Current Journal of Applied Science and Technology

39(47): 27-34, 2020; Article no.CJAST.64045
ISSN: 2457-1024
(Past name: British Journal of Applied Science & Technology, Past ISSN: 2231-0843,
NLM ID: 101664541)

A Hybrid Cryptographic Scheme for Improving Cloud Security Using ECC and TDES Algorithms

Simranjit Kaur[1*] and Lokesh Jain[1]

[1]Department of Electrical Engineering and Information Technology, Punjab Agricultural University, Ludhiana, Punjab, India.

Authors' contributions

This work was carried out in collaboration between both authors. Author SK designed the study, and wrote the first draft of the manuscript and the literature searches. Author LJ performed the analysis, analyses of the study. Author SK managed the experimental work. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/CJAST/2020/v39i4731184
Editor(s):
(1) Dr. Elena Lanchares Sancho, University of Zaragoza, Spain.
Reviewers:
(1) A. Prasanth, PSNA College of Engineering and Technology, India.
(2) Heru Susanto, Indonesian Institute of Sciences, Indonesia.
Complete Peer review History: http://www.sdiarticle4.com/review-history/64045

Review Article

Received 25 October 2020
Accepted 30 December 2020
Published 31 December 2020

ABSTRACT

The evolving cloud technology is capable of accommodating volumes of business processes. This feature attracts many individuals and organizations to store their data over cloud. But there are many security issues that require a deep insight. In this research, articles and surveys on cloud security have been reviewed to identify the issues. Efforts have been made to increase cloud data security by devising a hybrid cryptographic algorithm namely Hybrid Elliptic Curve Cryptography - Triple Data Encryption Standard (ECC-TDES). The hybridized algorithms make data more secure and immune to malicious attacks. The proposed algorithm was applied to database of a cloud-based web application and tested with audio, video, image and text files ranging between 10-100 kb file sizes to record performance metrics such as: encryption time, decryption time and accuracy. The recorded parameter values were compared with individual ECC and TDES algorithms. The findings indicated that ECC-TDES takes more time to encrypt/decrypt files but gives highest accuracy with 0.01% error rate.

**Corresponding author: E-mail: simranjitsvk@gmail.com;*

Keywords: Cloud computing; data security; security issues; hybrid cryptography; decryption; elliptic curve; triple DES.

1. INTRODUCTION

Cloud Computing (or simply Cloud) has become a popular term in the IT world over the last few years. The United States of America's National Institute of Standards and Technology (NIST) defines cloud computing as "…a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [1]. The basic concept for cloud architecture is to isolate the operating system from the actual hardware overlay, through virtualization and automation techniques [2]. Based on usage and consumer requirements, the services provided by a cloud can be categorized as;

1. Software-as-a-service (SaaS): In this type of service model, a third-party provider hosts applications and underlying database, which can be accessed by the customer over internet. Examples include Google Apps, Dropbox, Netflix etc.
2. Platform-as-a-service (PaaS): The third-party provider delivers hardware and software tools over the internet, usually required for application development. Windows Azure, Google App engine, IBM Cloud Foundry are some examples working on PaaS service model.
3. Infrastructure-as-a-service (IaaS): Here the vendor provides users access to computing resources such as servers, storage and networking. Examples are AWS EC2, Google Compute Engine Cisco Metapod etc.

The constant expansion and rising use of cloud services leaves security problems unparalleled. This requires continuous study by IT and information security experts. As a result, CSA and other organizations that promote cloud computing security have been set up to help make the best use of cloud computing tools and provide compact defense [3]. This applies both to technological issues and to issues relevant to standardization, management model, legislation and regulations. Solving cloud computing security from a technological point of view is therefore not sufficient, it also needs the cooperation of academics in information technology, industry and government departments [4].

One of the objectives of this research was to observe threats to cloud. For this purpose, various research articles discussing cloud security threats and solutions to it were reviewed. As per the CSA Comprehensive Report of 2019, the key risks to cloud security can be narrowly defined as: Data Breaches; Insecure interfaces and APIs; Lack of cloud security architecture and strategy; Insufficient Identity, Credentials, Access and Key Management; Weak Control Panel; Failure of Meta-structure and Appli-structure; Misconfiguration and Inadequate Change Control; Insider Threat; Limited Cloud Use Visibility; Account hijacking; Abuse and Nefarious Use of Cloud Services [5].

The surveys conducted to recognize significant cloud security issues disclose the following challenges: Eavesdropping; Hypervisor viruses; Legal Interception point; Abuse and nefarious use of Cloud Computing; Insecure application programming interfaces; Trusted transaction; Risk of multiple Cloud tenants; Smart phone data slinging; Malicious insiders; Virtual machine security; Shared technology vulnerabilities; and Service and traffic hijacking [6]. Identifying the cloud applications like IoT, cloud storage, cloud computing infrastructure, online gaming, security as a service, and Big Data; the key security concerns found are: privacy protection; data security and confidentiality; data audit; authentication and access control policy; virtual machine security and automated protection [4].

In general, cloud computing is a modern paradigm focused on the growth, usage and interaction of the Internet. As noted, a suitable set of security standards for each web application or service needs to be applied in the cloud [7,8]. From the studies analyzed, the most important cloud security problems in the last decade includes Insufficient Identity, Credential, Access and Key Management, Data Breaches and Data loss, Insecure API's, Insider threat, Account hijacking, Lack of cloud security architecture and strategy. Different researchers have used different approaches to resolve these issues.

According to Kaul and Shaikh [9] cites AES and ECC as the best two symmetric and asymmetric encryption algorithms. On this basis, the authors have developed a hybrid cryptography model with a combination of AES and Blowfish. For this, they used Message Digest 5 for data integrity, ECDH for key exchange and ECDHA for digital signature. The hybridized AES and Blowfish systems have characteristics of both algorithms and are powerful against the same flaws that the parent algorithms are resistant to. They also claimed that ECC is the best asymmetric encryption technique and offers the highest power per bit of any cryptosystem, resulting in faster processing, lower power consumption and memory. This also offers a mechanism for achieving high-speed, efficient and flexible implementation of authentication protocols and key agreements.

As per Rizvi and Tandra [10] stress the vulnerability to cloud storage, noting that security and reliability are the key concerns. In order to preserve and secure insecure data, a combination of Encryption (using different cryptographic algorithms), Authentication (providing username and password) and Authorization techniques was suggested. They used and contrasted TDES and AES cryptographic algorithms to encrypt the data for the "ifoodbag" web application. The results were accepted by AES as a superior algorithm.

According to Anonymous [11] noted that some of the risks are due to inaccuracies that arise due to human error (which cannot be avoided). Data, if not properly backed up, can be lost even if acts are not malicious. As discussed above, safety risks are inevitable, but luckily successful approaches have been and can be easily introduced and checked with acceptable results. Key protection and data encryption (cryptography) has been found to dramatically reduce the risk of a data breach by an average of 40% in the last few years.

Kumar and Singh [12] analyzed the cryptographic algorithms DES, TDES, AES, Blowfish, RSA and ECC based on their key size, block size, rounds and many more parameters. The purpose of cryptography is explained by four parameters of confidentiality, integrity, authentication and access control. The conclusion was that the TDES (or 3DES) algorithm is better suited to data protection because it uses three keys to encrypt and decrypt. In fact, the key size of TDES is much tougher for cracking and thus recommended for use in better data protection.

2. METHODOLOGY

The focus of this research has been on identifying the various security problems found in the cloud services and techniques used to remove them. Efforts have been made to apply some of these strategies to web applications running in a cloud environment. The web application called "Secure Cloud Computing" has been designed to be provided under the cloud-based Software-as-a-Service (SaaS) model. Network protection, physical security, legal enforcement, disaster or risk management are not part of the scope of this study. The inspiration for the proposed safety measure has been extracted from prior study conducted by other researchers.

Data passing across the network or stored as plain text in cloud is a significant security threat. In fact, cloud is a virtual network used by several entities or individual customers, which may or may not be interlinked. This multi-tenancy makes cloud data freely available to all of the services that are connected or distributed. Cryptography is one of the methods to encrypt and defend the confidentiality of customer data from malicious attacks and unauthorized access.

Each cryptographic algorithm has its own strengths and weaknesses. Hybrid cryptographic scheme is one way of integrating and leveraging the strengths of various algorithms to solve their limitations (mostly the simplicity of a public-key cryptosystem with the reliability of a symmetric-key cryptosystem). This results in enhanced protection techniques. Under this principle, one algorithm is used to encrypt a secret key, and this secret key is used to encrypt plain text using another technique. The hybridized algorithm inherits the properties of the parent algorithms; thus, it is safe against the same attacks that the parent algorithms are resistant to [13].

This work proposes a new hybrid cryptosystem that uses Elliptic Curve Cryptography (ECC) for key generation and Triple DES (TDES) algorithm for data encryption (asymmetric and symmetric algorithms, respectively). Secure Hash Algorithm 3 (SHA-3) used produces unique fingerprints for every file, ensuring authentication. The proposed algorithm has been tested with text, audio, image and video files to improve protection and efficiency on different file systems. Performance metrics have been reported in the form of encryption and decryption time; throughput; and accuracy of the reconstructed data.

ECC finds its primary use in cryptosystems in two ways such as ECDSA and ECDH. ECDSA (Elliptic Curve Digital Signature Algorithm) is the DSA (Digital Signature Algorithm) digital signature standard analogue for elliptical curves used to provide verifiable digital signatures for data messages. Elliptic Curve Diffie-Hellman (ECDH) is a version of the Diffie-Hellman algorithm that uses elliptic curve properties to generate key pairs for two A and B communication participants and to decide how to exchange public keys over an unsafe channel. It is standardised and free to use cryptosystem. The ECDH key exchange system can be combined with symmetric encryption methods and tailored to a number of data protection solutions [14].

To understand how the proposed algorithm works, it is important to clarify the generation of the ECC key and the arithmetic behind it. A finite field $n(2^m)$ as defined by Stallings (2017) consists of 2^m elements, along with additional and multiplication operations that can be specified over polynomials.

For elliptic curves over $n(2^m)$, a cubic equation (Eq. 1) is used in which all the variables and coefficients take on values in $n(2^m)$ for certain numbers m and in which calculations are made using arithmetic rules in $n(2^m)$. Fig. 1 demonstrates general elliptic curve representation.

$$y^2 = x^3 + ax + b \tag{1}$$

The prerequisites for a key exchange in elliptic curve are;

- The elliptic equation of curves defined over a finite field
- A large integer q representing either a prime number or an integer of the 2^m type parameters of the elliptic curve a and b satisfying Eq. (1)

ECDH key exchange algorithm can be explained as; given an elliptic curve over finite prime field E(a,b). Communication participants A and B agree on a point $G \in E(a,b)$ which is open to each participant in the communication medium. The participant A secretly selects probabilistic positive integer n_a, calculates public key $P_a = n_aG$, and forwards it to the party B. The user B also chooses probabilistic positive integer n_b, calculates $P_b = n_bG$ and sends it to the participant A. The shared secret $P = n_an_bG$. The participant A calculates P, by multiplying the received point n_bG with the secret private key n_a. The participant B respectively calculates P by multiplying the received point n_aG with the secret private key n_b [14]. The resulting keys will be such that K_a equals K_b (let resultant be K), hence completing the key exchange process. To break this scheme, an attacker would need to be able to compute K given G and KG, which is assumed to be hard, making the function irreversible. This trapdoor feature is called discrete logarithm problem for elliptic curves.

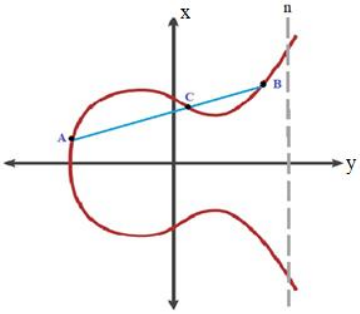

Fig. 1. General representation of elliptic curve

As mentioned earlier, the ECDH key exchange can be combined with any symmetric encryption technique. TDES is the strongest data protection algorithm as it uses three keys to encrypt and decrypt, and the key size of TDES is considerably difficult to crack [12]. So, TDES is the tool chosen for data encryption. As the name suggests, the Triple Data Encryption Standard (TDES) encrypts each data block thrice using symmetric key block ciphers. In the proposed algorithm, the public keys Pa and Pb (say, T1 and T2) generated by ECC are forwarded to TDES for the encryption process following the basic TDES procedure below:

1. The DES encryption is performed in plain text using the T1 key.
2. Reverse DES using the T2 key, performed on the encrypted file generated in Step 1.
3. T1 executes DES again to encrypt the file obtained from the T2 encryption.

Because the same key is used for encryption and decryption, DES can be cracked by Brute force attack i.e. trying as many keys possible as possible. Similarly, Double DES is vulnerable to Man-in-the-middle attack. Triple DES with two

56-bit keys holds 2^{112} combinations is harder to crack with such attacks.

2.1 Implementation

In order to understand the value of the proposed algorithm, it is necessary to implement it and to evaluate its output in an analytical manner. For this reason, the algorithm has been applied to a web application database called "Secure Cloud Computing" running in a cloud environment. The program was developed using the ASP.NET framework included in Microsoft Visual Studio. Data submitted to the cloud are stored in a database powered by SQL Server 2019. The work low diagram of the system is presented in Fig. 2 and can be explained in two parts as;

2.1.1 File encryption

Step 1: Upload the file and forward it for key generation.

Step 2: Perform the ECC key generation which would produce a set of public and private keys binding to user’s profile and the files uploaded.

Step 3: The next step in the encryption process is forwarding the public key generated in the previous step for encrypting the file using TDES.

Step 4: The final step in encryption process of the file is to store the cipher data generated by executing the above listed steps in a database.

2.1.2 File decryption

The decryption of a file is encryption performed in reverse chronological order. The procedure followed for decryption is as mentioned below.

Step 1: Load the encrypted file bytes from the database and forward to decryption module.

Step 2: Implement the TDES algorithm using private key generated by ECC, to decrypt the file.

Step 3: The decrypted file is downloaded to the user’s device. Also, it gives encryption time, decryption time and other parameters in output.

The “Secure Cloud Computing” web application encrypts files until they are downloaded, maintaining confidentiality and integrity. Users can use the cloud to store images, audio, video and text files in any format. To test the efficiency of the proposed technique, a comparison was made between the hybrid ECC-TDES algorithm, the ECC algorithm and the TDES algorithm. For this reason, the experiment was run on both the local machine and the hosted application.

3. RESULTS AND DISCUSSION

The performance analysis of the proposed methodology was compared to the individual ECC and TDES algorithms. All calculations were made on the basis of time to be encrypt, time to decrypt, accuracy and throughput. The experiments were carried out using four file types: text, image, audio and video, each with two different file sizes ranging from 10 KB to 100 KB. The experiment was also run 10 times each time on the localhost and remote machine by hosting the application on the cloud. The raw data observations were recorded in the form of accuracy, time for encryption and time for decryption from the "Results" page of the web application as shown in Figs. 3, 4 and 5

Table 1. Comparative accuracy and throughput for different files

		Average encryption throughput (KB/ms)			**Average decryption throughput (KB/ms)**			**Average accuracy**		
File Type	**File size (KB)**	**ECC**	**TDES**	**ECC-TDES**	**ECC**	**TDES**	**ECC-TDES**	**ECC**	**TDES**	**ECC-TDES**
Text (.txt)	9.94	12.57	12.40	12.45	13.10	12.98	11.13	98.55	97.48	99.90
Text (.txt)	100.00	14.36	12.32	11.28	13.10	12.98	11.13	98.55	97.48	99.90
Image (.jpg)	10.00	13.47	11.33	10.76	14.03	11.81	10.84	98.35	97.39	99.90
Image (.jpg)	100.00	13.30	11.80	10.92	13.86	12.33	11.15	98.03	97.55	99.90
Audio (.mpeg)	10.70	13.49	12.09	10.89	14.06	12.62	9.51	98.48	97.50	99.90
Audio (.mpeg)	99.30	12.81	12.07	11.94	13.35	12.62	11.44	98.54	97.51	99.90
Video(.3gp)	24.60	12.13	12.63	11.06	12.63	13.19	11.31	98.61	97.47	99.90
Video(.mp4)	111.00	13.49	11.36	10.84	14.05	11.88	11.18	98.48	97.58	99.90

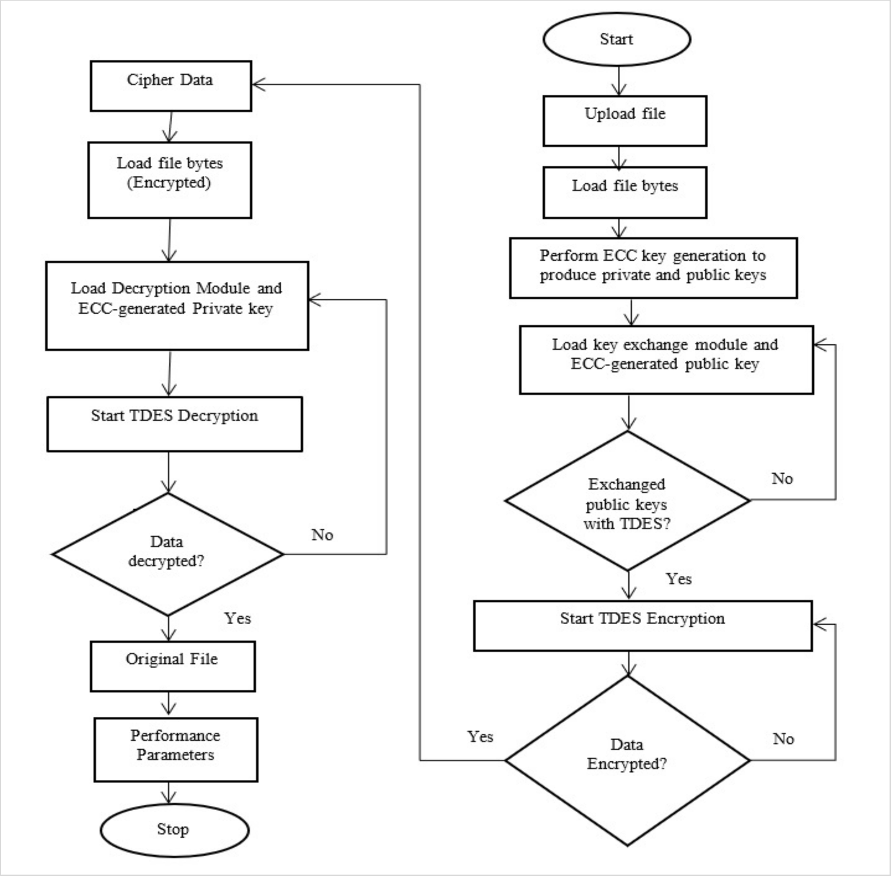
Start
Upload file
Load file bytes
Perform ECC key generation to produce private and public keys
Load key exchange module and ECC-generated public key
Exchanged public keys with TDES?
No
Yes
Start TDES Encryption
Data Encrypted?
No
Yes
Cipher Data
Load file bytes (Encrypted)
Load Decryption Module and ECC-generated Private key
Start TDES Decryption
Data decrypted?
No
Yes
Original File
Performance Parameters
Stop

Fig. 2. Work flow diagram for proposed scheme

(comparisons made in table I). Encryption Time is the key parameter that indicates the time taken by any algorithm to encrypt uploaded material. Likewise, decryption time indicates the time taken to decrypt and recover the original file from the database. The recovered file after decryption will have a negligible error rate. The accuracy of the decrypted file depends on the data frame recovery rate.

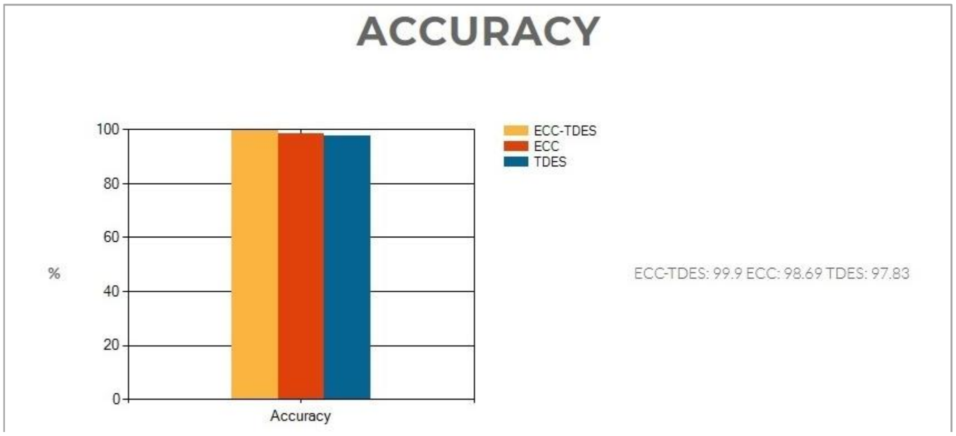

Fig. 3. Accuracy

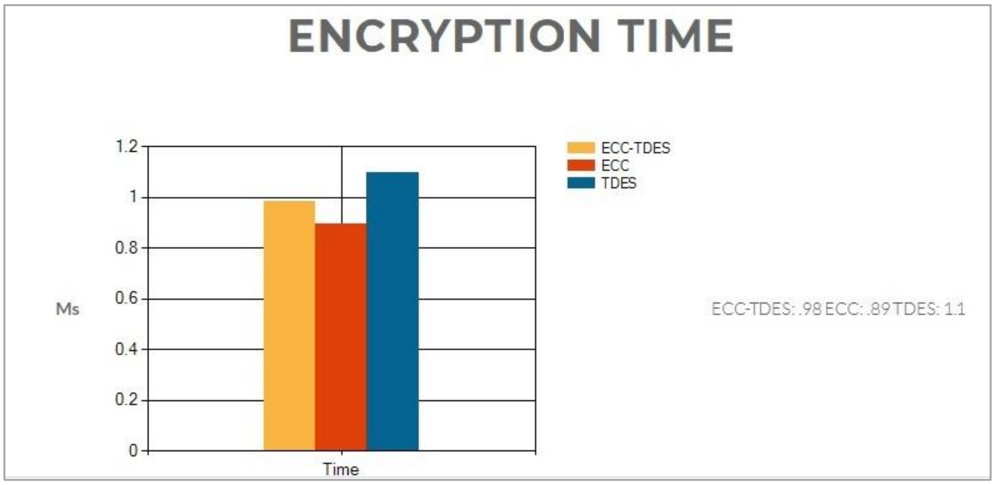

Fig. 4. Encryption time

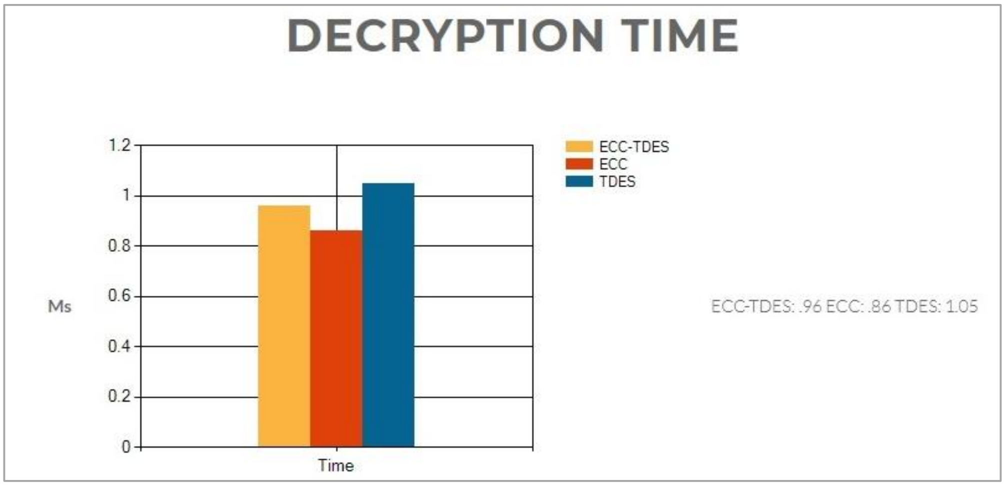

Fig. 5. Decryption time

The encryption and decryption throughput were determined separately as the file sizes in kilo bytes divided by the average encryption time (in milliseconds) and, in the case of the decryption, the file size in kilo bytes divided by the average decryption time (in milliseconds) [15].

4. CONCLUSION

Data encryption and decryption were carried out separately in the above experiment using the proposed hybrid ECC-TDES, ECC and TDES algorithm. The results indicate that the ECC provides the best output with the highest efficiency, which is also supported by other researchers. This finding also supports the selection of the ECC for key generation in the proposed hybrid cryptography scheme. Another point is that ECC-TDES has the highest accuracy in decrypted data.

While the experiment shows that the ECC-TDES hybrid is poor in performance compared to ECC and TDES, the improved TDES hybrid structure provides more security because of increased complexity. ECC is the best asymmetric encryption technology algorithm with a 160-bit key size that takes 9.6x 1011 MIP years to break down the best-known attack [16]. If evaluated individually, the findings of this work have shown that ECC-TDES is a less time-consuming and highly reliable cryptographic system and provides high security for cloud storage. ECC-TDES is a highly accurate algorithm that provides reliability and authenticity to the cloud data. As cloud computing is focused on fast, exponentially increasing networks, fast and lightweight key generation algorithms like ECC-TDES are crucial for expanding cloud performance. The only downside of the proposed cryptographic scheme is the reduced performance that could be attributed to the increased complexity of the hybridization of two algorithms.

ACKNOWLEDGEMENTS

Authors acknowledge the support provided by the Department of Electrical engineering and Information Technology, Punjab Agricultural University for completing the research work.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Grance T, Mell P. The NIST definition of cloud computing. NIST Special Publication. 2011;800-145.
2. Gandhi C, Kaushik S. Cloud data security with hybrid symmetric encryption. Int Conf on Techs in Info and Comm Tech. GGS Indraprastha University, New Delhi. 2016;36-640. DOI: 10.1109/ICCTICT.2016.7514656
3. Feng D, Zhang M, Zhang Y, Xu Z. Study on Cloud Computing Security. Journal of Software. 2010;22(1):71-83.
4. Ou Y. The concept of cloud computing and the main security issues in it. B. thesis, Turku University of Applied Sciences, Turku, Southwest Finland; 2015.
5. Cloud Security Alliance. Top Threats to Cloud Computing: The Egregious 11; 2019.
6. Maddineni VSK, Ragi S. Security Techniques for Protecting Data in Cloud Computing. M. thesis, Blekinge Institute of Technology, Karlskrona, Sweden; 2011.
7. Prasanth A, Jayachitra S. A novel multi-objective optimization strategy for enhancing quality of service in IoT-enabled WSN applications. Peer-to-Peer Networking and Applications. 2020;13(6): 1905-1920.
8. Prasanth A, Pavalarajan S. Implementation of efficient intra-and inter-zone routing for extending network consistency in wireless sensor networks. Journal of Circuits, Systems and Computers. 2020;29(08):2050129.
9. Kaul V, Shaikh AP. Enhanced Security Algorithm using Hybrid Encryption and ECC. IOSR J Comp Engg. 2014;16(3):80-85.
10. Rizvi SI, Tandra SA. Security for Cloud Based Services. M. thesis, School of Information and Communication Technology (ICT), KTH Royal Institute of Technology Stockholm, Sweden; 2014.
11. Ponemon Institute LLC. The 2018 Global Cloud Data Security Study; 2018.
12. Kumar S, Singh S. Analysis of Various Cryptographic Algorithms. IJARSET. 2018;5(3).
13. Cramer R, Shoup V. Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack. SIAM J Comp. 2003;33(1):167-226.
14. Magons K. Applications and Benefits of Elliptic Curve Cryptography. In Proc. SOFSEM. 2016;1548:32-42.
15. Sandha KS, Singh G, Singla AK. Through Put Analysis of Various Encryption Algorithms. Int J Comp Sci Engg. 2011;2(3):527-529.
16. ElTaweel GS, Hassan HE and Tahoun M. A robust computational DRM framework for protecting multimedia contents using AES and ECC. Alexandria Engg Jour. 2020;59(3):1275-1286.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/64045