DOI: 10.24996/ijs.2022.63.7.40

ISSN: 0067-2904

Elliptic Curve Cryptography Performance Evaluation for Securing Multi-Factor Systems in a Cloud Computing Environment

G.O. Ogunleye [*], S.E. Akinsanya

Department of Computer Science, Federal University, Oye-Ekiti, Ekiti State, Nigeria

Received: 14/10/2021 Accepted: 9/3/2022 Published: 30/7/2022

Abstract:

In the contemporary world, the security of data and privacy policies are major concerns in cloud computing. Data stored on the cloud has been claimed to be unsafe and liable to be hacked. Users have found it difficult to trust their data in the cloud. Users want to know that their data is accessible from anywhere and that an unauthorized user will not be able to access it. Another area of concern is the authentication of users over the cloud. There are a number of security concerns with Cloud Computing which include Distributed Denial of Service, Data leakage, and many more, just to mention a few. In this paper, an Elliptic Curve Cryptography (ECC) algorithm is used for the encryption and decryption of the information stored on the cloud, so that if someone gains access to the server, it would be unable to access the original information. Performance evaluation of the developed system is performed by comparing ECC with the RSA algorithm. The results show that the ECC method is more efficient than RSA when used to secure information in the cloud.

Keywords: ECC, Multi-Factor, Cloud Computing.

1. Introduction

Cloud Computing is a method of storing data in the cloud and allowing users to access various users' applications over the web. There are a number of Cloud Computing providers that provide a number of services to users, such as software-as-a-services, infrastructure-as-a-services, e-mail, storage, etc. Examples of Cloud Computing providers include Yahoo, Google, SalesForce, Amazon, etc. Cloud Computing is not limited to big companies, small companies, or medium-sized enterprises. Entrepreneurs would also enjoy the provided services by cloud computing, and this would save them a lot of expenses because they would have the chance of making use of only the services they need from the Cloud Computing service provider, such as the storage space, capacity of communication, computing power, etc. Recently, Cloud Computing has fundamentally developed into a significant zone of exploration [1]. There are numerous advantages to cloud computing. At the same time, it has numerous difficulties, such as data confidentiality, data security, integrity, data privacy, and various other issues that need attention [2]. The proposed work centers around securing the information that is transferred by utilizing the Elliptic Curve Cryptography (ECC) approach, which guarantees multi-factor authenticity in addition to keeping up the confidentiality of data, authenticity, and integrity. This system is proposed to ensure the information is stored in a safe and secure manner to avoid attacks and intrusions.

*Email: gabriel.ogunleye@fuoye.edu.ng

Over the years, customers have always wanted to store their information in a safe and secure environment. This led to the emergence of cloud computing, which is the combination of servers that enables access to resources and runs applications directly through the cloud without installing them on the local machine. Cloud Computing allows users to build, configure, and customize online applications simultaneously. With cloud computing, we can make use of software and hardware resources remotely. Through the rapid advancement of the internet, online work has turned into a basic factor of open movement [3]. Due to the accessibility of the system, individuals need to confront the dangers when they enjoy the help of the system. The Cloud Computing is a ground-breaking and powerful strategy for the customers to acknowledge assets in various accessible conditions like tele-medicine, clinical data framework, distributed computing, E-wellbeing, remote body territory system, and remote sensing [4]. Laptops, PCs, smartphones, and other electronic devices can be used to access multiple services like programs, storage, and platforms to develop applications that are provided by cloud providers over the internet [5]. Cloud Computing has helped to reduce business management, IT costs, and the cost of software and hardware maintenance. Cloud Computing also aids businesses in gaining access to expert IT solutions. When it comes to cloud computing, users do not have to worry about data loss, virus attacks, and other related problems when they are using the cloud in a normal way [7]. The security of data in Cloud Computing must be given prior attention, as security concerns such as confidentiality and privacy of customer's data must be well addressed. Elliptic Curve Cryptography (ECC) is an asymmetric cryptographic algorithm which is based on the concept of elliptic curves. Due to its smaller key size, ECC has an outstanding advantage over other cryptographic algorithms. Security in Cloud Computing is a bone of contention in our contemporary world. One of the ways users can secure their data is to encrypt it before they store it on the cloud [7]. This research work is intended towards providing security services such as the user’s authenticity and ensuring data confidentiality by using the ECC algorithm due to its advantages in terms of smaller key size, less memory usage, and lower CPU time [7]. Cloud Computing is a world view that provides a vast number of processing assets to end clients, depending on their requests. Clients can get different kinds of administrative services from the cloud such as asset pushing, versatile and adaptable utility administration throughput, execution, high accessibility, oversaw administrations, and the likes because of incorporated administration of the cloud framework [8]. Previous research studies have shown that Data leakage and Distributed Denial of Service are the two most serious security concerns with Cloud Computing [9]. Implementing various symmetric key methods can increase data security by ensuring that data on the server is stored in such a way that even if an intruder gains access, the person is unable to open the original data. It will be necessary to decode the original data [6].

As technology advances, there is a high demand for cloud services, and users have been concerned about the security and safety of their data on the cloud. Users claim that it is not safe to store data remotely as data is likely to be hacked. As a result, users feel concerned, and it is hard for them to trust their data in the cloud. Cloud users want to be sure no other person can access their data. The authentication of users over the cloud is guaranteed. Therefore, there is a need to develop a multi-factor system for a Cloud Computing environment to ensure the data is secured and a privacy policy is established over the cloud. The ECC algorithm is increasingly recommended as one of the public key cryptosystems for cloud environments. ECC is a relatively new public key cryptosystem that has been investigated for data security. Asymmetric cryptography is now used by the majority of E-commerce applications to maintain security. Unlike other public key cryptosystems such as Diffie-Hellman, RSA, and others, ECC delivers security with smaller key sizes, resulting in lower power consumption, faster calculation, and faster data transfer speeds, among other benefits.

This paper is aimed at developing an Elliptic Curve Cryptography for Multi-Factor system in Cloud Computing Environment. The implementation was done using an Elliptic Curve Cryptography for Multi-Factor System in a Cloud Computing Environment, while the system was evaluated by comparing it with other similar works in the literature.

2 Literature Review

2.1 SECURITY OF INFORMATION IN CLOUD COMPUTING

Security in Cloud Computing is a bone of contention in our modern world. One of the ways users can secure their data is to encrypt it before they store it on the cloud [10]. Cloud Computing is an outstanding technology that is virtualized with resources to offer a dynamic and scalable service via the internet. Various devices such as laptops, PCs, smartphones, and other mobile devices can be used to access cloud-based services such as storage, programs, and application development platforms [11].

To deliver information, most frameworks utilize a mix of strategies that incorporates:

1. Encryption is the process of encoding data using a combination of expressions. An encryption key is required if the user wants to decrypt the encrypted data files. While a user can obtain ciphertext, unapproved clients are prevented from obtaining the encryption key, ensuring the confidentiality of private data.
2. An authentication system manages the formation of a client name and a security secret word.
3. Authorization: the cloud service providers (CSP) enroll the clients who are approved to get access to the data that exists on the cloud server. To make sure client data is accessed each time it is accessed on various applications, differing cryptographic calculations and confirmation methods are utilized. Because of the limited processor speed and run-time memory, the devices necessitate an algorithm that can also be employed in small PC devices. Security of information and information experiencing significant change may be a worry while putting away delicate information on the distributed storage.

2.2 CLOUD SECURITY ISSUES

There are three fundamental issues with distributed computing security; they are: confidentiality, integrity, and availability, as demonstrated below:

1. Confidentiality: confidentiality ensures that the validated individual will have the option to get to and recover the information, and an unauthorised user cannot access the information. In order to secure the privacy of data, the data is encoded by the approved individual and later has the option to be decoded as the key is known only to the same individual. Snooping and traffic analysis are the two most serious risks to confidentiality. There are several methods for ensuring data confidentiality, including using record consents and access control records to limit access to sensitive information.
2. Integrity: integrity is a data security problem that demonstrates the protection of data from unauthorized changes. This is a method of assuring data integrity that involves hashing the data and comparing it to the hash of the underlying, unique message. The initial information hash must be provided in a safe and secure manner.
3. Availability: is the assurance that data is available for the approved user at whatever point is required [3]. There is a need to stress over secrecy and trustworthiness if the affirmed clients cannot get the data they are qualified for. It is one of the most significant highlights of data security [4].

2.3 BACKGROUND OF ELLIPTIC CURVE CRYPTOGRAPHY

Elliptic Curve Cryptography (ECC) is a type of public key encryption that uses elliptic curve theory over finite fields [11]. Cryptographic keys are made faster, smaller, and more efficient using ECC. 150 years ago, mathematicians investigated the properties and functions of an elliptic curve. A number of scientists proposed their usage in cryptography for the first time in 1985. Since the beginning of 1990, ECC has gained acceptance from a growing number of accredited organizations and security protocols [12]. They proposed elliptic curve cryptography in the 1980s. An elliptic curve is the arrangement of a non-particular cubic polynomial mathematical statement with two questions across a constrained field in discrete logarithmic cryptosystems.ECC is one of the most powerful asymmetric algorithms for a given key length, making it particularly appealing for security applications requiring limited integrated circuit area and computational capacity, such as PCs, smart cards, cards, and wireless devices. The security of the ECC algorithm is based on the difficulty of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). Many algorithms have been developed to solve the problem. Nonetheless, the effectiveness of any algorithm is determined by the type of curve and the characteristics of k, where k is a random number [13]. The cloud provider's primary liability is data security and data storage. Therefore, a technique that provides secure data encryption and decryption to give protection against theft and attackers is required for competent data security and consistency [10]. As a result, this technology is used to increase data security in Cloud Computing by implementing Elliptical Curve Cryptography, which adds security to data transmission, storage, authorization, and authentication operations.

2.4 RELATED WORKS

Arezou et al. [3] presented an efficient use of Elliptic Curve Cryptography in the construction of a three-factor authentication system for satellite communications when it comes to the application of ECC to Cloud Computing security. Huang et al. [8] proposed an efficient ECC-Based Authentication Scheme. This outstanding research on public-key cryptosystems based on elliptic curves revealed that ECC is suitable for the majority of existing applications. The elliptic curves are useful in applications where computer power is limited (intelligent cards, wireless devices, PC boards), memory size on an integrated circuit is limited, processing speed is critical, and digital signatures and their verification are frequently utilized.

Kumar & Grover's [11] study focused on applying the Elliptic Curve Cryptography algorithm, encryption, and decryption processes to improve Cloud Computing security and privacy protection. Kumari et al. [13], presented cloud and data security by providing protection to keys that are employed in encryption and decryption, which involves the use of Elliptic Curve Cryptography Encryption.

Anand and Perumal [2] presented a method for preventing any user from gaining unwanted access to confidential data stored in the cloud. Elliptical Curve Cryptography (ECC) is well known for being superior to public-key encryption approaches in wireless devices. It assists in reducing the device's processing time.

The existing system provides security for user data using the Elliptic Curve Cryptography technique. However, it has some limitations in terms of key distribution and the number of keys shared under different attacks [14]. The private key is currently stored by partitioning the private key into three parts and storing them in three different storage locations. As a result, there is a significant risk that hackers will launch multiple attacks to gain access to authenticated user data that is stored across multiple cloud locations [15].

Various books deal with cloud computing's broad security challenges, but just a handful deal with the user's privacy in the cloud. Sasidevi et al. [16, 17] presented a novel security approach for safeguarding privacy in cloud services. The main concerns that arise in the cloud when accessing data, and the security-related issues and countermeasures to address the problem, were addressed by [18].

Kumar & Grover [11] proposed an efficient technique for password authentication that preserves privacy. This study presents a system to prove the authenticated user's identity without the need to admit their passwords. The idea of using a data owner has been implemented in this project. Here, in this project, privacy has been the main focus, rather than the security of data. Wang et al. [19] presented a work based on the concept of ECC and provided a new method to secure the output of ECC. Shen et al. [17] facilitated the usage of ECC in Java by analyzing the capabilities and dealing with key generation, key exchange, and digital signatures.

Arumugam et al. [4] proposed a cloud computing to secure data sharing for mobile using RSA. They employed the idea of RSA calculation and Hash in conjunction with various encrypted devices to deliver information on a mobile cloud. Dhamodaran et al. [5] concluded that RSA on large blocks is computationally demanding and byte-parallel in nature. However, in this proposed study, the performance study with the use of ECC demonstrates an improvement in terms of execution time and security when compared to other studies in the literature.

3. RESEARCH METHODOLOGY

The primary goal of this paper is to develop an Elliptic Curve Cryptography for Multi-Factor system in Cloud Computing Environment. ECC is adopted for the encryption and decryption of data over the cloud because the size of the key used in ECC is small and this yields low computational power, resulting in low energy usage. The Java Servlets and JSPs, as well as the Apache Tomcat server and MySQL for storage, were used for the implementation of the model. HTML and CSS 3 were utilized to create the user interface.

3.1 PROPOSED ALGORITHM

Due to its small key size, the EEC technique is employed in this project for the encryption and decryption of file/message being uploaded to the cloud. Encryption is the process of converting a message or data into a ciphertext format. The ECC algorithm is utilized to encrypt a file or message. The data owner uploads an encrypted data file to the cloud along with the list of approved individuals. During the decryption process, the authorized user can download the encrypted file and use their shared private key to convert it back to the original message.

Encryption

The ECC algorithm is used to encrypt the message/file. The ECC is a public key encryption method based on the elliptic curve theory that allows for significantly faster and more efficient encryption and decoding. The encryption algorithm generates two ciphertexts using the prime numbers j and p, the random integer k [1 to q - 1] that belongs to Z*q, Message M, a base point on the elliptic curve P, the shared private key d, and the public key Q as input. The message as an integer cannot be utilized directly, so we apply the following formula to transform an integer to a coordinate:

x = M * k + j mod p, where →[j 0 to p – 1]

Using the elliptic curve formula, get the y coordinate for this x coordinate:

$y^2 = x^3 + Ax + B \bmod p$, Where A and B are constants.

Input: j;q;Z*q;p; k; M; P; d; Q

j= prime number

q= integer number

Z*q= domain of q

Q= public key

d = shared private key

p = prime number

P = base point on the elliptic curve

k = random number Z^*_q (1 to q - 1)

M = message

Output: C_1 and C_2 is the output

C1=cipher text 1

C2= cipher text 2

Step one: Determine the public key. Q = d * P mod p

Step two: Calculate C1 = k * P mod p using a random number k.
Step three: Calculate C2 by using C2=M + k * Q
Step four: Ciphertexts C_1 and C_2 are uploaded to the cloud

Decryption

The owners/users download the encrypted file and use their shared private key to decrypt it. The following formula must be used to recover the original message: C2 - d * C1 = M
The original message, M, has been decrypted with the owner's private key.
C1, C2, d as input
M is the output (original message)
Proof:
Message M is denoted by, M = C2 – d * C1
C2 – d * C1 = (M + k * Q) – d * (k * P mod p)
As, (C1 = k * P; C2=M + k * Q)
C2 – d * C1 = M + k * d * P mod p – d * k * P mod p
C2 – d * C1 = M (cancelling k * d * P mod p)
The original message is obtained.

3.2 System Flowchart

A flowchart is a pictorial representation of the separate steps of activities in sequential order. It involves the use of different shapes to represent what the system is doing. Figure 1. depicts the flowchart of the ECC encryption and decryption process utilized in this study. The following shapes are used: the rectangle shape denotes the process; the parallelogram denotes the input; the oval shape denotes the start and endpoints.

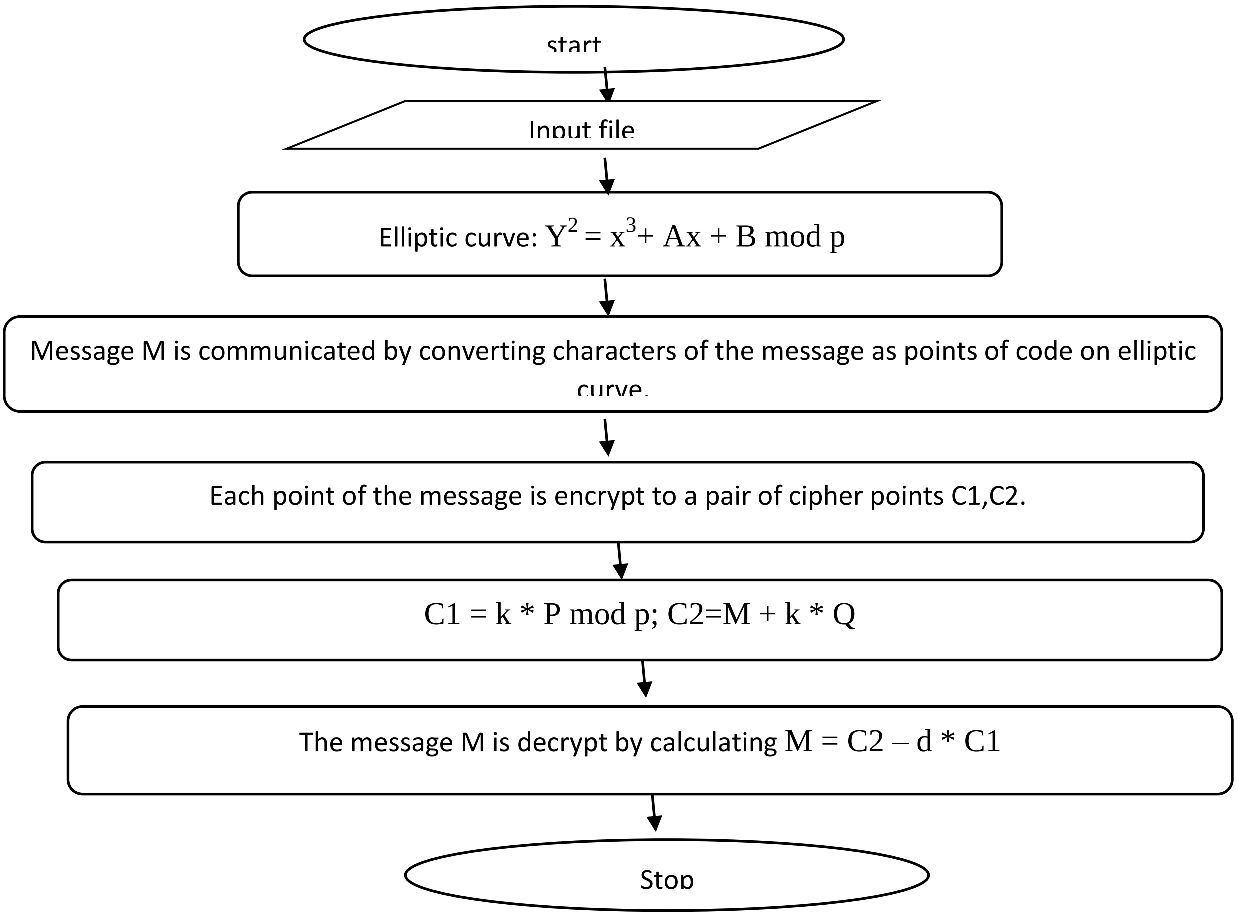

start
Input file
Elliptic curve: $Y^2 = x^3 + Ax + B \bmod p$
Message M is communicated by converting characters of the message as points of code on elliptic curve.
Each point of the message is encrypt to a pair of cipher points C1,C2.
C1 = k * P mod p; C2=M + k * Q
The message M is decrypt by calculating M = C2 – d * C1
Stop

Figure 1-Flowchart of ECC encryption and decryption.

A, B = constants
x, y = variables
p = prime number
P =base point on the elliptic curve
k = random number Z^{*}_{q} (1 to q - 1)
Q= public key
$C_{1,}$ C_2 = ciphertext
M= original message

Let us assume that a sender S who chooses a random number k needs to transmit something special M. The message is transmitted by transforming the characters of the message as code to the elliptic curve's ($y^2= x^3+ Ax + B$). Following the encoding of all the characters in the message, the gatherings use the code table to turn the pair of points in each message into text characters. The public key Q is then generated. Each message point is encoded as a pair of cipher text C1 and C2. The sender maintains his shared private key d by converting the cipher text into elliptic curve points, then process C1= k * P, C2= M + k * Q where k is a random number for encrypting distinct message points and recognizes the points of each character, then calculates M = C2 – d * C1 to decode the message.

3.3 SYSTEM ARCHITECTURE

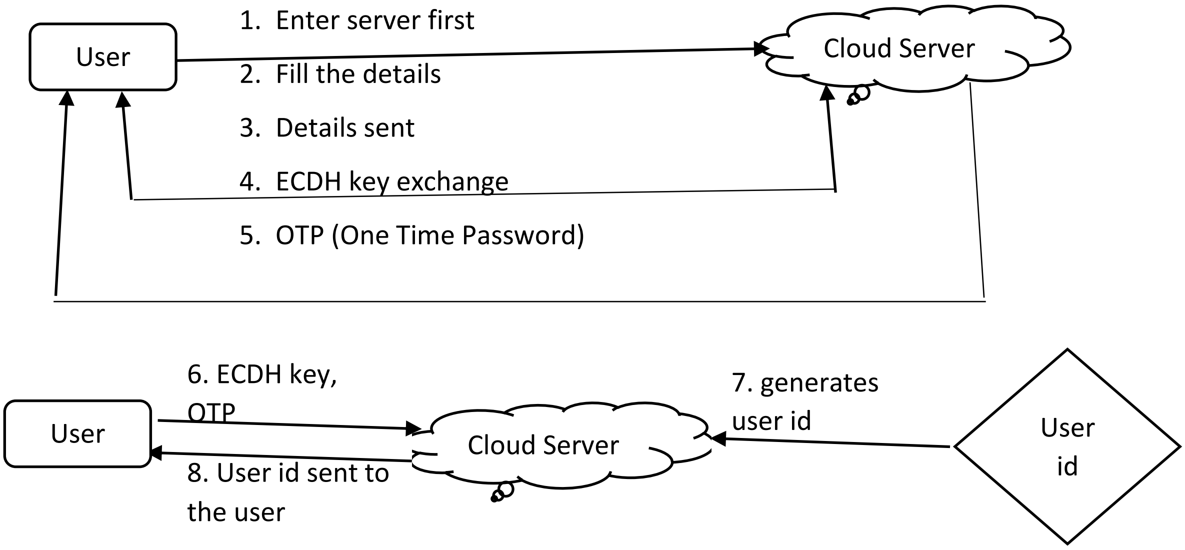


Figure 2-Component descriptions and dependency information of the Overall architecture.

In Figure 2, a new user that enters the server will have to fill in his details and send them to the server. Here, the key agreement of the Elliptic Curve Diffie-Hellman (ECDH) will be automatically generated, and an OTP will be sent to the client’s email address, which is given to the client at the point of registration. On successful validation, a unique user ID will be produced, and the user will be asked to save the ID as soon as the registration is complete. The key agreement has been successfully generated, and an OTP (one-time password) has been issued to the user's email address provided during the registration. Once the user has completed the registration process, he will utilize the secret key and unique ID to gain access to all functions and to validate the application's other features. After completing the registration process, the user can log in to access his profile and take advantage of the application's capabilities.

3.4 Data Flow Diagram (DFD) DIAGRAMS

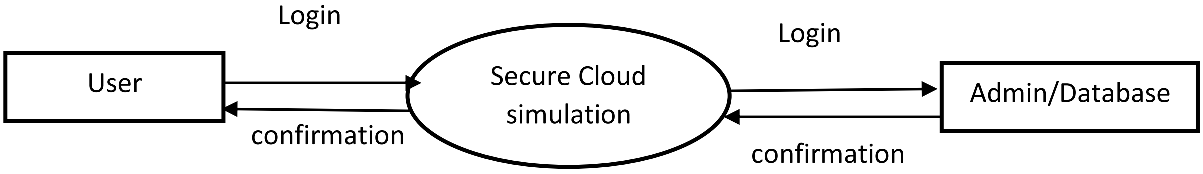


Figure 3-Level-0 Data Flow Diag

Figure 3. indicates the application **Login page,** this page consists of a form where the user will input their user ID and click on request for OTP which will be sent to the user email address provided at the point of registration, once the user is provided by the OTP and click on the submit button, the user ID and the OTP will be compared to the details in the database and once it is validated the user will be directed to my account page.

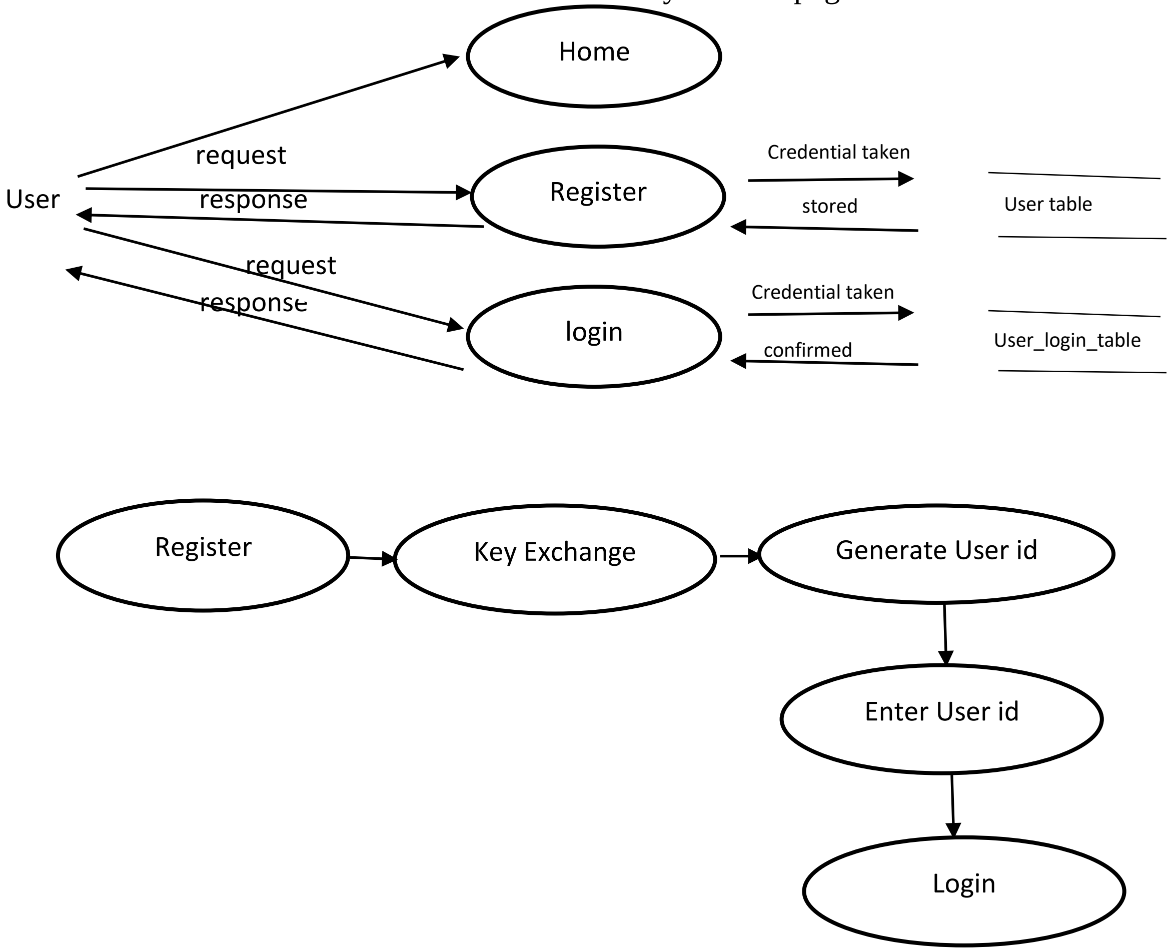


Figure 4-Level-1 Data Flow Diagram (DFD)

In Figure 4, the **home page** is the initial page that appears when the application is launched in the user's system browser. This page informs the user about the program and gives them the option of registering for the first time or logging in. It includes the app's logo and tagline, as well as information about the app. When a user is using the application for the first time or does not have a user account, the Application Registration Page appears. The user's complete name, email address, mobile number, date of birth, and gender are all required fields on this form, which are used to store the user's information and confirm the user before granting access to the app’s data. It uses regular expressions and pattern matching to check for all conceivable flaws in the credentials on the client side. If the data passes all of the test cases, it is then sent to the server to be validated and stored in the database, resulting in a working user profile.

The EEC algorithm is used to accomplish the key exchange, and here the user ID is generated. To improve key exchange, the user will provide the registration number and secret key; once the information is checked, the application will generate a unique user ID. The user must save the unique ID for future log-ins to the application, and the user will have access to all of the application's capabilities. When the user clicks the login tab button, the application **Login Page** will be loaded. The page consists of a form where the user will enter their user ID and request an OTP. Once the user enters the OTP and clicks the submit button, the user’s ID and OTP will be compared to the database's details, and the user will be directed to the **My Account Page**.

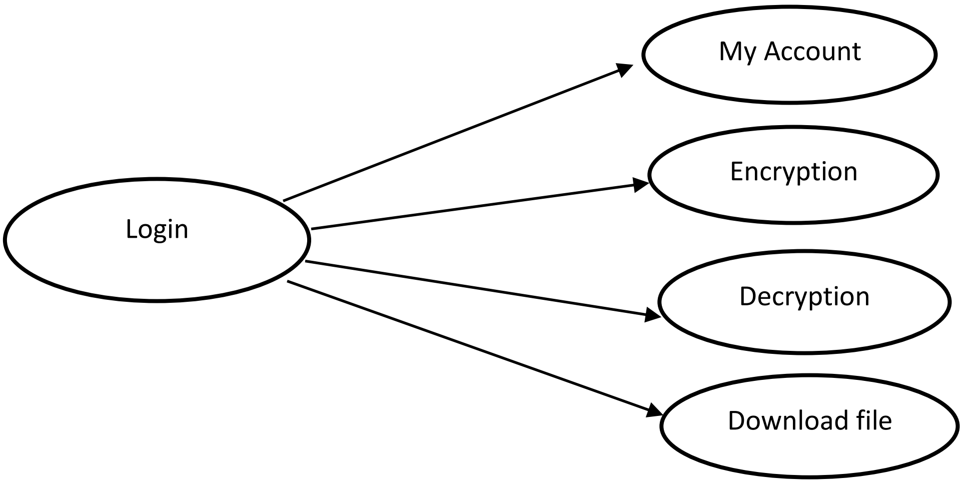
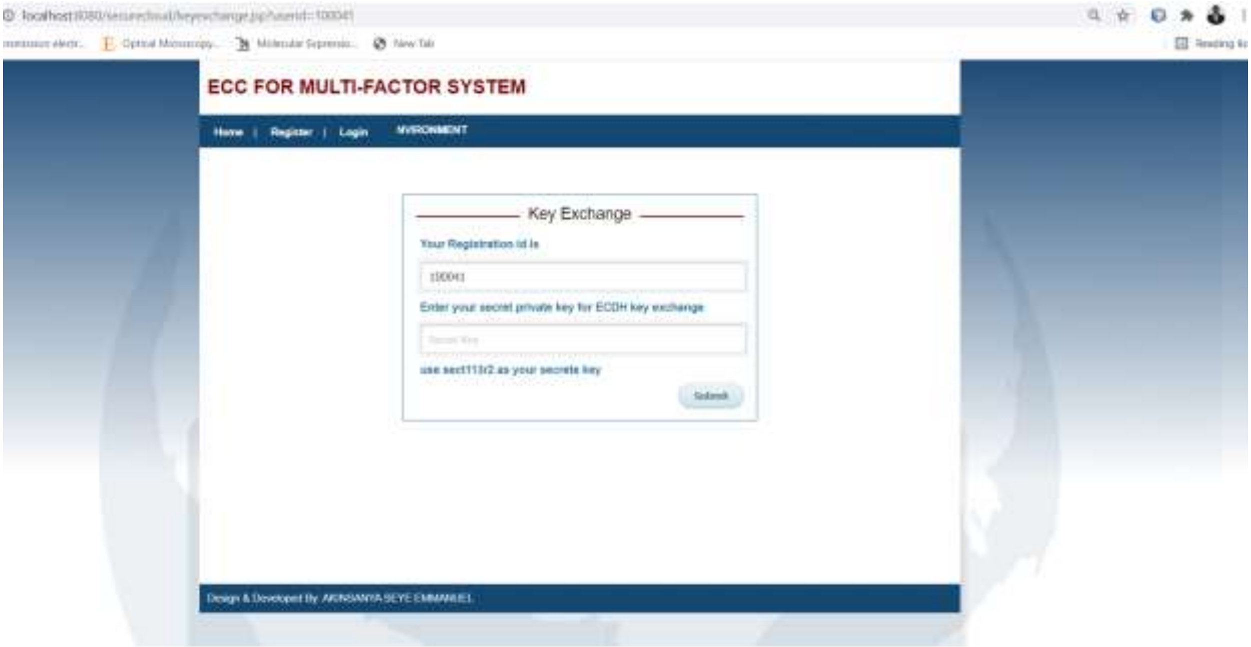


Figure 5-Level-2 Data Flow Diagram (DFD)

The features the user can access once a successful login to the application is done are shown in Figure 5. Once the user successfully login to the application, he/she will be taken to My Account Page where all the details stored in the database will be displayed with the ability of changing/updating the details, save the changes, and then click the submit button. My Account Dashboard is a page that serves as a navigation page where users can encrypt, decrypt, and download files. If the user wants to upload a new file, he will go to the encryption tab and enter the encryption key, which is the same as the secret key used during registration. The user will then be asked to select the desired file to be uploaded from the local system where the application is running. The user will then click submit to finish the encryption and save the file. Once the encryption is achieved, the user will click on the decryption tab to input the given key at the point of encryption and proceed to download the file. Once the chosen file is decrypted, the download will immediately start when the user hits the download button.

4. RESULTS AND DISCUSSIONS

Huang et al., (2021) and Arumugam et al., (2021) in the literature expressed how ECC and RSA were applied to secure data on the cloud. They both claimed that their methods were efficient, but they did not compare the efficiency of their methods with another scheme. In this research, Elliptic Curve Cryptography is proposed for securing multi-factor systems in cloud computing and evaluates its performance in order to highlight its advantages over the RSA algorithm. The public key generation time, encryption and decryption times using different key sizes were the basis of the comparison. The results showed that ECC takes less time compared to RSA in terms of confidentiality, integrity, and authentication.



ECC FOR MULTI-FACTOR SYSTEM

Key Exchange

Your Registration id is

100041

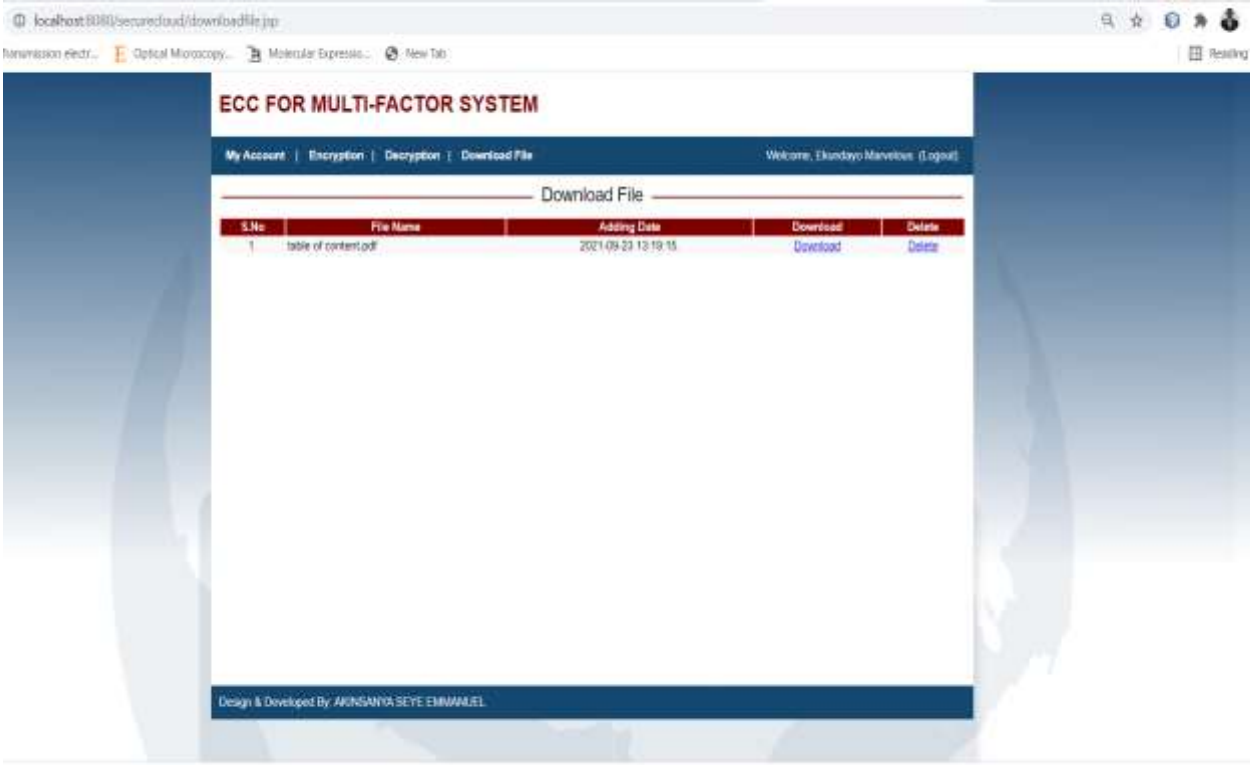
Enter your secret private key for ECDH key exchange

use sect113r2 as your secrete key

Design & Developed By AKINSANYA SEYE EMMANUEL

Figure 6-Key Exchange

Key Exchange was performed using the Elliptic Curve Diffie Hellman (ECDH) key agreement algorithm. Here the user would supply the registration number and the secret key to enhance the key exchange, as shown in Figure 6. Once the details are validated, the application will generate a unique user ID and an OTP will be sent to the user’s email address. The user would have to save the unique ID for subsequent logins to the application, and the user would be able to access every feature of the application.


localhost:8080/securedcloud/downloadfile.jsp
Transmission electr...
Optical Microscopy...
Molecular Expressio...
New Tab
Reading
My Account | Encryption | Decryption | Download File
Welcome, Ekundayo Marvelous (Logout)

ECC FOR MULTI-FACTOR SYSTEM

Download File

S.No	File Name	Adding Date	Download	Delete
1	table of content.pdf	2021-09-23 13:19:15	Download	Delete

Design & Developed By: AKINSANYA SEYE EMMANUEL

Figure 7-Download dashboard

This is the user's account download dashboard. Once the user's file has been encrypted, it will be stored on the server for download. The user will be able to download the encrypted file from this page by clicking on the download button, as shown in Figure 7. After clicking the download button, the user would be prompted to enter the ECC key that was provided after the encryption.

The experiment was performed using the ECC algorithm and its performance was evaluated with the RSA algorithm. RSA and ECC are popularly known as public cryptography algorithms. They are used for the encryption and decryption of data. ECC is preferred over RSA because of its smaller key, which results in lower power consumption and therefore enhances system performance.

Table 1-Security level of RSA and ECC

Size of bits	ECC key size	RSA key size
112	224	1024
128	256	3072
192	384	7680
256	521	15360

Table 1 illustrates the comparison between ECC and RSA algorithms according to the key size. The ECC that was adopted in this project utilized a smaller key size, which made it faster than the RSA algorithm and enhanced system performance by reducing power consumption.

Table 2-Configurations of The Test Machine

Parameter	Test machine 1	Test machine 2	Test machine 3
Speed	2.20Gz	2.67Gz	3.10Gz
Memory	1.00GB	3.00GB	4.00GB

The experiment was performed on three different machines, as shown in Table 2 with the set of configurations used.

Table 3-Time Taken By The Public Key Generation

Size of key	Test machine 1	Test machine 2	Test machine 3
RSA-1024	385307698ns	5187964668ns	97111209ns
ECC-224	26596114ns	14194733ns	12255809ns
RSA-3072	13422086748ns	12726719367ns	1562264922ns
ECC-256	27791115ns	14904971ns	12685866ns
RSA-7680	296322237232ns	323216127436ns	61033987666ns
ECC-384	32048976ns	19959395ns	15745928ns
RSA-15360	1428639568941ns	5698444364116ns	920175140712ns
ECC-521	37654887ns	22440413ns	20383198ns

Table 3: Shows the time taken by the public key generation in nanoseconds (ns) using different key sizes. The results show that ECC takes less time compared to RSA. The time taken to generate the private keys was ignored by the two algorithms.

Table 4-Time Taken For Encryption And Decryption

Parameter	Time Taken for Encryption			Time Taken for Decryption		
Key Size(bit)	Test Machine 1	Test Machine 2	Test Machine 3	Test Machine 1	Test Machine 2	Test Machine 3
RSA-3072	5727698ns	3894194	2604827ns	584453792ns	466162557ns	412829667ns
ECC-256	11241686ns	7368374	6366438ns	12391891ns	4921616ns	4508154ns
RSA-7680	20791425ns	30491425	30712328ns	8747349045ns	6663514152ns	6172930987ns
ECC-384	16436042ns	14101575	12139272ns	12545874ns	8000465ns	7324881ns
RSA-15360	72591742ns	129103126	121774673ns	68343461676ns	134986245214ns	123349513962ns
ECC-521	26760363ns	17427949	16983460ns	18304369ns	12373524ns	11736032ns

Table-4 demonstrates the time taken by both algorithms to perform encryption and decryption**.** The results clearly show that the ECC algorithm is more efficient than the RSA algorithm, as ECC performs encryption and decryption faster than the RSA algorithm for different key sizes.

5. CONCLUSION AND RECOMMENDATION

The focus of this study is on the security of user data in the cloud. The system is designed to allow users to upload files of various formats with sufficient security concepts such as secure OTP, encryption, and decryption over the internet. The adopted method is the use of an ECC algorithm to achieve the encryption and decryption process. The experiment reveals that ECC performs very well compared to the RSA algorithm from other similar works found in the literature.

For future work, using a more robust hosting service, it will be possible to reduce the processing time taken in all encryption-decryption procedures. To provide cutting-edge security and protection, several innovative hybrid cryptography methods and systems can be implemented.

6.REFERENCES

[1] Adesh Kumari, M. Yahya Abbasi, Vinod Kumar & Akber Ali Khan, "A secure user authentication protocol using elliptic curve cryptography," Journal of Discrete Mathematical Sciences and Cryptography, 22:4, 521-530, 2019. DOI:10.1080/09720529.2019.1637155.

[2] Anand, S., & Perumal, V., "EECDH to prevent MITM attack in cloud computing," Digital Communications and Networks, 5(4), 276-287, 2019.

[3] Arezou Ostad-Sharif, Dariush Abbasinezhad-Mood, Morteza Nikooghadam, "Efficient utilization of elliptic curve cryptography in design of a three-factor authentication protocol for satellite communications," Computer Communications 147, 85–97, 2019.

[4] Arumugam, M., Deepa, S., Arun, G., Sathishkumar, P., & Jeevanantham, K. ,"Secure data sharing for mobile cloud computing using RSA," In *IOP Conference Series: Materials Science and Engineering* (Vol. 1055, No. 1, p. 012108), (2021, February).

[5] Dhamodaran, M., Punarselvam, E., Varshan, S. D., Kumar, P. D., Saravanan, C., & Prathap, K., "Security and privacy of sensitive data in cloud computing using RSA," International Journal of Scientific Research in Science and Technology, 657-661, 2021.

[6] Domingo-Ferrer, J., Farras, O., Ribes-González, J., & Sánchez, D., "Privacy-preserving cloud computing Cloud Computing on sensitive data: A survey of methods, products and challenges," Computer Communications, 140, 38-60, 2019.

[7] Ene D., Anireh V.I.E., Matthias D., " Data link layer encryption for the internet of things using elliptic curve cryptography over visible light communication channel," International Journal of Computer Sciences and Engineering, Vol.-8, Issue-2, 2020. E-ISSN: 2347-2693.

[8] Huang, H., Miao, X., Wu, Z., & Wei, Q., "An Efficient ECC-Based Authentication Scheme against Clock Asynchronous for Spatial Information Network," Mathematical Problems in Engineering, 2021.

[9] IbrahimA. A., CheruiyotW., and KimweleM. W., "Data Security in Cloud Computing Cloud Computing with Elliptic Curve Cryptography," *International Journal of Computer (IJC)*, vol. 26, pp. 1-14, 2017.

[10] R. J. Essa, N. A. Abdulah, and R. D. Al-Dabbagh, "Steganography Technique using Genetic Algorithm", Iraqi Journal of Sciences, vol. 59, no. 3A, pp. 1312–1325, Jul. 2018.

[11] [11] Kumar, D., & Grover, H. S., "A secure authentication protocol for wearable devices environment using ECC," Journal of Information Security and Applications, 47, 8-15, 2019.

[12] Kumar, V., Jangirala, S., & Ahmad, M., "An efficient mutual authentication framework for healthcare system in cloud computing. Journal of medical systems, 42(8), 1-25, 2018.

[13] Kumari, A., Jangirala, S., Abbasi, M. Y., Kumar, V., & Alam, M., "ESEAP: ECC based secure and efficient mutual authentication protocol using smart card," Journal of Information Security and Applications, 51, 102443, 2020.

[14] Mai Rady, Tamer Abdelkader, Rasha Ismail, "Integrity and Confidentiality in Cloud Outsourced Data," Ain Shams Engineering Journal 10, 275–285, 2019.

[15] Roy, S., Das, A. K., Chatterjee, S., Kumar, N., Chattopadhyay, S., & Rodrigues, J. J., "Provably secure fine-grained data access control over multiple cloud servers in mobile cloud computing Cloud Computing based healthcare applications," IEEE Transactions on Industrial Informatics, 15(1), 457-468, 2018.

[16] Sasidevi Jayaraman, Sugumar Rajendran and Shanmuga Priya P., "Fuzzy c-means clustering and elliptic curve cryptography using privacy preserving in cloud," Int. J. Business Intelligence and Data Mining, Vol. 15, No. 3, 2019.

[17] Shen, Y., Sun, Z., & Zhou, T., "Survey on Asymmetric Cryptography Algorithms," In 2021 IEEE International Conference on Electronic Information Engineering and Computer Science (EIECS) (pp. 464-469), September, 2021.

[18] Sowjanya K., Mou Dasgupta, Sangram Ray, "An elliptic curve cryptography based enhanced anonymous, authentication protocol for wearable health monitoring systems," Springer-Verlag GmbH Germany, 2019.

[19] Wang H., He D., Ji Y., "Designated-verifier proof of assets for bitcoin exchange using elliptic curve cryptography," Future Generation Computer Systems, 2017. http://dx.doi.org/10.1016/j.future.2017.06.028.