RSA AND AES BASED HYBRID ENCRYPTION TECHNIQUE FOR ENHANCING DATA SECURITY IN CLOUD COMPUTING

RIMA AKTER

MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

RSA AND AES BASED HYBRID ENCRYPTION TECHNIQUE FOR ENHANCING DATA SECURITY IN CLOUD COMPUTING

RIMA AKTER

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Master of Science in Information and Communication Engineering

Department of Information and Communication Engineering

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

AUGUST 2022

SUPERVISOR'S DECLARATION

We hereby declare that we have checked this thesis and in our opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Master of Science in Information and Communication Engineering.

_____

Name of Supervisor: DR. MD. ASHIKUR RAHMAN KHAN

Position: PROFESSOR & CHAIRMAN

Date:

_____

Name of Co-Supervisor: SULTANA JAHAN SOHELI

Position: ASSISTANT PROFESSOR

Date:

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is my own except for quotations, references and summaries which have been duly acknowledged. This thesis has not been accepted for any degree and is not concurrently submitted for award of other degree.

_____

Name: RIMA AKTER

Roll: BFH1911MSc109F

Date:

Dedicated to my Parents

ACKNOWLEDGEMENTS

I am grateful and would like to express my sincere gratitude to my supervisor Professor Dr. Md. Ashikur Rahman Khan for his germinal ideas, invaluable guidance, continuous encouragement and constant support in making this research possible. He has always impressed me with his outstanding professional conduct, his strong conviction for science, and his belief that a M.Sc. program is only a start of a life-long learning experience. I appreciate his consistent support from the first day I applied to graduate program to these concluding moments. I am truly grateful for his progressive vision about my training in science, his tolerance of my native mistakes, and his commitment to my future carrier. I also would like to express very special thanks to my co-supervisor Sultana Jahan Soheli for his suggestions and co-operations throughout the study. I also sincerely thanks for the time spent proofreading and correcting my many mistakes.

My sincere thanks go to all my lab mates, members of the staff of the Information and Communication Engineering Department, Noakhali Science and Technology University (NSTU), who helped me in many ways and made my stay at NSTU pleasant and unforgettable.

I acknowledge my sincere indebtedness and gratitude to my parents for their love, dream and sacrifice throughout my life. Special thanks should be given to my committee members. I would like to acknowledge their comments and suggestions which were crucial for the successful completion of this study.

ABSTRACT

The cloud computing security has become the most important problems in its growth. Encryption has been developed as a solution and plays an important role in the security of information systems in cloud computing. Many methods are required to secure the shared data. The advanced internet, networking firms, health information and the cloud applications have significantly increased our data every minute. The current work focuses on cryptography to provide the protection for sensitive data that exchanged between personal users, companies, organizations, or in the cloud applications and others during the transfer of data across the network. Firstly, Data sent from sender to network receiver must be encrypted using the cryptographic algorithm. Secondly, the recipient shows the original data using the decryption technique. In this paper, we have proposed a model to use a hybrid encryption and decryption process based on AES-128 and RSA algorithm. Furthermore, we used HMAC algorithm to ensure the integrity and authenticity of data. Our experiment work has been done to explain the time required and throughput for encryption and decryption of three encrypting algorithms such as AES, RSA and hybrid algorithms based on different size of data and their efficiency is compared. Results of the experiments show that the hybrid algorithm is better in term of security.

Keywords: Cloud Security, Rivest, Shamir, and Adleman (RSA), Advanced Encryption Standard (AES), Hash-Based Message Authentication Code (HMAC), Hybrid Algorithm, Encryption, Decryption.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Today's cloud computing (CC) is a rapidly evolving technology that is in use all over the world. It is a new and emerging idea that makes use of virtual computing services over the internet or network to access a variety of shared resources, including servers, storage, interfaces, networks, services, and applications that can deploy, allocate, or reallocate computing resources dynamically and manage and monitor resource usage constantly. With the help of the cloud, you can access resources online at any time and from any location. Additionally, a cloud computing environment enables the sharing of resources among servers, users, and persons, making files and data stored there publicly available. As a result, it can be used by a variety of sectors, including those in the healthcare, finance, business, organization, social network, and educational sectors.

The most significant issues with cloud computing's expansion are its security measures. Between the cloud computing platform and the user, there are a lot of data interactions [1]. As a result, we must concentrate on data storage and transfer. Data security is now required in a variety of ways due to the advancement of information technology sciences as well as the value of data. With the advancement of information technology and the use of Internet networks through data interaction between various parties, organizations, and individuals, insecure data is now regarded as one of the concerns. Information security is a word used to describe a process used to meet security criteria [2]. One technique to guarantee data confidentiality and access to the receiving party without risk of interference from outside parties is through encryption. More broadly, cryptography includes the creation and study of algorithms that prevent third party or public reading of private messages [3].

Cryptographic algorithms come in two different varieties. The first one is public cryptography (PKC), or an asymmetric cipher, which employs two keys, one for sender and receiver usage and one for receiver use only. The second is a symmetric cipher, which

encrypts and decrypts data using the same key for both the sender and the recipient. Data that has been encrypted remains private by hiding the user's confidentiality [4]. In this article, a hybrid encryption algorithm is designed and built based on AES (Advanced Encryption Standard) + RSA (Rivest, Shamir, and Adleman) encryption algorithm. In order to ensure that user data is transferred securely, it uses double encryption technology in combination with hash-based message authentication code (HMAC) technology. Multiple cipher types, typically based on differing strengths, are used in a hybrid cryptography. The plan is to create a unique encryption key, which will then be encoded using the participant's public key. To secure the original material and ensure that the data is kept secret from any external assault until the recipient receives these data, the encrypted key is transmitted to the recipient along with the cipher text. This algorithm making some improvements that is increasing the level of protection for these data.

1.2 Problem Statement

As a communications and transmission of data over networks has increased exponentially since the last few years, there is need of security in such data transfer. With the use of cloud computing, an organization can access services, and data will be stored in physical locations controlled by vendors rather than the company. It has a huge interactions number of data between the platform of cloud computing and user. In the process of data transmission, the data which transfer between the users and cloud computing could possibly be intercepted, it may lead in to leaked of secret data for the enterprise. The ability of the cloud to create a variety of security concerns and issues, such as data confidentiality, integrity, and privacy, meant that every company needed or desired a reliable cloud computing environment where the system or service providers protect the data's secrecy.

Data is at considerable risk if security measures are not appropriately supplied for data operations and transmissions [5]. There is a chance of having a large data risk because cloud computing enables a group of users to access the stored data. It takes a system that can carry out verification, authentication, and encrypted data transmission, so it should retain the confidentiality of the data, to foster that level of trust in cloud computing. By identifying security concerns and methods to address them, the strongest security measures may be put into place. In order to ensure the security of data in entire technology of cloud computing, the service provider and client must encrypt the data. Furthermore, to avoid data leakage and loss the network between the client and cloud must be controlled and protected well. The cloud

providers have started to enhance and improve this aspect. Figure 1.1 [6] makes it obvious that Data Security and Privacy are the most crucial and vital factors to take into account.

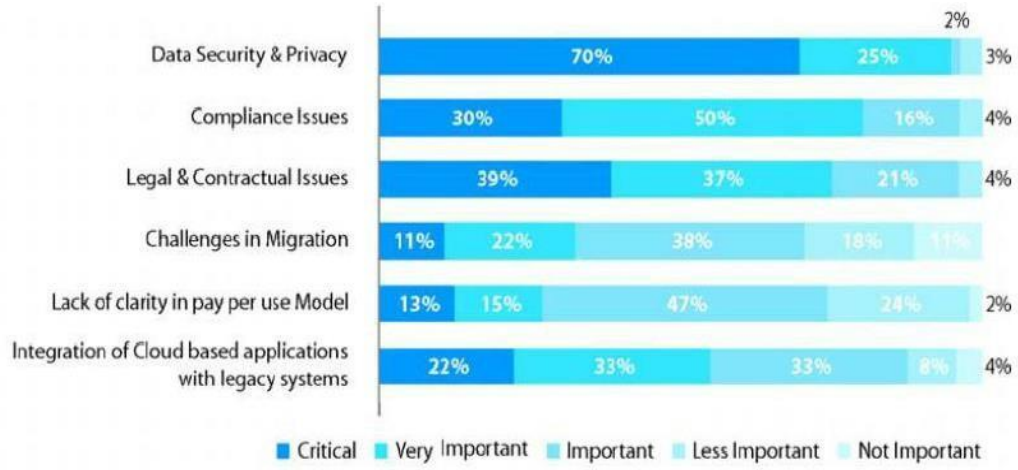

Figure 1.1 Data security and privacy - major inhibitor to cloud adoption.

1.3 Research Objectives

The objective of this study is to develop a system that will provide Security and Privacy to Cloud Storage. To achieve this objective the following aims will have to materialize:

i. To develop a hybrid encryption technique for protecting data in the cloud.
ii. To increase the confidentiality and security of data.
iii. To mitigate security issues at the particular level authentication and storage level in cloud computing.
iv. To enhance security for secure data transmission.
v. To overcome the challenges in cloud computing security.

1.4 Outline of the Report

The thesis is written in five chapters including the introduction. The following is a brief description of these chapters:

Chapter 2: the theoretical basis of the cloud computing architecture and security implications is presented. And some literature survey related to it are provided and discussed.

Chapter 3: the thesis's concept is explained in details and presents the detailed information about the architecture and working procedure of the proposed work.

Chapter 4: illustrates the implementation work of the proposed system and discusses the outcome.

Chapter 5: provides the conclusion and future work related to the proposed system.

CHAPTER 2

LITERATURE REVIEW

This chapter discusses the fundamental basics of cloud computing technology and issues related to security, mainly privacy, in cloud computing services. Understanding the cloud computing concept and how this concept is implemented in different service delivery and deployment models helps to identify the security issues facing this new technology. In this chapter the service delivery and deployment models of cloud computing services are described in Sections 2.1 and 2.2. In Section 2.3 and 2.4, challenges and issues in cloud computing are investigated in general and some specific examples are provided. Some literature survey related to it are provided and discussed in Section 2.5 and in Section 2.6 we discussed the performance evaluation of three of the most widely debated encryption algorithms.

2.1 Cloud Service Models

Unquestionably, cloud computing offers a variety of hosted services over the internet. These hosted services can be broadly categorized into three major service models [7], namely Infrastructure as a Service (IaaS), Platform as a Service (Platform as a Service), and Software as a Service (SaaS), which have been explored below:

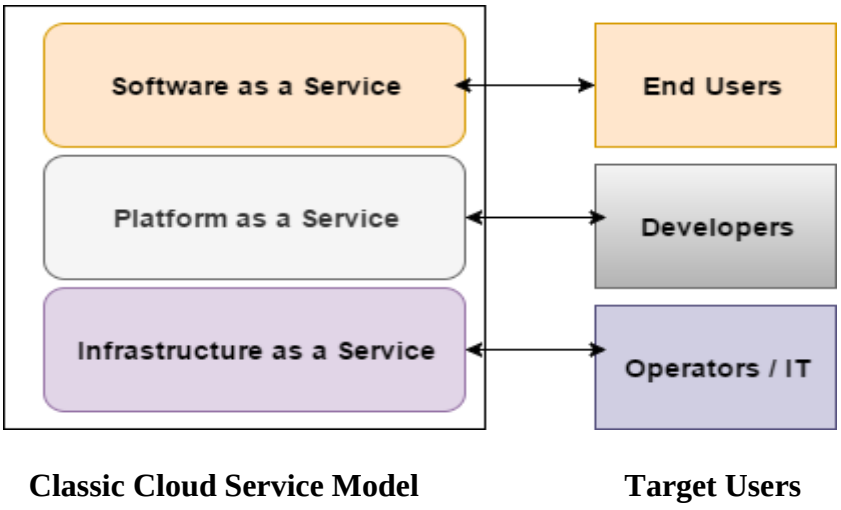

Figure 2.1 Cloud service delivery models.

The cloud service delivery models, as in Figure 2.1 [8], include:

- **Infrastructure-as-a-Service (IaaS)**: Infrastructure as a Service, sometimes abbreviated as IaaS, contains the basic building blocks for cloud IT and typically provide access to networking features, computers (virtual or on dedicated hardware), and data storage space. With the most flexibility and management control over your IT resources, Infrastructure as a Service is most comparable to the current IT resources that many IT departments and developers are accustomed to today.

- **Platform-as-a-Service (PaaS)**: Platforms as a Service allow you to concentrate on the deployment and administration of your applications by removing the need for companies to manage the underlying infrastructure (often hardware and operating systems). As a result, you can run your application more efficiently as you won't have to deal about things like resource acquisition, capacity planning, software maintenance, patching, or any other undifferentiated heavy lifting.

- **Software-as-a-Service (SaaS)**: With software as a service, the service provider gives you a finished product that is operated and managed on your behalf. The majority of the time when software as a service is mentioned, end-user applications are meant. With a SaaS solution, you only need to consider how you will utilize that specific piece of software; you do not need to consider how the service is managed or how the underlying infrastructure is maintained. Web-based email is a typical example of a SaaS application since it allows you to send and receive emails without having to manage feature updates or upkeep for the servers and operating systems that the email program is using.

  The construction of a common security model for each service delivery model is complicated by the various implementations that each service delivery model may take, as shown in Figure 2.1. Additionally, it's possible for multiple service delivery models to coexist on a single cloud platform, further complicating the security management procedure.

2.2 Cloud Deployment Models

A paradigm for providing various services in an on-demand delivery manner is cloud computing. The four types of cloud computing deployment models—Public cloud, Private

cloud, Community cloud, and Hybrid cloud—are categorized to coexist peacefully with other components [7] based on the cloud services and features already discussed.

• **Public Cloud:** a cloud platform where anyone may sign up and access the available infrastructure.

• **Private Cloud:** a cloud platform is intended for a particular business.

• **Community Cloud:** the infrastructure of the cloud is shared by several organizations and supports a specific community with shared concerns.

• **Hybrid Cloud:** a cloud infrastructure that is a composition of two or more clouds i.e. private, community or public.

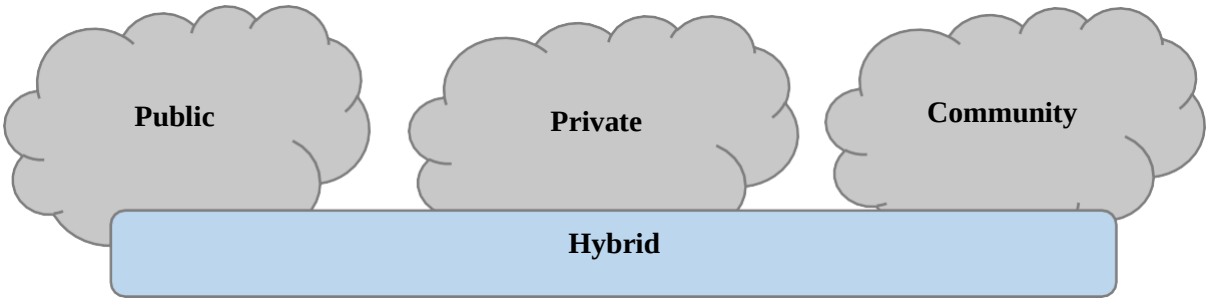

Figure 2.2 Cloud deployment models.

2.3 Challenges to Cloud Security

For IT leaders, security has been one of the most difficult problems, especially when using clouds. Numerous security concerns exist and are keeping businesses from utilizing the cloud's benefits. Security is described as the main level challenge for cloud users in a number of studies, including the one by Amit Sangroya et al. [9]. This section contains a taxonomy of cloud computing security terms.

Figure 2.3 [10] is a schematic diagram of the cloud computing hierarchy that highlights the security vulnerabilities that might arise with both the deployment and service models as well as network-related problems. The categorisation presented above highlights a number of typical problems with cloud computing. The deployment approach is further divided into Private, Public, and Hybrid Clouds, and these models' security flaws have all been discovered. The Service model is further divided into the SaaS, PaaS, and IaaS briefings, each of which highlights the security problems they share. The difficulties with network

security are also demonstrated by the fact that networks are the foundation of all internet-based services, including cloud computing.

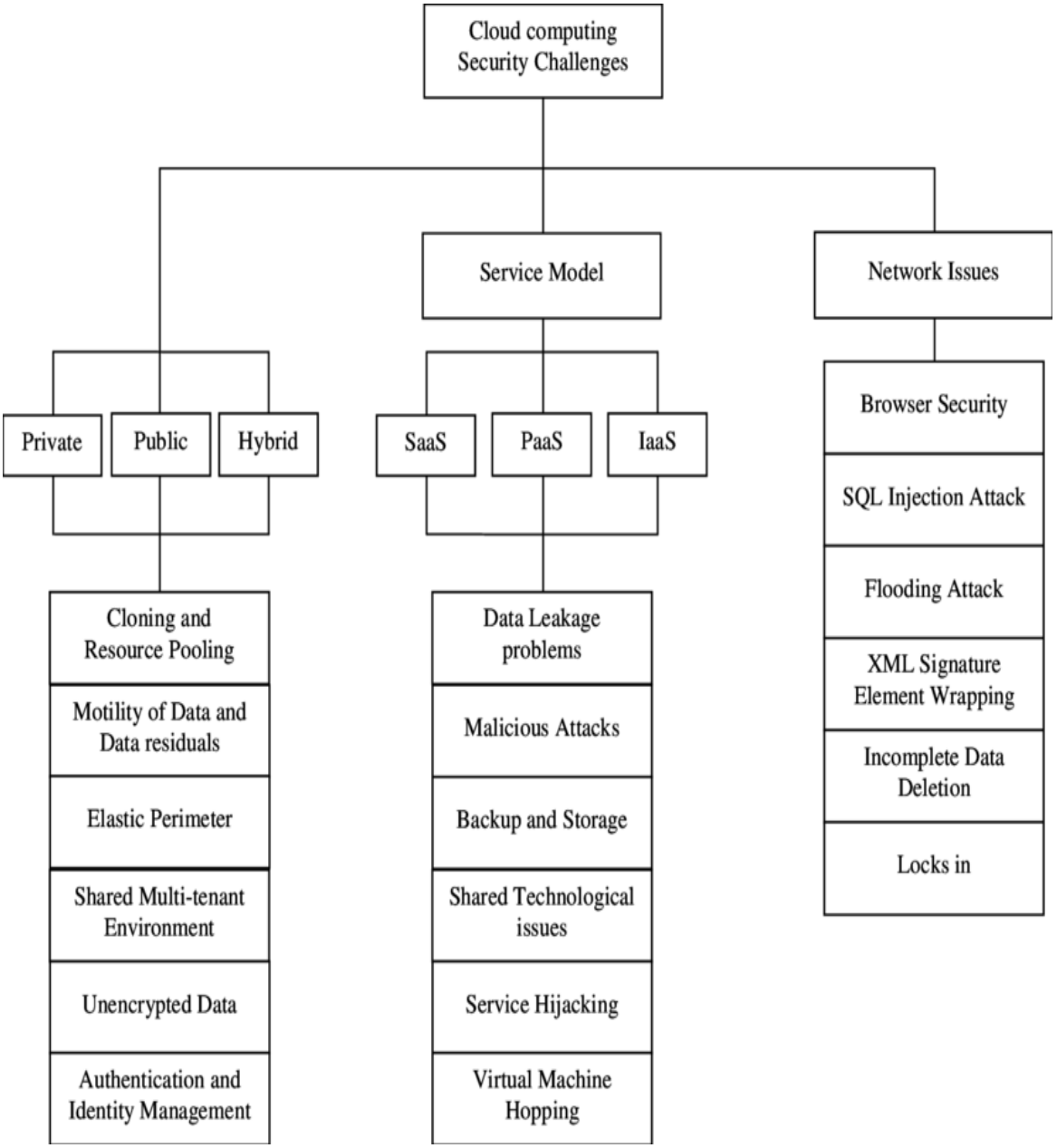

Figure 2.3 Cloud Computing Security Challenges.

2.4 Cloud Computing Security

Wikipedia defines Cloud Computing Security as “Cloud computing security (sometimes referred to simply as "cloud security") is an evolving sub-domain of computer security, network security, and more broadly information security [11]. It refers to a broad set of

policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing."

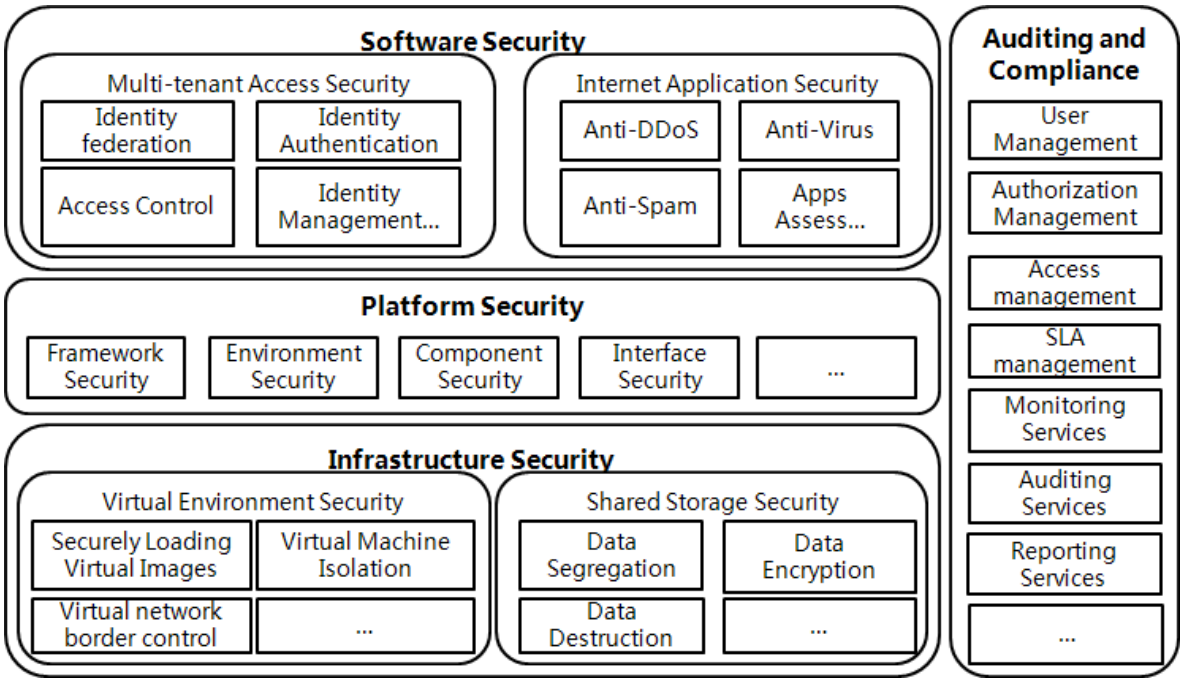

Figure 2.4 Cloud Computing Security Architecture [12].

Data misuse is possible when multiple organizations share resources. Therefore, it is essential to protect data repositories as well as the data that is involved in processing, transport, or storage in order to reduce risk. The most significant difficulties in cloud computing are related to data protection. It's crucial to offer authentication, authorization, and access control for cloud-stored data in order to improve security in cloud computing. There are two primary facets to data security:

i. **Authentication:** The security of the cloud is significantly influenced by authentication. Authentication is the process used to assess user identities with confidence. The degree of authentication assurance must be correct and appropriate given the sensitivity of the application, the information assets accessible, and the risk present [13]. The submitted credentials are compared with those stored in a database of instructive data about authorized users on a local operating system or within an authentication server during the authentication process [14]. If the credentials match, the process is finished and the person is given access rights. To control who has access to data, authorization is used [15]. It's the process by which a method decides how much access a particular authenticated user should have to the system's secured resources.

ii. **Confidentiality:** To keep private or confidential data in the cloud, users must ensure data confidentiality. Data confidentiality is typically protected by authentication and access control techniques [16] [17]. By making the cloud more dependable and trustworthy, the problems with data confidentiality, authentication, and access control could be resolved. Access to sensitive and protected data is made possible by confidentiality in the context of computer systems [18]. Certain controls protect data from nefarious intrusion and guarantee confidentiality [19]. The capacity to communicate private information among a group of users while upholding the rights that each member of the group has been given by the owner of the data is known as data confidentiality. Any user who is not a member of the community is deemed to have no privileges at all [20].

2.5 Cloud Security Related Works

Cloud computing security challenges and issues discussed by various researchers. The Cloud Computing Use Cases group [21] discusses the different use case scenarios and related requirements that may exist in the cloud computing model. They consider use cases from different perspectives including customers, developers and security engineers. ENISA [22] investigated the different security risks related to adopting cloud computing along with the affected assets, the risks likelihood, impacts, and vulnerabilities in cloud computing that may lead to such risks. Similar efforts discussed in "Top Threats to Cloud Computing" by CSA [23]. We provide some related work in the area of cloud computing security and cryptographic algorithm [24] used in cloud computing security.

Mahalle & Shahade, 2014 developed hybrid cryptographic algorithm for cloud systems for providing security to data [25]. There hybrid algorithm is mixture of famous symmetric (AES-128) and asymmetric (RSA-1024) encryption algorithm implemented in eye-OS. The file uploaded by cloud user is first stored in temporary storage where it is encrypted using AES-128 and uploaded to the cloud storage system. The secret key utilized for encryption is encrypted using RSA-1024 using the public key which can be recovered by user only. This model can be utilized for large file size due to fact that only AES is used for data encryption which is considered faster and secure too.

Kiruthika et al., 2015 have reviewed the security challenges encounter in cloud services with various stats also mentioned some key issues in cloud security [26]. Their model provides the solution to this key issues using AES encryption. They have compared various symmetric encryption algorithmslike 3DES, DES and RC2 with parameters like

encryption speed and throughput. The idea is simply encrypting the data that is stored in cloud storage with AES encryption. The separate application or software will be used for uploading data to cloud which encrypts the data at user side.

Soman & Natarajan, 2017 developed an enhanced hybrid data security algorithm using various algorithms like AES, ECDSA and SHA256 for providing security to data stored inside [27]. These algorithms are arranged appropriately for securely communicating, uploading and downloading data on cloud. The AES Algorithm is used for confidentiality, ECDSA and SHA256 provides authentication and integrity check.

Bindu et al., 2018 developed hybrid cryptography algorithm using 4 symmetric encryption algorithms (AES, RC6, BLOWFISH and BRA) for secure file storage in cloud environment [28]. The file is sliced into 8 equal parts and encrypted using above algorithms simultaneously with multithreading technique. Batra et al., 2018 developed hybrid encryption algorithm using three different symmetric encryption algorithm for secure file storage in cloud [29]. Hybrid encryption uses all three encryption algorithms for providing enhanced security to cloud storage. The data is partitioned into three equal parts and encrypted separately using these three algorithms RC4, DES and AES respectively. The key shared to receiver using Steganography.

L. Kumar & Badal, 2019 reviewed various hybrid cryptographic models and suggested to use AES and FHE which can overcome security issues like data confidentiality, privacy and integrity [30]. Theauthors also listed and elaborated some major security challenges in cloud which includes Data Confidentiality, Authenticity, Integrity, Authorization, Freshness and Non-repudiation. Zhang et al., 2019 implemented hybrid security model for securing medical patient's data stored in cloud [31]. The authors have modified AES algorithm and named as P-AES which is used in conjunction with RSA toprovide privacy and security to medical data stored in cloud. Their hybrid model can only be used fortext and doesn't work for securing images and videos.

Viswanath & Krishna, 2020 developed a Hybrid encryption framework containing modules like data slicing, encryption, distribution, uploading, indexing, retrieval and merging [32]. Encryption and decryption in their hybrid framework is done using AES and Fiestel network block wise. The sliced data is encrypted, indexed and stored in multi cloud environment.

The Table 2.1 below provides the brief overview of various security models developed for cloud security.

Table 2.1 Overview of hybrid security models of cloud

Security Model Studied	Algorithms Used	Gist
(Mahalle & Shahade, 2014)	RSA and AES	• Every file is encrypted with AES using newlygenerated key which is stored in encrypted form. • RSA is used for encrypting secret key. • System Supports sharing of file.
(Kiruthika et al., 2015)	AES and Key Management Application	• Data on Cloud must be retrieved or uploaded through Application • Application encrypts or decrypts the Data using AES algorithm at user side in conjunction with separate Key management application managed byuser.
(Soman & Natarajan, 2017)	AES, ECDSA with SHA256	• System resolves data confidentiality and integrityissue. • The AES Algorithm used for File Encryption. • ECDSA with SHA256 provide message authentication service.
(Bindu et al., 2018)	AES, BLOWFISH, RC6, BRA and Steganography	• System provides enhanced security using hybridization. • Data file is sliced in parts and encrypted usingthese algorithms. • Key sharing is done through image steganography.
(Batra et al., 2018)	AES, DES, RC4 and Steganography	• System provides confidentiality to data using hybrid algorithm. • Data file sliced into 3 parts and each part is encrypted once with different algorithm as mentioned. • Secret Key distribution is done through imagesteganography.
(Viswanath & Krishna, 2020)	AES and Feistel Algorithm	• Model distributes data to multi cloud for storage. • Model partitioned data in blocks of 256 bit and then encrypted using hybrid algorithm formed using AES and Feistel algorithm. • Model requires indexing, retrieval and merging modules for smooth working.

Aside from this, there are still far too many areas that may use improvement, such as the creation of more effective algorithms that would raise the security of cloud storage. In our research, we thoroughly examined the cloud model to pinpoint the underlying reasons and significant contributing factors in the security issues/problems covered by the earlier work. This will make it easier to comprehend the issue and offer solutions.

2.6 Basic Cryptographic Algorithms

Here we discuss comparative studies and performance evaluation of three of the most widely debated encryption algorithms RSA, AES, and DES on the basis of Ogundoyin, Adebanji & Okeyode studies results [33].

A. RSA (Rivest, Shamir, and Adleman)

Public key cryptography algorithms like the RSA algorithm are very popular. Three mathematicians, Rivest, Shamir, and Adleman, developed RSA. RSA is a widely used asymmetric encryption method that can be applied to data encryption, digital signatures, and key exchange. A variable-size key and an adjustable-size encryption block are used in RSA. The key-pair is made up of a very large integer, n, which is created by multiplying two prime numbers together using unique algorithms.

B. AES (Advanced Encryption Standard)

AES is based on the notion of substitution-permutation network architecture. In AES, the key size is 128, 192, or 256 bits, while the block size is 128 bits. The state, a 4 × 4 column-major order byte matrix, is used by AES. The majority of AES calculations are performed in a finite field. Plaintext input is transformed into ciphertext output by a series of transformation rounds known as the AES cipher.

C. DES (Data Encryption Standard)

Data is encrypted in 64-bit chunks using the DES method. DES accepts 64 bits of plaintext as input and generates 64 bits of cipher text as output. The length of the key is 64 bits. Each of the 264 possible 64-bit permutations, each of which can be either 0 or 1, is created using DES. Two 32-bit blocks, one on the left and one on the right, are split up into each 64-bit block. The DES algorithm transforms a 64-bit cipher block C into a 64-bit message block M. The Electronic Code Book (ECB) kind of encryption is used to individually encrypt each 64-bit block. The two types of DES encryption are Chain Block Coding (CBC) and DES (DES).

2.6.1 Comparison among RSA, AES & DES

Java programming was used to simulate and implement the encryption methods (RSA, AES, and DES) using IntelliJ IDEA. Tables 2.2, 2.3, and 2.4 display the outcomes of the simulation that was run using data with varying file sizes. Table 2.2 displays the Encryption time of the chosen algorithms based on the provided metrics (AES, DES, RSA). AES took 40ms, DES 38ms, and RSA 36ms to encrypt 50 bytes of data when the three algorithms were performed simultaneously. AES took 65 ms, DES 60 ms, and RSA 69 ms to encrypt 500 bytes of data when the three chosen algorithms were executed. A graph comparing the encryption times of the chosen algorithms is shown in Figure 2.5.

Table 2.3 displays the selected algorithms' decryption times (AES, DES, RSA). AES took 37 ms, DES 35 ms, and RSA 34 ms to decode 50 bytes of data when the three chosen methods were used. AES had a decryption time of 61 ms, DES had 58 ms, and RSA had a decryption time of 65 ms when the three algorithms were ran on a data size of 500 b. A graph comparing the decryption times of the chosen algorithms is shown in Figure 2.6. Table 2.4 displays how much memory the chosen algorithms use (AES, DES, RSA). AES consumed 12MB, DES used 08MB, and RSA used 10MB when the three algorithms were tested on data sizes of 50 bytes. AES used 25MB, DES used 17MB, and RSA used 21MB of the 500 bytes of data available when the three algorithms were executed. A graph comparing the memory usage of the chosen algorithms is shown in Figure 2.7.

Table 2.2 Encryption time of the selected algorithms (AES, DES, RSA)

Data Size (Byte)	**Encryption Time (ms)**		
	AES	DES	RSA
50	40	38	36
100	50	46	47
150	60	53	61
200	62	56	65
500	65	60	69

Table 2.3 Decryption time of the selected algorithms (AES, DES, RSA)

Data Size (Byte)	Decryption Time (ms)		
	AES	DES	RSA
50	37	35	34
100	46	43	44
150	57	50	58
200	60	52	62
500	61	58	65

Table 2. 4 Memory utilization of the selected algorithms (AES, DES, RSA)

Data Size (Byte)	Memory Utilization (Byte)		
	AES	DES	RSA
50	12	08	10
100	17	12	15
150	18	14	14
200	23	15	20
500	25	17	21

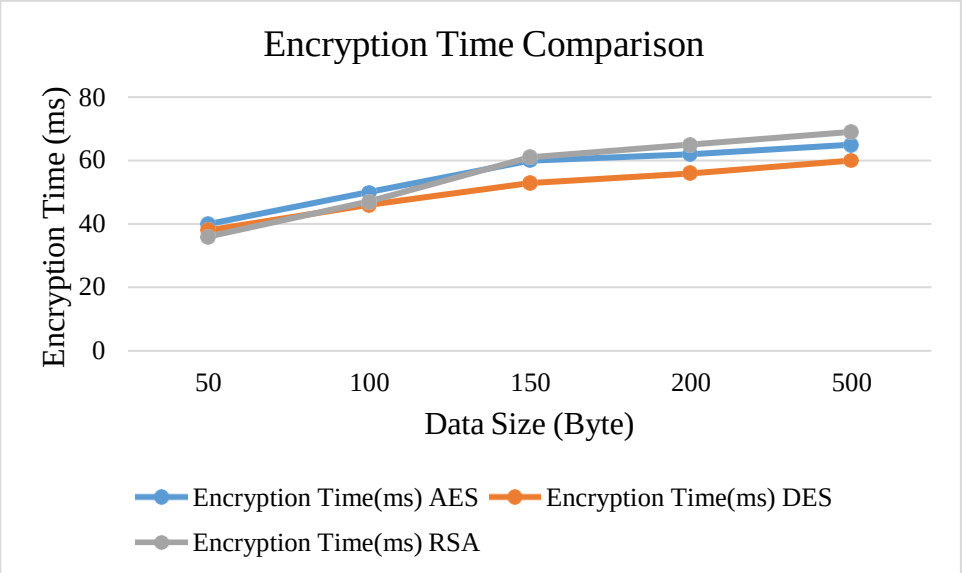

Figure 2.5 Comparative analysis of encryption time of the selected algorithms (AES, DES, RSA).

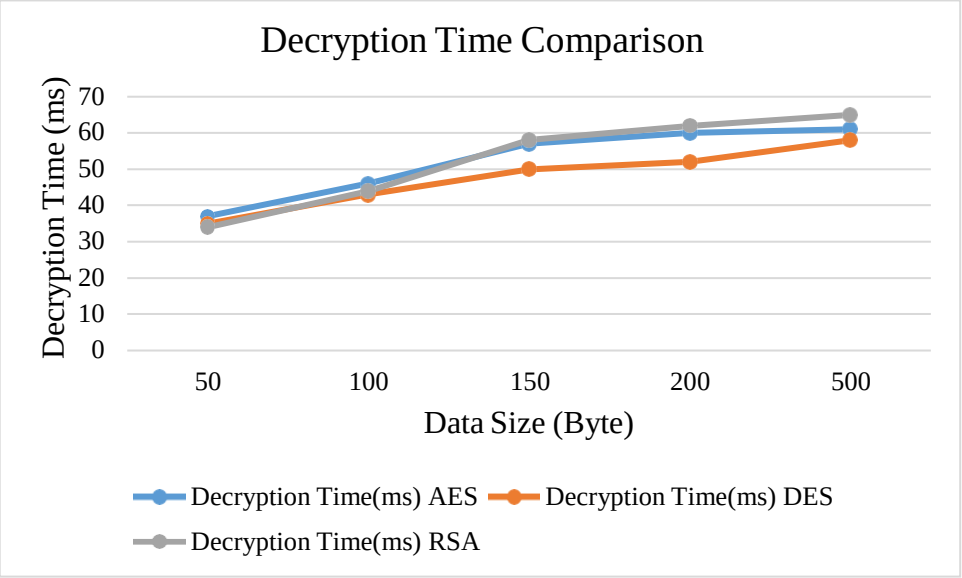

Figure 2.6 Comparative analysis of decryption time of the selected algorithms (AES, DES, RSA).

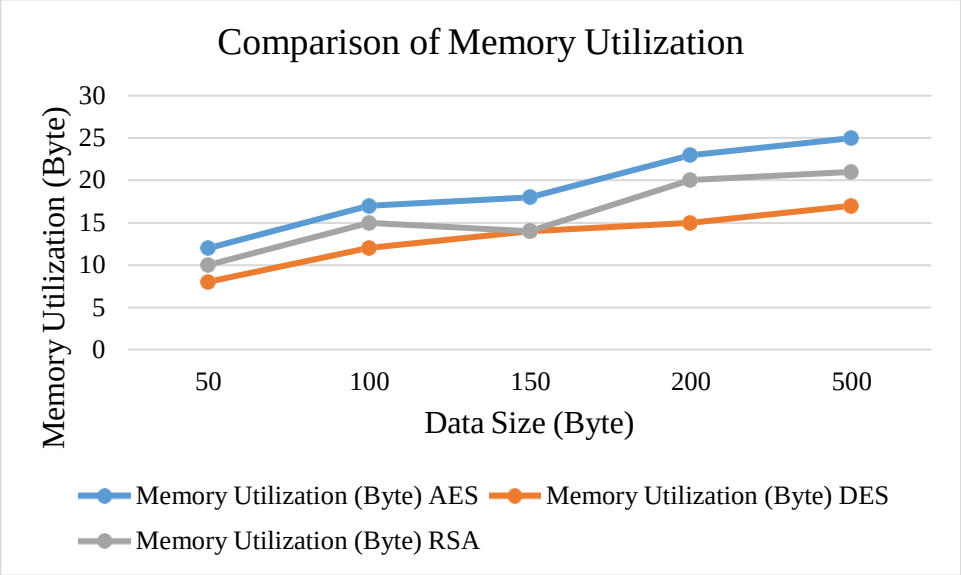

Figure 2.7 Comparative analysis of memory utilization of the selected algorithms (AES, DES, RSA).

When discussing the findings, it was noted that RSA had the best performance in terms of encryption and decryption time for small data sizes. The performances of RSA did, however, gradually deteriorate as the bulk of the data rose. When a data size of 500 Bytes was employed, RSA performed the worst; DES performed the best.

A study of the outcomes also showed that AES only performed well at huge data sizes and that it did not perform much better at small data sizes.

We have suggested a blend of RSA and AES encryption in order to fully leverage the benefits of both algorithms while avoiding their drawbacks. In this study, the data is secured using RSA and HMAC. The recommended methods ensure the integrity and confidentiality of customer data. By employing the suggested system, data encryption and decryption take less time, and using HMAC, data confidentiality and integrity are improved.

CHAPTER 3

RESEARCH METHODOLOGY

The main strategy used in this research is a hybrid approach that combines the strengths of the AES and RSA algorithms to secure data transport over an information network. This part provides a full explanation of the thesis's idea as well as information on the architecture and operation of the suggested work.

3.1 Modules Description

- **User Module**

The user module is used to view two options Encryption and Decryption. The options are selected depends upon the user. If user want to encrypt the file select the encryption option. Otherwise select the decryption option.

- **Encryption Module**

This module used to encrypt the document easily. First of all select the encryption option and then select the encrypting file. Finally click the encryption button. The file is to be encrypted.

- **Decryption Module**

This module used to decrypt the document easily. First of all select the decryption option and then select the encrypted file. Finally click the decryption button. The file is to be decrypted.

3.2 AES Algorithm

Popular symmetric block cipher algorithm is AES. According to the AES standard, the algorithm can only take blocks of 128 bits. The name of the standard is AES-128, AES-192,

or AES-256 depending on the version that is being utilized (Douglas, 2010). Larger data must be divided into blocks in order to be encrypted because AES only supports 128-bit blocks. Four steps make up the AES algorithm's round, which is iterated ten times for keys with a length of 128 bits, twelve times for keys with a length of 192 bits, and fourteen times for keys with a length of 256 bits. A list of key schedule words is then created using the expanded key. The four stages are as follows: Substitute bytes, Shift rows, Mix columns and Add round key. Its algorithm proceeds as follows:

– Given a plaintext x, initialize State to be x and perform an operation ADD ROUNDKEY, which x-or the AddRoundKey with State.

– For each of the first N^{r-1} rounds, perform a substitution operation called SubBytes on State using an S-box;

– Perform a permutation ShiftRow on State; perform an operation MixColumns on State; and perform AddRoundKey.

– Perform subbytes; perform shiftrows; and perform addroundkey.

– Define the ciphertext y to be State.

AES is based on a design principle known as a substitution-permutation network (s-box) (Jigar et Al, 2013). The fact that the S-box is built on inversion in a specific GF representation (28) increases the likelihood of algebraic assaults to ascertain critical bits These assaults try to create and solve equations that relate different input, output, and crucial values (Liam Keliher, 2013). The plan was to integrate AES within the Feistel network in order to reduce algebraic attacks on the AES. As a result, the Hybrid AES-DES algorithm was created to increase security. Feistel use makes exhaustive key searching more difficult, but it may slow the DES cipher because it only has 56 bits (Jigar et Al, 2013). AES uses, Static S-box means the same S-box will be used in each round. Which render it vulnerable to algebraic attacks. Julia Juremi et Al, 2012, proposed that the improvement of S-box to reduce algebraic attacks can be done by the use of key-dependent S-Box approach. The purpose of the proposed approach is to generate the random S-boxes changing for every change of the secret key.

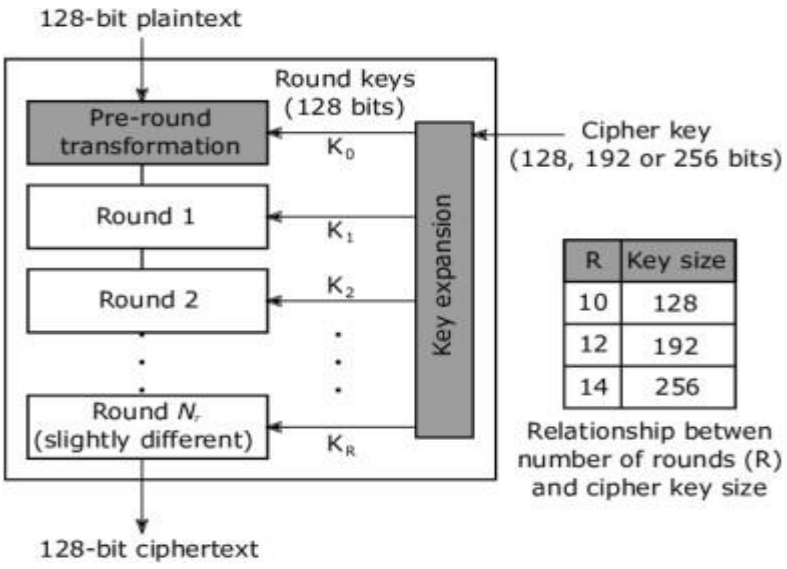

Figure 3.1 AES algorithm grouping and encryption diagram.

3.3 RSA Algorithm

RSA algorithm consist of three phases, namely key generation, encryption then decryption. The figure below gives the description of RSA said phases.

Key Generation

– Select random and secret two large primes p, q and check that p!=q.

– Compute modulus n = pq

– Compute phi, ϕ (n) = (p – 1) (q – 1)

– Select random public exponent: e such as 1 < e < n and gcd (e, ϕ) = 1

– Compute d such as e. d ≡ 1 mod ϕ(n) and 1 < d < ϕ (n)

– Public Key: (n, e)

– Private Key: (n, d)

Encryption Algorithm

– Obtain the recipient's public key (n, e)

– Represents the plaintext message as positive integer m

– Compute the ciphertext c = me mod n

Decryption Algorithm

– Uses his private key (n, d) to compute integer m=cd mod n

– Extract the plaintext from the integer representative m

RSA even if it is efficient algorithm it is vulnerable. Intruder can use brute force method to find private key. It has undergone numerous enhancements to stop private guesses. In 2013, Rajan and Geeta developed a brand-new method for using secure RSA. Secure RSA uses an integer in place of n during encryption and decryption. rather than delivering the e value directly as a public key, the RSA technique is implemented using two public key pairs and some mathematical reasoning. In the event that the private key has been found, RSA still needs to address a number of security flaws. An improvement over this has been implemented to be applied to the first function of SSL that would give more secure communication. The RSA algorithm has been modified from the domain of integers to the domain bit stuffing. The introduction of bit stuffing will complicate the access to the data even after getting the access to the private key. The main disadvantage of this form of bit-stuffing is that the code rate is unpredictable.

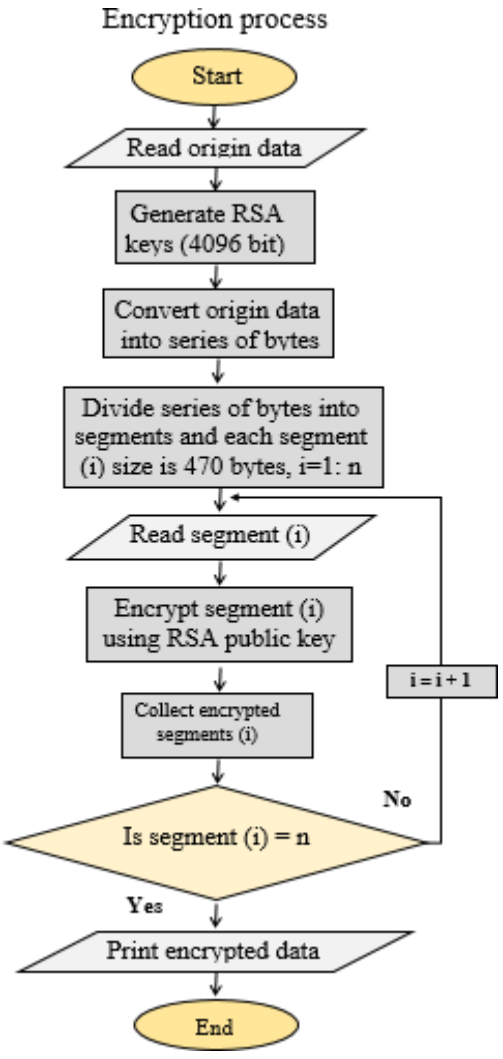

Figure 3.2 RSA encryption algorithm flowchart.

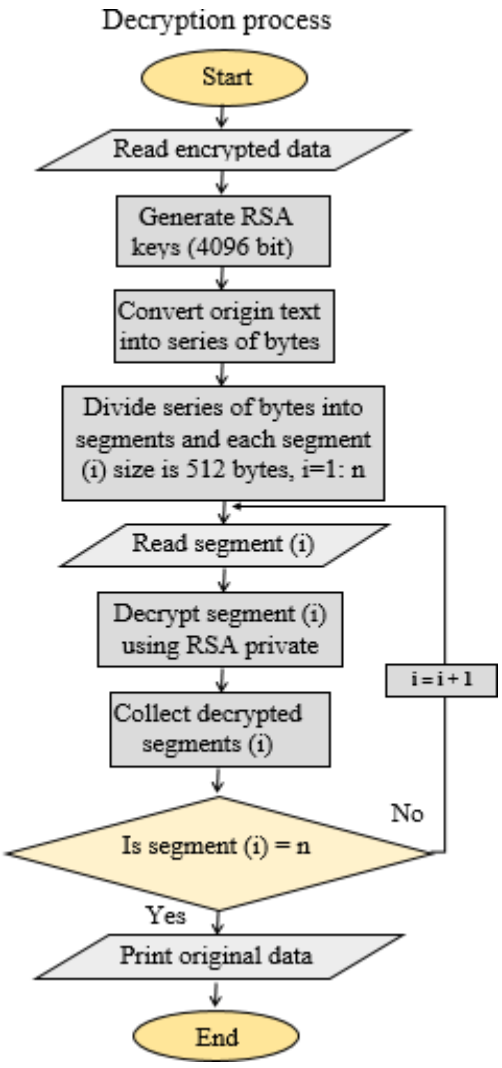

Figure 3.3 RSA decryption algorithm flowchart.

3.4 Hash-Based Message Authentication Code (HMAC)

Two parties want to communicate, but they want to ensure that the contents of their connection remain private. They also distrust the internet, and they need a way to verify that the packets they receive haven't been tampered with. HMAC is a valid solution. Hash-based message authentication code is a tool for calculating message authentication codes using a cryptographic hash function coupled with a secret key.

HMAC keys consist of two parts. These are:

1. **Cryptographic keys:** An encryption algorithm alters data, and a recipient needs a specific code (or key) to make it readable once more. HMAC relies on two sets of keys. One is public, and one is private.

2. **Hash function:** A hash algorithm alters or digests the message once more. HMAC uses generic cryptographic hash functions, such as SHA-1, MD5.

When complete, the message is considered irreversible, and it's also resistant to hacking. Someone who intercepts this message won't even be able to guess at its length. The work renders the message contents absolutely useless to anyone without a key or a code. We can use an HMAC to verify both the integrity and authenticity of a message.

3.5 Proposed Model

The fundamental tenet of our suggested architecture is that, in order to transfer the secret key of AES securely, we first encrypt the plaintext using the AES method before encrypting the secret key using the RSA technique. Thus, we may utilize these two algorithms' essential benefits, such as RSA secret key management's ease and security and AES's high-speed encryption [34] [35]. To further confirm the integrity and validity of the message, we also executed the HMAC function using the secret key and the generated cipher text to attach the generated value to the end of the encrypted message before sending it to the cloud server [36]. This fusion of the strength of AES and RSA algorithms is known as a Hybrid Algorithm.

3.5.1 Hybrid Algorithm

A hybrid encryption approach uses two different types of encryption to protect data transfer: Text files are encrypted using the AES technique, and the AES key is encrypted using the asymmetric RSA algorithm to prevent third parties from confirming the transmission between clients or clients and servers and to make it more difficult for attackers to access. With this approach, the input is the original data, and the output is the encrypted data using the AES key and the AES key encrypted using the RSA public key during a single execution stage rather than in multiple stages. The process of decryption is finished at the receiver. The process of decoding, which yields the original text, is the opposite of that of encoding. The algorithm has three steps to it. They are:

- Key generation
- Encryption and
- Decryption

I. Hybrid Algorithm Key Generation

In this analysis, a key generation algorithm produces the RSA public and private keys in the hybrid algorithm randomly. The following seven steps are carried out in the process of creating the public key and the private key:

1) First, two unequal large prime numbers p and q must be randomly selected.

2) Then calculate the product n of p and q, that is, n = p x q.

3) Calculate the Euler function ø(n) = (p-1)(q1).

4) A positive integer e is randomly selected, and 1 < e < ø(n) is made, and gcd(e, ø(n)) = 1.

5) According to the equation ed = 1 mod ø(n), the result of d is obtained, where 0<d<n

6) According to the formula PU = {e, N}, the public key of the RSA algorithm is saved, where e is a public key.

7) According to the expression PR = {d, p, q}, the private key is saved, where d is the private key.

II. Hybrid Algorithm Encryption Principle

The hybrid encryption algorithm employs two-layer AES and RSA encryption, and the encryption process goes through a series of adjustments and steps. The actions involved in the two methods' file encryption schemes are detailed below, in the order of encryption. The AES algorithm clusters the processing units, and the ordered 128 bit data is assigned to a 4 by 4 state matrix. According to Table 3.1, the AES algorithm's rounds number is determined bythe key length.

Table 3.1 Rounds number according to key size

Key Size	Rounds Number
128-bit	10 Rounds
192-bit	12 Rounds
256-bit	14 Rounds

All transformations in the algorithm are finished and are centered on the state matrix. Four straightforward arithmetic operations—Sub Bytes, Shift Rows, Mix Columns, and Add Round Key—are used in the procedure.

a. Sub Bytes: The sole non-linear byte change in an AES algorithm encryption round is called Sub Bytes, commonly known as s-box permutation. Each byte in the state is determined separately using the substitute table. The unit is obtained as the output with the column value serving as the index from the corresponding place in the s-box using the Sub Bytes mapping approach, which uses the high 4 byte bits as the matrix's row value and the bottom 4 byte bits as its column value.

b. Shift Rows: The ith row of the state matrix is shifted left by i bytes [37] and each row is cyclically pushed to the left in the forward Shift Rows by a row number offset.

c. Mix Columns: Mix Every column in the state is affected by the Columns transform, which treats each column as a polynomial of fourth degree. Each column is independently worked on. The data is combined from all of the state columns to produce additional columns.

d. Add Round Key: When converting Add Round Key, the value obtained is the

   128-bit State X-OR by bit and the 128-bit key. When encrypting the AES key with an RSA algorithm, the plain text is divided into groups, and the binary values of each group m are all less than n, where n is the product of the large prime numbers p and q, e is a random positive integer, and the cypher text c generated can be obtained from the following formula [38]:

$$c = me \bmod n, \text{ and } 0 <= m < n \tag{3.1}$$

III. Hybrid Algorithm Decryption Principle

In the hybrid algorithm, the private key of the RSA algorithm is used to decode the cypher text encrypted by the public RSA key in the first layer, and then the AES key is used to decrypt the cypher text and get the plaintext. As RSA decryption is used, the encrypted cypher text c is decrypted and transformed, and the plain text m is obtained by the following calculation [38].

$$m = cd \bmod n \tag{3.2}$$

Where d is calculated by the key generation algorithm, where n is the product of the large prime numbers p and q.

In the decryption process of the AES algorithm, Sub Bytes, Shift Rows and Mix-Columns are the inverse operations of the encryption process, but in Add Round Key, the inverse operation is the same as the forward transformation because the X-OR operation is its own inverse.

3.6 Details of Model in Encryption Process

A. Ensuring the Confidentiality

Figure 3.4 shows P as our representation of the plaintext. Assuming that the plaintext will be encrypted using the AES-128 technique, the EA stands for the AES encryption algorithm, and the K for the AES secret key. The attacker cannot access the plaintext without knowing the secret key of the encryption technique, as shown in figure 3.4, where CP is the cipher text of original plaintext P [39]. The amount of possible key combinations if unauthorized interceptors attempt to brute force the secret key, which is 128 bits, is obviously 3.4028237e+38, which would take 1021 years to crack using modern high speed computers. Therefore, a brute force assault cannot be used to crack the key [40].

B. Transfer the Secret Key

Sending the secret key via the internet will make it vulnerable to cracking. Therefore, in our suggested architecture, we advise employing the RSA technique to encrypt the secret key. To demonstrate this procedure, we used KEr, the cloud's RSA public key, to encrypt the AES algorithm secret key K and produce the cipher text CK of the AES secret key in the sender side, which could then be sent securely to the receiver side. Allow the encrypted plaintext CP and AES algorithm secret key CK to be sent to the recipient after that via the internet channel [34].

C. Ensuring Integrity and Message Authentication

We stated in Section B earlier that we will provide the CK and CP via the internet channel in order to clarify the secrecy. Additionally, as we want to guarantee the authenticity and integrity of the message, we propose using the HMAC technique to generate an M1 value, attach it to the end of the cipher text CP, and send the entire message to the cloud server. The SHA-256 algorithm is used by the embedded hash function [39].

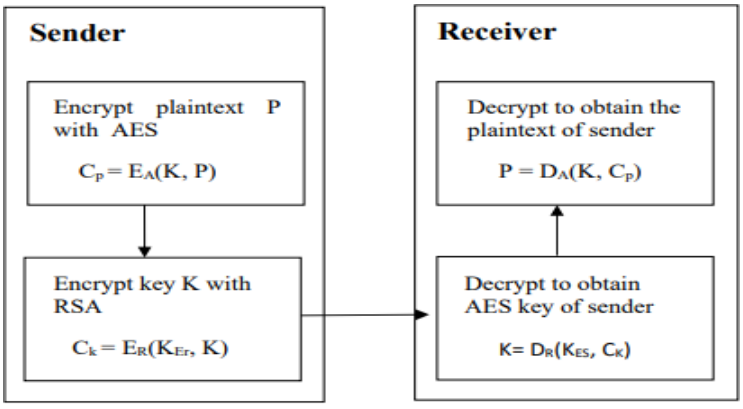

Figure 3.4 Encryption and decryption process of the proposed model.

3.7 Details of Model in Decryption Process

When the cloud server receives the message, it will first use its own RSA private key KEs to gain the AES algorithm secret key K, and then it will retrieve the new value M2 of HMAC using the secret key and cipher text CP to verify the authenticity and integrity of the message. If they are equal, the plaintext is integrated, verified, and must be processed by the cloud server before being stored in its data center. Next, compare the created value of M2 with the received value of M1. In such a case, the plaintext must be ignored [39] [41].

The process of decrypting the cipher text must be started after verifying the message's authenticity and integrity. To do this, the AES method must be used to generate the original plaintext P using the secret key K and the cipher text CP. Figure 3.4 explains the processes for encrypting plaintext and the secret key, as well as for decrypting the secret key and cipher text.

Figure 3.4 uses the abbreviations ER, DR, and DA to denote the RSA encryption and decryption algorithms, respectively. DA stands for the AES decryption algorithm. The most crucial thing in this situation is that both the user and the cloud must remove the trial secret key after the data encryption and decryption processes have been completed successfully. This is done to prevent secret key theft.

3.8 Performance Metrics

The following metrics were used to evaluate the performance of the selected encryption and decryption algorithms:

Encryption time: The time taken to generate a cipher text from plain text is known as encryption time. It was calculated as follow:

$$T = \frac{DS}{S} \tag{3.3}$$

Where, T = Time, DS = Data size, S = Speed.

Decryption time: Decryption time is the time it takes to convert cipher text to plain text. Equation 3.3 is also used to calculate decryption time.

Throughput: The Throughput is based on the Encryption and Decryption time that indicate the communication speed of an encryption scheme. The value of throughput for encryption as well as decryption in Byte/s has been calculated according to the following formula:

$$\text{Throughput} = \frac{Total\ size\ of\ plaintext\ in\ Byte}{Total\ encryption\ and\ decryption\ time\ in\ second}$$

3.9 Flowchart of Proposed Framework

Figure 3.5 outlines the entire encryption and decryption procedure in our suggested paradigm on the user and cloud server sides, which ensure confidentiality, integrity and authentication. The term "internet media" in this context refers to an internet channel where unauthorized users could try to hack into or change the data we have sent through.

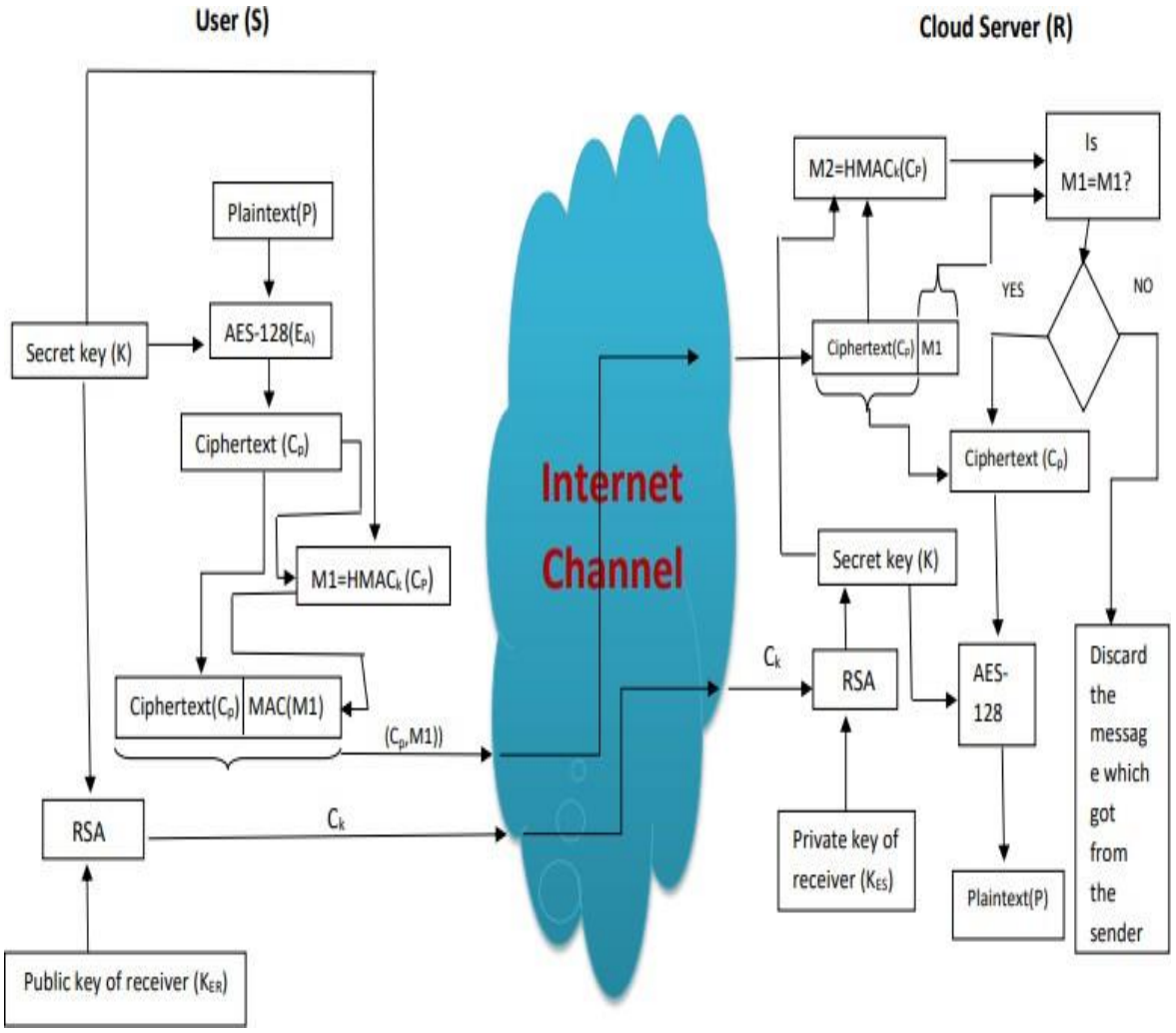

Figure 3.5 Flowchart of the proposed model.

CHAPTER 4

RESULTS AND DISCUSSION

This section provides the implementation of Rivest, Shamir, Adleman (RSA), Advanced Encryption Standard (AES), and Hybrid encryption technique algorithm according to proposed structure and simulation results is discussed in detail. All the algorithms has been programmed with Wolfram language in Mathematica platform. The experiment has been done using HP laptop which having windows 10 as operating system. The proposed model is implemented on the different size of data ranging from 64 Byte to 288 Byte. We calculate the encryption and decryption time, throughput for all the three algorithms and compared results.

4.1 RSA Encryption and Decryption Time

The Table 4.1 and Figure 4.1 below is showing the required time for the process of encryption and decryption for RSA Algorithm.

Table 4.1 RSA encryption and decryption time

Data Size (Byte)	Time required for encryption (s)	Time required for decryption (s)
64	1.074	1.08057
96	1.07541	1.07648
144	1.07015	1.08404
176	1.06584	1.08666
208	1.06945	1.09343
240	1.08034	1.09673
288	1.08249	1.08561

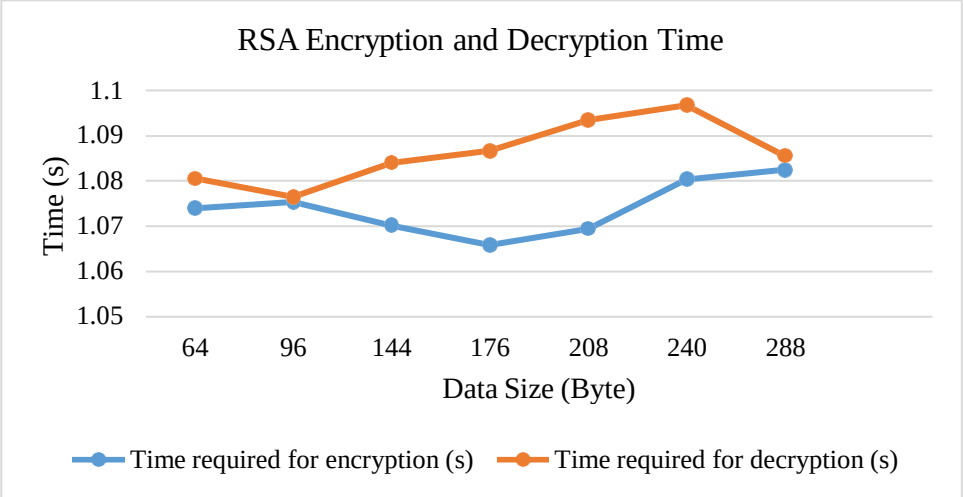

Figure 4.1 Encryption and decryption time consumption of RSA.

4.2 AES Encryption and Decryption Time

The required encryption and decryption time for AES Algorithm is shown in Table 4.2 and Figure 4.2.

Table 4.2 AES encryption and decryption time

Data Size (Byte)	Time required for encryption (s)	Time required for decryption (s)
64	0.00025241	0.00032401
96	0.00023034	0.00033865
144	0.00020102	0.00031522
176	0.00024265	0.00030787
208	0.00019955	0.00054922
240	0.00025144	0.00027889
288	0.00021165	0.00029941

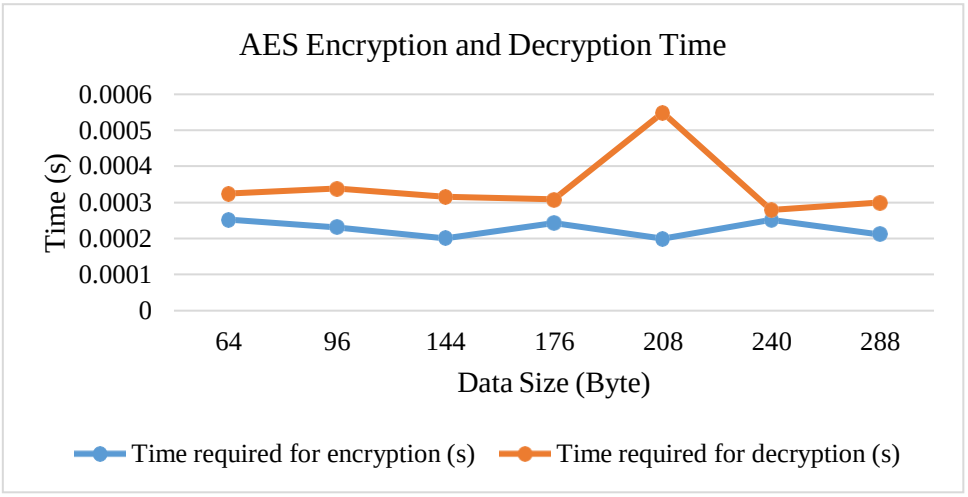

Figure 4.2 Encryption and decryption time consumption of AES.

4.3 Encryption and Decryption Time for Proposed Hybrid Algorithm

The required time for the process of encryption and decryption for proposed Hybrid Algorithm is shown Table 4.3 and Figure 4.3 below.

Table 4.3 Proposed Hybrid algorithm's encryption and decryption time

Data Size (Byte)	Time required for encryption (s)	Time required for decryption (s)
64	1.04707	0.0540557
96	1.04708	0.0540354
144	1.0471	0.0539762
176	1.04707	0.0539808
208	1.04709	0.0539886
240	1.04707	0.0539728
288	1.04706	0.0539677

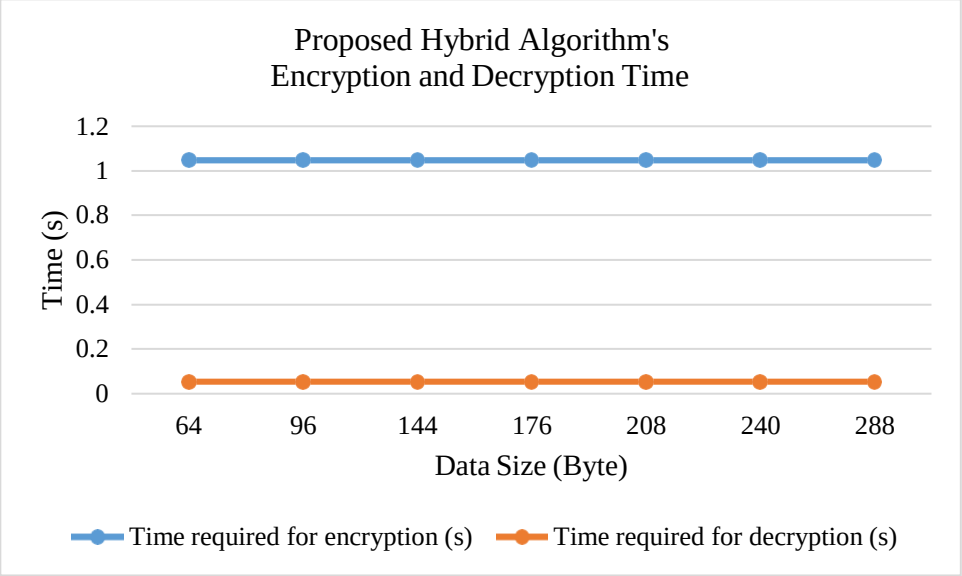

Figure 4.3 Encryption and decryption time consumption of proposed Hybrid algorithm.

It can be concluded that the proposed hybrid scheme allows the user to encrypt data with hybrid algorithms that uses two strong encryption algorithms without taking more time for both encryption and decryption for various message lengths.

4.4 Throughput

Throughput is described as the amount of data that passes via a network system. It is a result of dividing all data send in Byte by the average time required to send all data in second. Throughput value in (B/Sec) for each algorithm as shown in Table 4.4.

Table 4.4 Throughput value of the three algorithms

RSA Throughput	AES Throughput	Hybrid Throughput
562.92	2126.53	1104.38

4.5 Comparative Analysis of the Results

This part showing comparative analysis based on the metrics such as encryption time, decryption time, total time, throughput and data size to measure the performances of the selected algorithms.

Table 4.5 Time taken for each algorithm in second

Data size (Byte)	RSA encryption	RSA decryption	AES encryption	AES decryption	Hybrid encryption	Hybrid decryption
64	1.074	1.08057	0.00025241	0.00032401	1.04707	0.0540557
96	1.07541	1.07648	0.00023034	0.00033865	1.04708	0.0540354
144	1.07015	1.08404	0.00020102	0.00031522	1.0471	0.0539762
176	1.06584	1.08666	0.00024265	0.00030787	1.04707	0.0539808
208	1.06945	1.09343	0.00019955	0.00054922	1.04709	0.0539886
240	1.08034	1.09673	0.00025144	0.00027889	1.04707	0.0539728
288	1.08249	1.08561	0.00021165	0.00029941	1.04706	0.0539677

Table 4.6 Total time for each algorithm in second

Data Size (Byte)	RSA Total Time	AES Total Time	Hybrid Total Time
64	2.15457	0.00057642	1.1011257
96	2.15189	0.00056899	1.1011154
144	2.15419	0.00051624	1.1010762
176	2.1525	0.00055051	1.1010508
208	2.16288	0.00074877	1.1010786
240	2.17707	0.00053033	1.1010428
288	2.1681	0.00051106	1.1010277

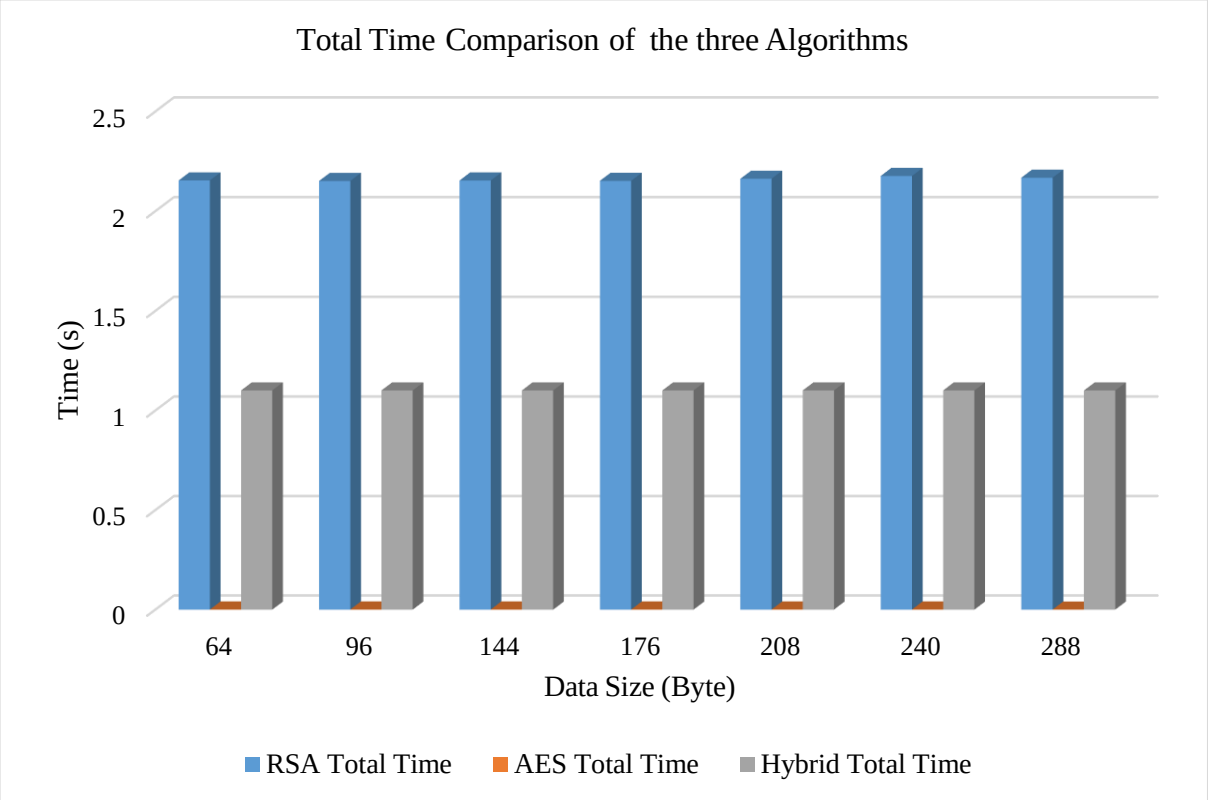

Figure 4.4 Comparative analysis of the total time of encryption and decryption of the three algorithms.

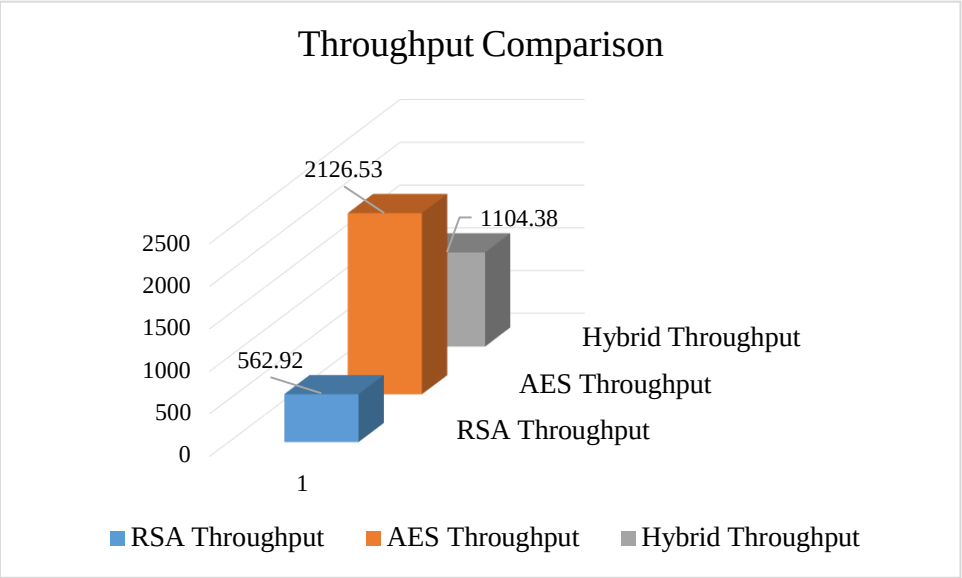

Figure 4.5 Throughput value of the three algorithms.

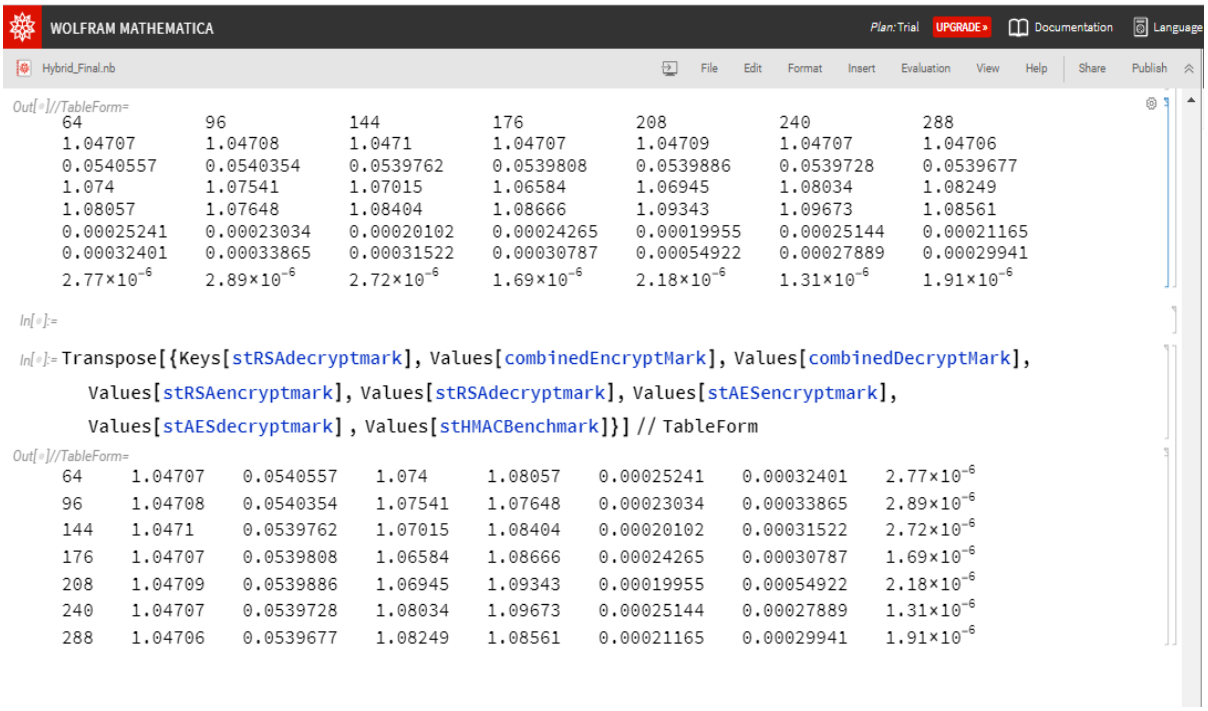

Figure 4.6 The encryption-decryption time of the three algorithms in Wolfram Mathematica.

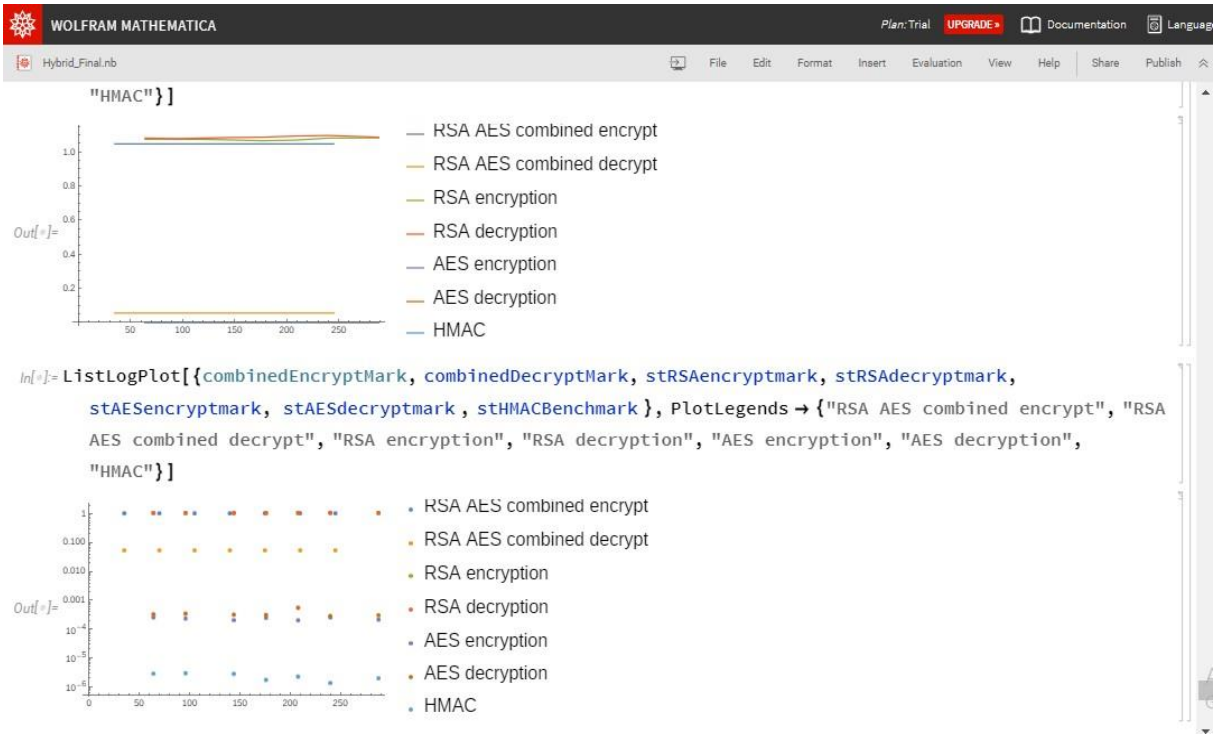

Figure 4.7 The encryption-decryption time value of the three algorithms in a tabular form in Wolfram Mathematica.

Three algorithms are used in this experiment to encrypt and decrypt the same data size. The encryption-decryption schedule of the three is shown in Table 4.5 with different data size. Table 4.6 shows the total processing time in second. According to the results shown in the Table 4.6, it is clear that the hybrid algorithm is better than the other algorithms (AES and RSA) after testing the three algorithms and calculating the time required to complete the processing on several data of different sizes. Results show that the best algorithm is the hybrid algorithm which provides the reliability and the highest security for the data sent when the RSA, AES and hybrid algorithms are compared, the encryption and decryption time differences evaluated in Figure 4.4. Throughput analysis is shown in the Figure 4.5. The Figure 4.6 and 4.7 presents a table and two figure of encryption and decryption time value of the three algorithms respectively in Wolfram Mathematica platform. The hybrid encryption algorithm provides the highest level of data security than other algorithms, and its speed is much faster than RSA and slightly slower than AES algorithms.

4.6 Discussion

According to the experiments that applied in this paper, the decryption time of the RSA algorithm and the AES algorithm increases with an increase in data size. The RSA algorithm

has increased dramatically, almost linear development, and the AES algorithm has increased marginally. The decryption time of the hybrid encryption algorithm is stable close to a certain value, close to the AES algorithm. Compared to the RSA algorithm, the improvement in decryption performance has a noticeable impact on large files. Through contrasting encryption and decryption above, the hybrid algorithm greatly increases encryption performance relative to the RSA algorithm while encrypting and decrypting broad data. Owing to the use of dual-layer encryption, the decryption complexity is intensified as the data is decrypted, and the hybrid algorithm optimises the dilemma that the AES algorithm key is leaked and the authentication is ineffective.

Experimental analysis result of attacks and functionalities of AES and RSA assist to determine gaps. Each category has the elimination process of attacks. This is section gives the proposal how hybrid encryption can eliminate gaps.

Text and cipher attacks: Use of AES-256 eliminate plaintext and cipher text attacks accessing.

Related-key attacks: Encryption AES key using RSA cryptography.

Side channel attacks: High speed of AES on data minimize attacks related to events.

The functionalities of algorithm determine the level of data security. Below table describe nature of algorithm, security level, and key management and time parameters for AES, RSA and Hybrid. The table gives the summary of functionalities enhance regarding values assign to hybrid.

Table 4.7 Summary of the functionalities of the algorithms AES, RSA and Hybrid

FACTORS	AES	RSA	HYBRID
NATURE			
Type of algorithm	Symmetric	Asymmetric	Hybrid
Data size	Big	Small	Big
Block size	128,192, 256 bits	Minimum 512 bits	128 bits
Power consumption	Low	High	Low
Rounds	10,12,14	1	AES-14, RSA-1
Memory space	Less	More	More
TIME			
Encryption/Decryption	Fast	Slow	Fast
Speed of computation	Fast	Slow	Fast
Software Implementation	Fast	Slow	Fast
Hardware Implementation	Fast	Slow	Fast

SECURITY			
Confidentiality	Moderate	High	High
Integrity	None	High	High
Authentication	None	Moderate	High
Non- repudiation	None	None	High
Digital signatures	None	Possible	Available
Hash function	None	Possible	Available
KEY			
Key generation	Yes	Yes	Yes
Key storage	No	Yes	Yes
Multiple keys	No	Yes	Yes
Secret key	Yes	Yes	Yes
Public key	No	Yes	Yes
Shared key	Yes	No	No
Distribution public key	No	Yes	Yes
Distribution of secret key	Yes	No	No
Pair keys	No	Yes	Yes
Inverse keys	Yes	No	Yes

This hybrid technique combines the features of both asymmetric cryptography and symmetric cryptography. Proposed schema based on two levels of data encryption and key encryption. The technique provide a higher level of security and efficiency level. It is better than either of the technique used separately. The security of data in this hybrid is achieved by AES-128 algorithm to confirm the confidentiality. Data integrity is the process which is difficult to meet with only symmetric key. This security goal was guaranteed by the combination of RSA to encrypt AES secret key. Authentication of sender and non-repudiation of data is ensured using HMAC function. Furthermore, they must be archived under time-based management so that they would be available for retrieving. The combination of different cryptography algorithms provide a maximized efficiency, correcting or compensating. This approach offers a solution for various weaknesses, that must be faced in a security crypto system including. The results showed that the hybrid encryption is 67.47% faster than the RSA algorithm and 32.39% slower than AES algorithm. That's why the algorithm of the hybrid encryption can be used in software application, system design, and other fields that required to exchange data security, which can effectively protect the data, in addition to the performance and fast execution time.

CHAPTER 5

CONCLUSION

Cloud computing is the latest trend in IT. But security is the biggest challenge in this area. So, researchers mainly concentrate in this area. Encryption is the best security method, now different kinds of encryption techniques apply in cloud computing environment, some extend hacking can be prevented in this way. In this paper, we have suggested a mechanism of hybrid encryption and decryption based on AES-128 bits algorithm to encrypt the original plaintext and then encrypt the secret key of AES by RSA algorithm. Furthermore, we used the value which produced by HMAC algorithm and attached in the end of encrypted plaintext to check the integrity and authenticity of message. The results of experiment shows the time required for encryption and decryption different size of data started from 64 Byte up to 288 Byte. Moreover, the throughput value has been measured for the three algorithms. According to the experiments that applied in this paper, the algorithm of the hybrid encryption can be used in software application, system design, and other fields that required to exchange data security, which can effectively protect the data, in addition to the performance and fast execution time.

REFERENCES

[1] M. Al Morsy, J. Grundy, and I. Müller, "*An analysis of the Cloud Computing Security Problem*," ResearchGate, no. December, pp. 7[1] M. Al Morsy, J. Grundy, and I. Müller," 2010.

[2] Oleiwi, Z. C., Alawsi, W. A., Alisawi, W. C., Alfoudi, A. S. and Alfarhani, L. H., (2020). "*Overview and Performance Analysis of Encryption Algorithms*," J. Phys. Conf. Ser., vol. 1664, no. 1, doi: 10.1088/1742-6596/1664/1/012051.

[3] Chinnasamy, P., Padmavathi, S., Swathy, R. and Rakesh, S., (2021). "*Efficient data security using hybrid cryptography on cloud computing*," Lect. Notes Networks Syst., vol. 145, no. September, pp. 537– 547, doi: 10.1007/978-981-15-7345-3_46.

[4] Banasode, P. and Padmannavar, S., (2018). "*Protecting and Securing Sensitive Data in a Big Data Using Encryption*". EAI Endorsed Trans. Smart Cities, vol. 0, no. 0, p. 163991.

[5] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou. Achieving secure, scalable and fine grained data access control in cloud computing, in: IN-FOCOM, 2010 Proceedings IEEE, 2010.p.1-9.

[6] R Velumadhava Rao, K Selvamani. Data Security Challenges and Its Solutions in Cloud Computing, https://www.researchgate.net/publication/277935944.

[7] Peter Mell, and Tim Grance, "The NIST Definition of Cloud Computing," 2009, http://www.wheresmyserver.co.nz/storage/media/faq-files/cloud-def-v15.pdf, Accessed April 2010.

[8] Garg, S. K. and Buyya, R. (2011): *Green Cloud computing and Environmental Sustainability*. Cloud computing and Distributed Systems (CLOUDS) Laboratory Department of Computer Science and Software Engineering. University of Melbourne. Australia.

[9] Amit Sangroya, Saurabh Kumar, Jaideep Dhok, and Vasudeva Varma, "*Towards Analyzing Data Security Risks in Cloud Computing Environments*", Springer-Verlag Berlin Heidelberg 2010, pp. 255- 265.

[10] https://api.semanticscholar.org/CorpusID:33236322.

[11] Cloud computing security, http://en.wikipedia.org/wiki/Cloud_computing_security.

[12] D. Chen and H. Zhao, "*Data security and privacy protection issues in cloud computing*," Proc. - 2012 Int. Conf. Comput. Sci. Electron. Eng. ICCSEE 2012, vol. 1, no. 973, pp. 647-651, 2012.

[13] Majda Omer Elbasheer and Dr.Taring Mohammed, "*Signing and verifying certificates by NTRU and RSA algorithm*" International Conference on Cloud Computing (ICCC), pp. 1-4. IEEE.2015. 15 | P a g e

[14] Vijay Kumar Pant, Jyoti Prakashand Amit Asthana, "*Three step data security model for cloud computing based on RSA and Stegnography techniques*" International Conference on Green Computing and Internet of Things (ICGCIoT), pp. 490-494. IEEE. 2015.

[15] Sakinah Ali Pitchay, Wail Abdo Ali Alhiangem, Farida Ridzuan and MadihahMohd Saudi, "*A proposed systemconcept on enhancing the encryption and decryption method for*

cloud computing"17th UKSim-AMSS International Conference on Modelling and Simulation (UKSim), pp. 201-205. IEEE. 2015.

[16] PreetiGarg and Dr.Vineet Sharma, "*An efficient and secure data storage in mobile cloud computing through RSA and hash function*" International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 334-339. IEEE. 2014.

[17] Vishwanath S Mahalle and Aniket K Shahade, "*Enhancing the data security in cloud by implementing hybrid(RSA & AES) encryption algorithm*", International Conference on Power, Automation and Communication (INPAC), pp. 146-149. IEEE. 2014.

[18] Mr. Prashant Rewagad and Ms. Yogita Pawar, "*Use of digital signature with Diffie Hellman key exchange and AES encryption algorithm to enhance security in cloud computing*" International Conference on Communication Systems and Network Technologies (CSNT), pp. 437-439. IEEE.2013.

[19] Yi, Xun, Russell Paulet, and Elisa Bertino, "*Homomorphic encryption and applications*" Cham: Springer, vol 3.pp 27-49.2014.

[20] Kota, C.M. and Aissi, "*Implementation of the RSA algorithm and its cryptanalysis*," In proceedings of the ASEE Gulf-Southwest Annual Conference, pp. 20-22. March 2002.

[21] Cloud Computing Use Case Discussion Group, "Cloud Computing Use Cases Version 3.0," 2010.

[22] ENISA, "*Cloud computing: benefits, risks and recommendations for information security*," 2009, http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment, Accessed On July 2010.

[23] Cloud Security Alliance (CSA). (2010). Available: http://www.cloudsecurityalliance.org.

[24] B. K. Dewangan, A. Agarwal, & A. Pasricha, "*Credential and security issues of cloud service models*", 2nd International Conference on Next Generation Computing Technologies (NGCT), (pp. 888-892), IEEE 2016.

[25] V. S. Mahalle and A. K. Shahade, "*Enhancing the data security in Cloud by implementing hybrid (Rsa & Aes) encryption algorithm*," 2014 Int. Conf. Power, Autom. Commun. INPAC 2014, no. I, pp. 146–149, 2014, doi: 10.1109/INPAC.2014.6981152.

[26] R. Kiruthika, S. Keerthana, and R. Jeena, "*Enhancing Cloud Computing Security using AES Algorithm*," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 5, no. 3, pp. 630–635, Mar. 2015.

[27] V. K. Soman and V. Natarajan, "*An enhanced hybrid data security algorithm for cloud*," Int. 125 Conf. Networks Adv. Comput. Technol. NetACT 2017, pp. 416–419, Jul. 2017, doi: 10.1109/NETACT.2017.8076807.

[28] B. Bindu, L. Kamboj, and P. Luthra, "*Secure File Storage in Cloud Computing Using Hybrid Cryptography Algorithm*," Int. J. Adv. Res. Comput. Sci., vol. 9, no. 2, pp. 773–776, 2018, doi: 10.26483/ijarcs.v9i2.5916.

[29] M. Batra, P. Dixit, L. Rawat, and R. Khalkar, "*Secure File Storage In Cloud Computing Using Hybrid Encryption Algorithm*," Int. J. Comput. Eng. Appl., vol. XII, no. VI, pp. 30–36, Jun. 2018.

[30] L. Kumar and N. Badal, "*A Review on Hybrid Encryption in Cloud Computing*," Proc. - 2019 4th Int. Conf. Internet Things Smart Innov. Usages, IoT-SIU 2019, pp. 1–6, 2019, doi: 10.1109/IoT-SIU.2019.8777503.

[31] F. Zhang, Y. Chen, W. Meng, and Q. Wu, "*Hybrid Encryption Algorithms for Medical Data Storage Security in Cloud Database*," Int. J. Database Manag. Syst., vol. 11, no. 01, pp. 57–73, 2019, doi: 10.5121/ijdms.2019.11104.

[32] G. Viswanath and P. V. Krishna, "*Hybrid encryption framework for securing big data storage in multi-cloud environment*," Evol. Intell., no. 0123456789, 2020, doi: 10.1007/s12065-020-00404- w.

[33] Ogundoyin, I.K., Ogunbiyi, D.T., Adebanji, S. and Okeyode, Y.O., "*Comparative Analysis and Performance Evaluation of Cryptographic Algorithms,*" UNIOSUN Journal of Engineering and Environmental Sciences. Vol. 4 No. 1. March. 2022, DOI: 10.36108/ujees/2202.40.0140.

[34] Rege, Komal, et al. "*Bluetooth Communication using Hybrid Encryption Algorithm based on AES and RSA,*" International Journal of Computer Applications 71.22 (2013).

[35] Chandra, Sourabh, et al. "*A comparative survey of symmetric and asymmetric key cryptography,*" Electronics, Communication and Computational Engineering (ICECCE), 2014 International Conference on. IEEE, 2014.

[36] Crocker, Paul, and Pedro Querido. "*Two Factor Encryption in Cloud Storage Providers Using Hardware Tokens,*" 2015 IEEE Globecom Workshops (GC Wkshps). IEEE, 2015.

[37] Leng, F., Xu, J., Pei, S.: "*Network information security method based on RSA fusion AES algorithm,"* J. Huaqiao Univ. (Nat. Sci.)38(01), 117–120 (2017).

[38] Wan, Z.: "*Public key encryption scheme based on QI hyperchaos and its FPGA implementation,"* Tianjin Polytechnic University (2018).

[39] C. Xue-zhou, "*Network Data Encryption Strategy for Cloud Computing*," 2015.

[40] Rajput, Somesh Kumar, and Anuradha Konidena. "*Performance Enhancement In Image Encryption Using Aes*." (*2015).*

[41] Zhu, Xiaoyan, et al. *"Efficient privacy-preserving authentication for vehicular ad hoc networks.*" Vehicular Technology, IEEE Transactions on 63.2 (2014): 907- 919.