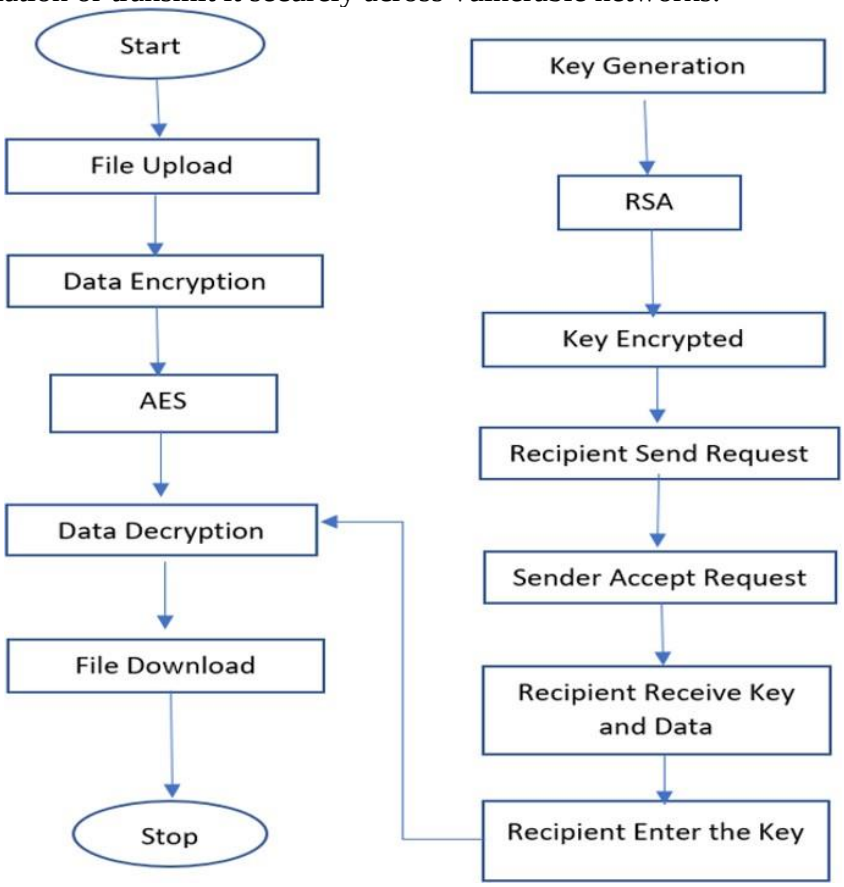


Figure 29: Hybrid crypto system flowchart

This method ensures that only the intended recipient can access the data making it ideal for securing data transmitted over insecure networks. Numerous cryptographic algorithms have been developed &researched by institutions worldwide. This research presents a performance analysis framework for cryptographic algorithms specifically focusing on the AES & RSA algorithms. The effectiveness of these algorithms along with the hybrid cryptographic algorithm, is evaluated based on two factors namely time & file size.

File Size (MB)	AES (Time in Sec)	Hybrid (Time in Sec)
1	5.093	4.493
2	12.28	10.240
4	23.289	18.59
8	46.295	37.263

Table 3: Encryption Time between AES Hybrid Algorithm

The encryption time, which measures the time taken by the algorithm to convert plain text into cipher text that can be determined for different file sizes. The decryption time representing the time taken to recover plain text from cipher text is also considered. The aim is to achieve maximum security through an efficient approach that can perform operations quickly. By combining the strengths of these algorithms this hybrid approach offers enhanced security & faster execution compared to using a single high-security algorithm.

6.1 Results for Algorithm Used

The proposed system utilizes the AES and RSA algorithms to ensure data security. The solution incorporates a hybrid approach combining both RSA and AES. We conducted tests on files of various sizes, ranging from 1 to 8 megabytes and 100 to 800 kilobytes, to evaluate the speed of data execution by these algorithms. The hybrid method and other algorithms were implemented using the .Net programming language, and testing was carried out on a system with an Intel Core i5 processor running at 2.50 GHz and 16 gigabytes of RAM.

File Size (KB)	RSA (Time in Sec)	Hybrid (Time in sec)
100	1.493	1.393
200	2.814	2.240
400	5.589	5.134
800	8.195	7.263

Table 4: Encryption Time between RSA Hybrid Algorithm.

File Size (MB)	AES (Time in Sec)	Hybrid (Time in sec)
1	7.093	5.493
2	16.28	15.240
4	33.289	28.59
8	40.295	35.263

Table 5: Decryption Time between AES Hybrid Algorithm.

In terms of file encryption, the suggested technique demonstrates the shortest execution time, as shown in Figure 30. The system combines symmetric and asymmetric cryptography techniques, running them simultaneously. Compared to the current system, the hybrid algorithm achieves a reduction of 17 to 20 percent in data file encryption time. Using a single algorithm in a public cloud environment may compromise data security. Figure 31 illustrates that the current system requires 15 to 17 percent more time for file decryption compared to the hybrid technique. The AES algorithm exhibits the fastest decryption time but offers less data protection. Increasing the key size in AES naturally extends the encryption and decryption time. When compared to alternative symmetric techniques, the standard algorithm used in the proposed approach requires the least amount of time for file encryption as depicted in Figure

[30]. The suggested hybrid technique employs a secret key for both data encryption and decryption. [Figure 31] shows that the proposed approach for file decryption is 10 to 12 percent faster than RSA. In a hybrid approach, file decryption takes more time compared to encryption. The RSA algorithm requires less time for data file decryption when compared to the AES technique. Overall, the RSA technique exhibits the longest decryption time among the tested algorithms. The findings demonstrate the advantages of the hybrid approach, which offers a balance between speed and data protection. By combining the strengths of AES and RSA the proposed system achieves efficient encryption and decryption processes ensuring data security in cloud computing environments.

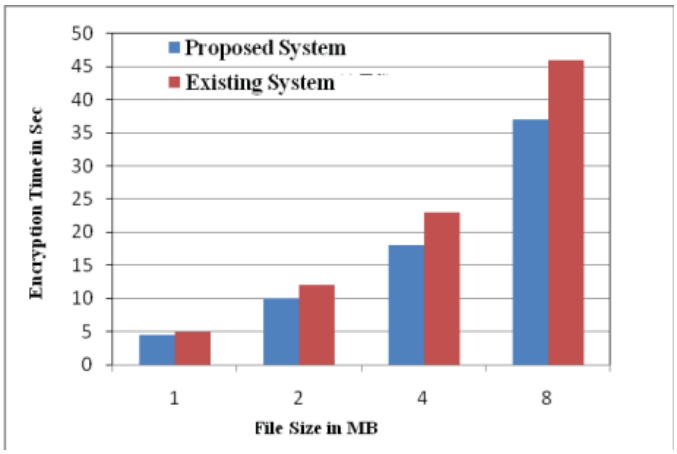


Figure 31: Comparison of Encoding Time of Hybrid Model to AES

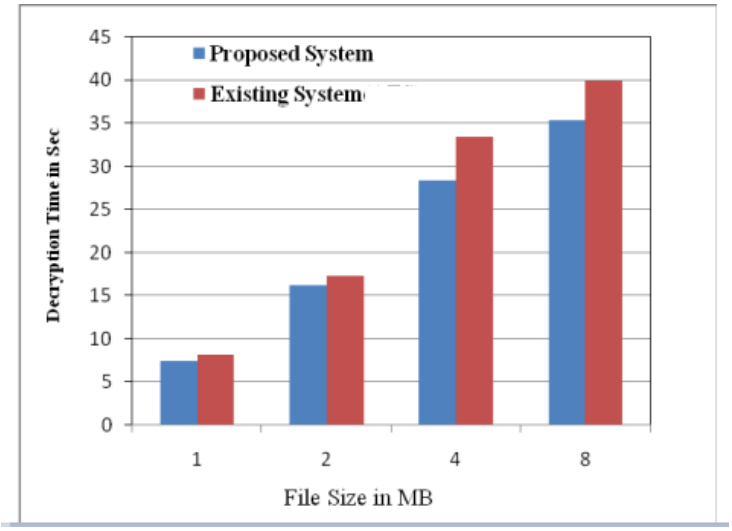


Figure 30: Comparison of Decoding Time of Hybrid Model to AES

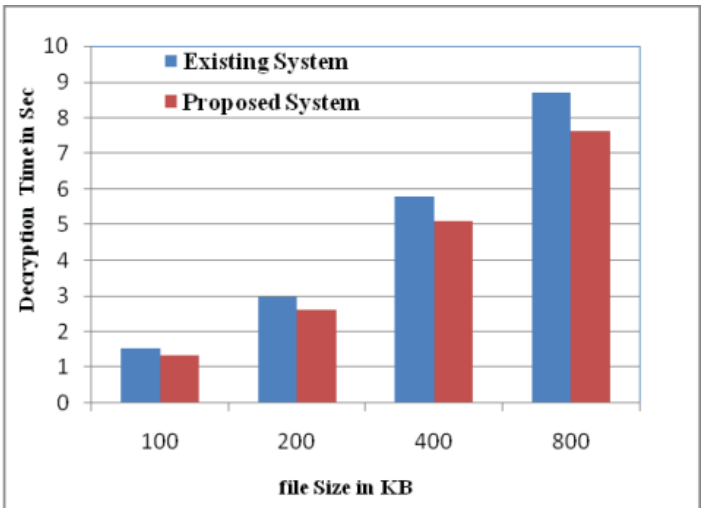


Figure 32: Comparison of Decoding Time of Hybrid Model to RSA

6.2 Security Analysis

6.2.1 Protection against computational attacks

In the hybrid cryptographic model, this research employed two distinct algorithms for encrypting both the data and the key. This proposed approach significantly strengthens the security of both the data and the key. Resulting the data remains protected and immune to potential attacks.

6.2.2 Protection against Side-Channel Attacks

The security of the AES algorithm relies on a complex round key operation. Each operation is performed using distinct coordinates requiring multiple computations to complete the full cycle. This characteristics of AES provides protection against different cyber attacks which ensures the integrity as well as robustness of the encryption process.

7 Conclusion and Future Work

This research aimed to address the critical challenge of protecting the privacy of electronic health records (EHRs) in cloud-based systems by developing a hybrid cryptographic technique based on Pretty Good Privacy (PGP). The primary objective was to ensure that EHR owners retain full control over their private health data while enabling secure patient data exchange. To achieve this proposed scheme combined symmetric encryption using the Advanced Encryption Standard (AES) & asymmetric encryption using the RSA algorithm. This approach was designed to store healthcare data in an encrypted format on third-party cloud infrastructure therefore maintaining data privacy & simplifying key management while ensuring user anonymity.

The research successfully answered the research question by developing as well as implementing the proposed hybrid cryptographic technique. The objectives were met as the technique effectively enhanced the privacy of health information in the cloud provided secure access to large textual , image based datasets & improved overall performance by reducing the time as well as cost associated with encryption.

Key findings from this research indicate that the proposed hybrid cryptosystem offers a robust solution for securing EHRs in cloud environments. It achieves a balance between security, efficiency & usability, making it a feasible option for healthcare organizations. The implementation of this technique ensures that sensitive data remains protected during storage & transmission, therefore meeting the stringent security & privacy requirements of modern healthcare systems.

However, the research also identified certain limitations. The current implementation is limited to the encryption & decryption of individual files which may not be sufficient for managing the increasingly large datasets common in healthcare. Additionally while hybrid approach simplifies key management it may still introduce computational overhead particularly in large scale deployments.

Looking ahead future research could focus on meaningful advancements beyond merely sweeping more parameters in the current model. First potential direction is the development of a more sophisticated system that supports the encryption & decryption of multiple files simultaneously. So enhancing the efficiency of this method & usability in the real world scenarios. Exploring alternative cryptographic algorithms that could further reduce computational overhead without compromising security could be a valuable extension of this work.

Another promising path for the future research is the integration of the proposed cryptosystem with emerging technologies such as blockchain which could provide additional layers of security & transparency in the management of electronic health records (EHRs). Investigating the potential for commercializing the proposed solution particularly in the healthcare markets where data security is a significant concern that could provide practical benefits & drive further innovation in this proposed field. By exploring these , the follow up research could build on the current work offering the improved solutions for protecting the sensitive healthcare data in increasingly complex cloud environments.

References

Agarwala, A., Singh, P., and Atrey, P. K. (2017). Dice: A dual integrity convergent encryption protocol for client-side secure data deduplication. Proceedings of the 2017 IEEE *International Conference on Systems, Man, and Cybernetics (SMC),* pp. 1099-1104.

Domadiya, N., and Rao, U. P. (2022). Elgamal homomorphic encryption-based privacy-preserving association rule mining on horizontally partitioned healthcare data. Journal of The *Institution of Engineers (India):* Series B 103(3): 497–507.

Farzana, S., and Islam, S. (2019). Symmetric key-based patient-controlled secured electronic health record management protocol. Journal of *High-Speed Networks 25*(2): 101–114.

John, A. O., Shola, P., and Philip, S. (2015). Comparative analysis of discrete logarithm and RSA algorithm in data cryptography. International Journal of *Computer Science and Information Security* 13(1): 21–28.

Lin, H.-Y., and Jiang, Y.-R. (2020). A multi-user ciphertext policy attribute-based encryption scheme with keyword search for a medical cloud system. *Applied Sciences* 10(12): 4312.

Liu, X., Yang, X., Luo, Y., Wang, L., and Zhang, Q. (2020). Anonymous electronic health record sharing scheme based on *decentralized hierarchical attribute-based encryption in a cloud environment*. IEEE Access 8: 115219–115229.

Lu, H., Chen, J., and Zhang, K. (2021). Verifiable dynamic searchable symmetric encryption with forward privacy in cloud-assisted e-healthcare systems. Proceedings of the *International Conference on Algorithms and Architectures for Parallel Processing*, pp. 39–48.

Mythri, G., and Jayram, B. G. (2017). Feature-based encryption for data privacy and access control for medical applications. Proceedings of the 2017 *International Conference on Current Trends in Computer, Electrical, Electronics, and Communication*, pp. 435–440.

Nadaf, S. J., and Patil, R. (2016). Cloud-based privacy-preserving secure health data storage and retrieval system. Proceedings of the 2016 *International Conference on Inventive Computation Technologies (ICICT)*, pp. 1–5.

Obiri, I. A., Xia, Q., Xia, H., Affum, E., Abla, S., and Gao, J. (2022). Personal health records sharing scheme based on attribute-based signcryption with data integrity verifiable. *Journal of Computer Security* 30(4): 545–568.

Oh, S. E., Chun, J. Y., Jia, L., Garg, D., Gunter, C. A., and Datta, A. (2014). Privacy-preserving audit for broker-based health information exchange. Proceedings of the 4th ACM conference on *Data and Application Security and Privacy*, pp. 103–112.

Owolabi, O. Y., Shols, P., and Jibrin, M. B. (2017). Improved data security system using a hybrid cryptosystem. *IJSRSET* 3(3): 201–209.

Pariselvam, S., and Swarnamukhi, M. (2019). Encrypted cloud-based personal health record management using the DES scheme. Proceedings of the 2019 IEEE International Conference on *System, Computation, Automation, and Networking (ICSCAN)*, pp. 1–6.

Shabbir, M., Shabbir, A., Iwendi, C., Javed, A. R., Rizwan, M., Herencsar, N., and Lin, J. C.-W. (2021). *Enhancing security of health information using the modular encryption standard in mobile cloud computing*. IEEE Access 9: 27905–27917.

Sharma, K., Agrawal, A., Pandey, D., Khan, R., and Dinkar, S. K. (2019). RSA-based encryption approach for preserving the confidentiality of big data. *Journal of King Saud University-Computer and Information Sciences* 31(4): 553–561.

Thakur, J., and Kumar, N. (2011). DES, AES, and Blowfish: Symmetric key cryptography algorithms simulation-based performance analysis. *International Journal of Emerging Technology and Advanced Engineering* 1(2): 6–12.

Xu, C., Wang, N., Zhu, L., Sharif, K., and Zhang, C. (2019). Achieving searchable and privacy-preserving data sharing for a cloud-assisted e-healthcare system. IEEE *Internet of Things Journal* 6(5): 8345–8356.

Zhang, Q., and Ding, Q. (2015). Digital image encryption based on the Advanced Encryption Standard (AES). Proceedings of the 2015 Fifth International Conference on Instrumentation and *Measurement, Computer, Communication, and Control (IMCCC)*, pp. 54–58.