Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 44 No. 7
July 2023

Cloud Storage Data Security Using BLOWFISH-ECC Model

J Venkata Giri,
Assistant Professor
Department of Computer Science and Engineering
Sri Venkateshwara College of Engineering Bengaluru, Karnataka, India
venkatajgiri9@gmail.com

Sanjeev C Lingareddy,
Professor& HOD
Department of Computer Engineering
Sri Venkateshwara College of Engineering
Bengaluru, Karnataka, India
hodcs@svcengg.edu.in

Abstract—In a cloud computing system, the end user can store and retrieve personal data more affordably over remote storage via an Internet connection. The user can access data whenever they want and from anywhere. But cloud-based data transmission isn't always safe. Because the end user can only access the data with a third party's assistance, data integrity and authentication can be jeopardized. Furthermore, because cloud computing allows multiple users to access and retrieve their data online via different Internet connections, sensitive user data may be lost, exposed, or leaked in a number of different locations. Numerous algorithms and protocols have been created in order to guarantee the security and integrity of the data utilizing cryptographic methods like Elliptic Curve Cryptography (ECC). This study offers a way for exchanging cloud data that is secure and reliable while maintaining the integrity of the data. The proposed system primarily uses the ECC and Blowfish (BF) approach to provide authentication and data integrity. When compared to existing methods, the experimental results show that the suggested strategy is efficient and yields superior outcomes.

Keywords— Elliptic Curve Cryptography (ECC); Blowfish (BF); cloud computing; data security; authentication; data integrity

I. INTRODUCTION

Unrestricted database storage, networks, and communications, among other services, are all readily available at any location thanks to the many distinguished computing models that include cloud computing. Due to its alluring characteristics, people are relying more and more on the cloud, which has resulted in enormous amounts of data and raised worries about privacy and security. Cloud users may unintentionally or willfully cause significant downsides of cloud services, particularly data security and data breaches. As a result, unauthorized and unauthenticated sources should be subject to restrictions on data access. Since users may be permitted to reuse the APIs and data, devices can also be a source of data breaches. Thus, the primary use of cryptographic methods is to safeguard data stored in the cloud by implementing encryption/decryption methods with the use of various keys. Asymmetric key encryption of data and symmetric key encryption of data are the two methods that are used to encrypt data using keys [1].

Public-key cryptography is another name for asymmetric key encryption. It has two keys, a public key and a private key, which are used to encrypt and decode the communication, respectively. However, symmetric key encryption, which encrypts data with a single private key and then decrypts it, is also used to secure data. The message is encrypted using the private key to protect it from possible hacker activities. It has been determined that the key magnitude, which must be big enough to assure good safety, is what makes it difficult to adopt symmetric cryptographic algorithms.

I. RELATED WORK

The Data Encryption Standard (DES) was intended to be replaced by the BlowFish (BF), which is built on the symmetric basicencryption procedure [2]. Since BF's key size is 128 bits instead of DES's 64 bits, it is significantly faster than DES. Elliptic Curve Cryptography (ECC), based on a symmetric key encryption [3], outperforms other cryptographic methods because it offers greater security with a

smaller key size than other currently employed schemes like the Rivest-Shamir-Adleman (RSA) algorithm [2], resulting in smaller latency and less computational/hardware complexity.

Chen et al. [4], who employed BF and ECC to increase system security, suggested using BF alongside ECC. For the data transfer, they nevertheless additionally utilized Shamir's secret sharing key. In this research, we suggested a hybrid BF and ECC approach for cloud storage data security without a third party. The proposed hybrid approach (BF-ECC) is used to effectively maintain system security when using cloud storage. The major goal of this hybrid strategy is to shorten the time required to maintain system security while reducing the key size of the data. Although there are numerous authentication methods, computational time and cost are not decreased.

Recent developments in cloud storage have given consumers access to a variety of services that enable experienced data encryption and decryption without the need for a third party. This improves the system's ability to increase its security and the effectiveness of the storingon behalf ofsafeconveniencein addition quick data retrieval. Through this service, it is simple for various user types to share cloud resources, and the use of cryptographic algorithms increases the system's capacity by securing more storage [7].

ECC, a type of efficient cloud services that is utilized for data encryption and decryption, is offered. These services employ asymmetries in their data encryption and decryption. It employs RSA data encryption and decryption, which is more efficient than other methods because the key size is smaller than with symmetric encryption and decryption. For instance, utilizing the RSA algorithm with ECC will reduce the size of the key's data from 1020 bits to 163 bits. Additionally, ECC is better suited for network access via cellphones. In order to extort people, many hackers are keen to obtain their personal information or other data. As a result, ECC was developed primarily for users who access their data through insecure devices and are thus at risk of hacking [8].

II. PROBLEM STATEMENT

Information safety is currently a big difficult that can be exploited in a variety of ways overinner or exterior sources. Numerous encryption approaches are used toward safe information communication on the Internet. The issue with these systems is that in order to guard the information, they want huge key dimensions, huge quantities of memory, in addition a large of dealing out Energy. A key is made shortly afterward the input folder is uploaded, just like in BF encryption, and we are aware that BF uses the symmetric key encryption approach, where a sole key is available for both encryption and decryption. In order to prevent the user from realizing that the input folderread by third person, the input folder can simplydecrypt and then encrypted again if the sole key is known by the someone else. BF is one of the safer algorithms, yet knowing the one key can make it less secure. ECC, apart from, employs asymmetric encryption, it requires two keys—the public and private keys, respectively—for encryption and decryption. Since, it has a better more of protection, creating it hardintended for attacker to simultaneously decrypt together keys.

The key selling point of ECC is additional compact key length. Asrelated to some other algorithms, ECC be able to offer the equalsafety lby small key length. The development of a system that offers data security via the cloud at a lower computational cost and faster encryption/decryption procedure is necessary. In order to use the qualities of both methods, we mix them in our suggested model.

III. PROPOSEDSYSTEM

Using ECC and BF, the most advanced and efficient cryptographic approach for cloud storage is produced. Single BF requires a larger key size, which makes it slightly slower than the hybrid (ECC-BF) technique. The hybrid approach, however, provides for a smaller key size and a faster security mechanism for protecting the data. Because the small key size of ECC is its main distinguishing feature, performance is increased when BF uses ECC for encryption [31].

Using standards for encryption and decryption keys, ECC decreases key size and creates protected key systems. ECC is the ideal technique to use in conjunction with BF to encrypt the data and protect it from unauthorised use. Once the key size has been decided upon, data encryption and decryption will result in cypher text. The key that is generated by ECC is used by BF. The combined impacts of ECC and BF are sufficient for the offered method at cloud storage to obtain

the protected system. Secure data storage size can be reduced in this way. Figure 5 below displays the block diagram of the suggested algorithm.

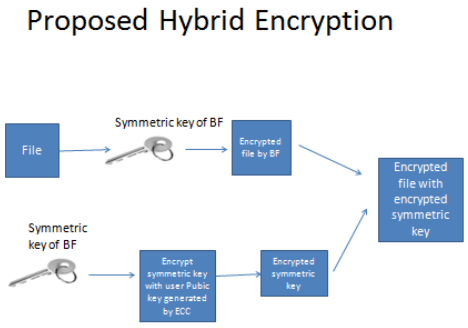

Fig:1 Hybrid Encryption

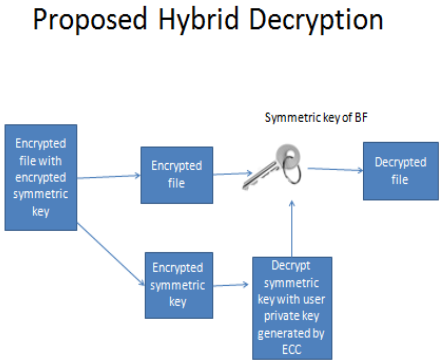

Fig:2 Hybrid Decryption

It is clear from the aforementioned graphic that BF and ECC collaborate to safely protect data that is kept in the cloud. The innovative nature of the suggested solution is illustrated by the new proposed diagram, which depicts secure user data transmission to a server followed by storage that is further secured by encrypted data. Further, computational time and expense can be utilised to measure uniqueness.

A technique for preventing assaults is, however, as follows: In the suggested solution, for instance, the input file is turned into encrypted text with the use of BF encryption after the user uploads the file, ensuring that the content is completely encrypted. This is helpful if a hacker intends to attack from the user's perspective in order to steal the user's personal data or for some other reason. As a result, even if an attacker is successful in their attack and takes down the user-uploaded file, the information is meaningless because it was already encrypted when it was submitted. Similar to this, if an attack is launched from the other end, the attacker won't be able to decrypt the encrypted file, safeguarding the data from hacks.

A. Public Key Generation Using ECC

I. Choose some m number as the prime number.

Step II. Choose some integer as m (b), where m (b) m, to create public key.

Step III: Compute a point on curve as H.

where H exceeds m

Step Four. Q = m (b) H is the formula used to calculate the open key length.

Return the calculated public key P in step five.

B. Using the BlowFish for Encryption and Decryption.

Make input file as a first phase.

Step II: Next, include the open key madethrough ECC.

In Step III, the input folder is BF-encrypted using the ECC-generated public key.

After being encrypted using BF in Step III, the file is encrypted in Step IV and taken into the server.

Step V. File then translated by means of the open key provided by ECC once it has been uploaded and downloaded at the server, allowing the original file to be unencrypted.

Step six. The combined effects of ECC and BF are what determine how well the system performs, including the strengthening of security services provided by the cloud server and the optimization of storage space.

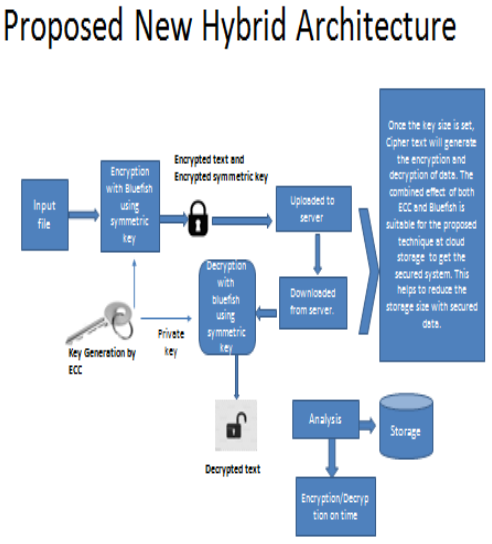

Fig 3: Proposed Hybrid Architecture

IV. EQUIPMENT AND PROCEDURES

Details regarding the materials we used to carry out our research work are provided in this part. Here are some general information regarding the implementation.

Library	Python Simple CV
Image	PNG
Graph	EXCEL
Software Simulation	ifogsim
Software	VS code editor
Language	Python

Table 1 : Implementation Environment

V. ENVIRONMENT

We implemented our suggested algorithm in Python and tested the effectiveness and originality of our suggested method. To read the photos, we used Python's Simple CV package. Our code utilized the Portable Network Graphics picture format. The system's uniqueness and effectiveness were improved in a novel way by the combination of ECC in addition BF, which produced safe connections for encryption in addition decryption of information and security, many of information stored in cloud storage.

VI. DATA VOLUME FOR THE SUGGESTED SCHEME

Because images typically take longer to process than text data, we used three distinct image datasets to compare our proposed method with other current schemes. We were interested in both the computational cost and the processing time for both the encryption and decryption of the photos. The purpose of using various image sizes is to compare performance across various contexts. The picture datasets used in the studies have a combined size of 3233, 4830, and 6308. The comparison of encryption times for various techniques is shown in Figure 7.

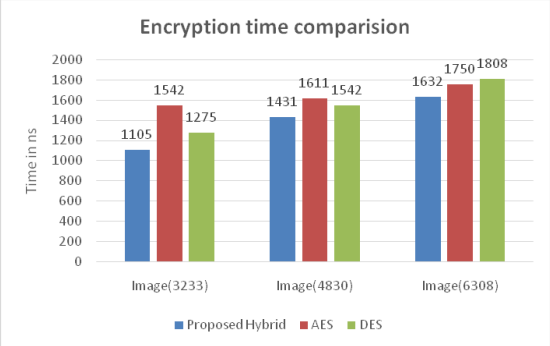

Fig 4: Comparison of Encryption time

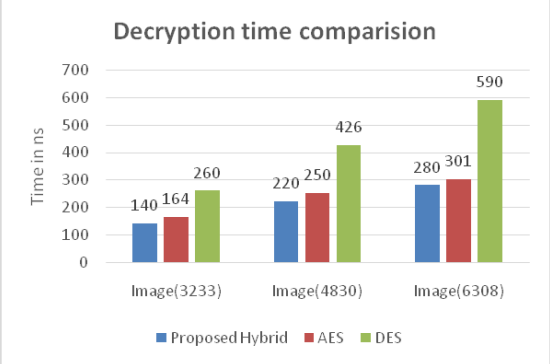

Fig 5: Comparison of Decryption time

Figure 8 compares the lengths of time required to decrypt data using various encryption techniques. The suggested algorithm is the hybrid algorithm. Results are notable in that the hybrid ECC-BF approach's reduced key size makes data decryption faster than that of the other approaches. Additionally, the hybrid ECC-BF algorithm combines the best features of both algorithms to increase security by boosting system complexity and resistance to assaults. It is also obvious that the suggested hybrid method decrypts data significantly faster than other algorithms. Our computing cost is also decreased, which is quite effective as the decryption time is decreased. Therefore, compared to other strategies, ours is more effective.

VII. Evaluation

The comparisons then findings are run on computer that met the succeeding requirements: Microsoft Windows 11, Intel i6-8650U running at 1.82GHz, 4 Cores and 8 Logical Processors. On a Python platform, the suggested approach was tried.

1. Encryption Time

For final confirmation, we also contrasted the encryption and decryption times for our proposed hybrid algorithm with various key sizes and with the already-in-use methods (BF, DES, and Blowfish). Unlike keys, including 64, 128, 192, and 256 bits, were used for tests.

Keysizes	Hybrid	BF	DES
64	2.30	3.32	3.79
128	2.37	3.49	4.0
192	2.44	3.38	4.01
256	2.50	3.50	4.24

Table 2: Encryption time calculated using different key sizes

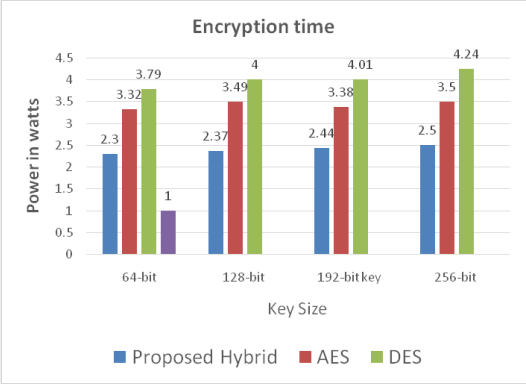

Fig 6: Comparison of Proposed Hybrid Encryption with different key sizes

2. Decryption

Keysizes	Hybrid	BF	DES
64	1.54	2.59	3.01
128	1.67	2.72	3.20
192	1.78	2.83	3.28
256	2	3.1	3.48

Table 3: Decryption time calculated using different key sizes

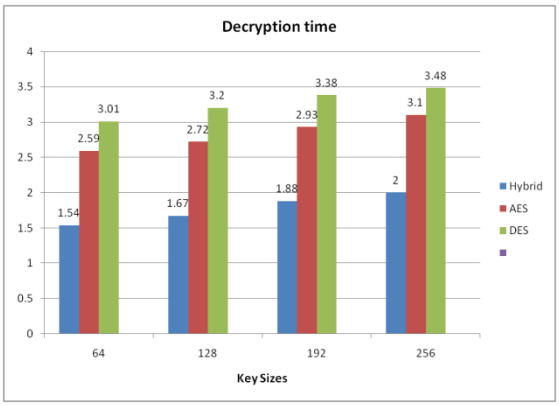

Fig 7: Comparison of Proposed Hybrid Decryption with different key sizes

3. Power consumption in encryption

Power Consumption	Proposed Hybrid	BF	DES
64-bit key size	2.3	3.32	3.79
128-bit key size	2.37	3.49	4
192-bit key size	2.44	3.38	4
256-bit key size	2.5	3.5	4.24

Table4: Comparison of Power Consumption

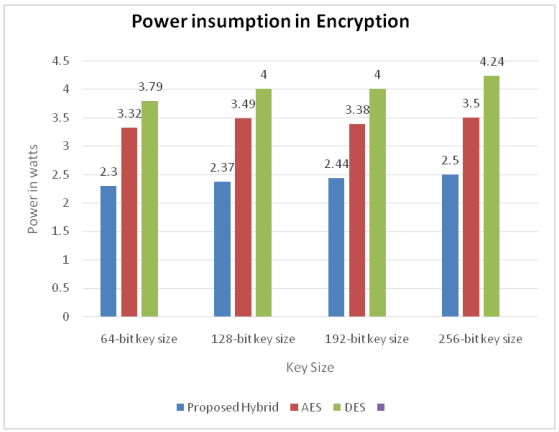

Fig 8: Comparison of power consumption

4. Power consumption in decryption

Power Consumption	Proposed Hybrid	BF	DES	Blowfish
64-bit key size	1.54	2.59	3	3.79
128-bit key size	1.67	2.72	3.2	3.84
192-bit key size	1.78	2.83	3.28	4
256-bit key size	2	3	3.48	4.15

Table 5: Comparison of Power Consumption

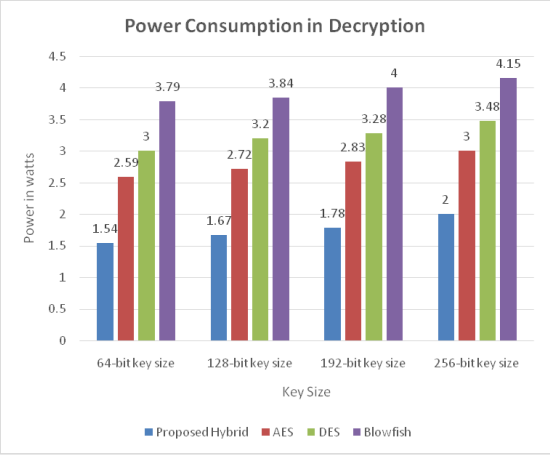

Fig 9: Comparison of power consumption

IX. CONCLUSION

No matter how tech-savvy the user is, IT-related services like cloud computing offer effective services. Regardless of the user's location, data can be easily saved, maintained, upgraded, and retrieved through a cloud interface offered by independent cloud service providers via the Internet. Numerous features are available to users of cloud services. According to the type of cloud service, users are those who use the services. In addition to being advantageous for so many customers to use information to wherever, facilities are offered at minimal cost. Since no need to carry the gadgetby him, cloud facilitiesbe made available on any system.

The low data security of cloud services is a disadvantage, albeit it can be fixed by using specific techniques, which must be guarded. ECC is utilized to simplify the activities, precisely due to production of key length. The enhancement is substantially superior compared tofurther cryptographic approaches because of its lesser key length. The enhancement then safety of information can be greatly improved when BF is used in conjunction with ECC. However, in order to develop the idea of cloud computing through cryptographic methods in the future, much security is still required. Forth coming investigation can be strengthened through improving the fusion method's safety. Towards increase computer efficiency and output, multiple safety levels will be presented.

References:

1. 1. Shukla, D.K.; Dwivedi, V.K.; Trivedi, M.C. Encryption algorithm in cloud computing. *Mater. Today Proc.* **2020**, *37*, 1869–1875.
2. Yahia, H.S.; Zeebaree, S.R.M.; Sadeeq, M.A.M.; Salim, N.O.M.; Kak, S.F.; Al-Zebari, A.; Salih, A.A.; Hussein, H.A. Comprehensive survey for cloud computing based nature-inspired algorithms optimization scheduling. *Asian J. Res. Comput. Sci.* **2021**, 1–16.
3. Qazi, R.; Khan, I.A. Data security in cloud computing using elliptic curve cryptography. *Int. J. Comput. Commun. Netw.* **2019**, *1*, 46–52.
4. Chen, Y.; Liu, H.; Wang, B.; Sonompil, B.; Ping, Y.; Zhang, Z. A threshold hybrid encryption method for integrity audit without trusted center. *J. Cloud Comput.* **2021**, *10*, 3.
5. Agrahari, V. Data security in cloud computing using cryptography algorithms. *Int. J. Sci. Dev. Res.* **2020**. Available online: www.ijsdr.org (accessed on 22 October 2021).
6. Abdullahi Ibrahim, A.; Cheruiyot, W.; Kimwele, M.W. Data security in cloud computing with elliptic curve cryptography core. *Int. J. Comput.* **2017**, *26*, 1–14. Available online: http://ijcjournal.org/ (accessed on 22 October 2021).
7. Manaa, M.E.; Hadi, Z.G. Scalable and robust cryptography approach using cloud

computing. *J. Discret. Math. Sci. Cryptogr.* **2020**, *23*, 1439–1445.
8. Madhavi, G.; Samatha, J. Secure data storage and access of data in cloud using Elliptic curve cryptography. *IEEE J.* **2020**, *11*. Available online: **www.jespublication.com** (accessed on 22 October 2021).
9. Sridharan, S.; Arokiasamy, A. Effective secure data storage in cloud by using ecc algorithm. *Middle-East J. Sci. Res.* **2017**, *25*, 117–127.
10. Selvam, J.M.; Srivaramangai, P. Time complexity analysis of cloud authentications and data security: Polynomial based hashing and elliptic curve cryptography. *Int. J. Anal. Exp. Modal Anal.* **2020**, *12*, 850–860.
11. Manaa, M.E. Data encryption scheme for large data scale in cloud computing. *J. Telecommun. Electron. Comput. Eng.* **2017**, *9*, 1–5. Available online: **https://jtec.utem.edu.my/jtec/article/view/2759** (accessed on 22 October 2021).
12. Astuti, N.R.D.P.; Aribowo, E.; Saputra, E. Data security improvements on cloud computing using cryptography and steganography. *IOP Conf. Series Mater. Sci. Eng.* **2020**, *821*, 012041.
13. Awad, W.S. A framework for improving information security using cloud computing. *Int. J. Adv. Res. Eng. Technol.* **2020**, *11*, 264–280.
14. Kumar, V.; Ahmad, M.; Kumari, A. A secure elliptic curve cryptography based mutual authentication protocol for cloud-assisted TMIS. *Telemat. Inform.* **2018**, *38*, 100–117.]
15. Singla, S.; Singh, J. Cloud computing security using encryption technique. *Int. J. Adv. Res. Comput. Eng. Technol.* **2013**, *2*, 673.
16. Almorsy, M.; Grundy, J.; Müller, I. An analysis of the cloud computing security problem. *arXiv* **2016**, arXiv:1609.01107.
17. Jena, O.P.; Tripathy, A.; Swagatam, S.; Rath, S. Dual encryption model for preserving privacy in cloud computing. *Adv. Math. Sci. J.* **2020**, *9*, 6667–6678.
18. Arockia, P.; Dharani, N.; Aiswarya, R.; Shailesh, P. Cloud data security using elliptic curve cryptography. *Int. Res. J. Eng. Technol.* **2017**, *4*, 32–36.
19. Li, Y.; Gai, K.; Qiu, L.; Qiu, M.; Zhao, H. Intelligent cryptography approach for secure distributed big data storage in cloud computing. *Inf. Sci.* **2017**, *387*, 103–115.
20. Saeed, Z.R.; Ayop, Z.; Azma, N.; RizuanBaharon, M. Improved cloud storage security of using three layers cryptography algorithms. *Int. J. Comput. Sci. Inf. Secur.* **2018**, *16*, 34–39.
21. Al-Dhuraibi, Y.; Paraiso, F.; Djarallah, N.; Merle, P. Elasticity in cloud computing: State of the art and research challenges. *IEEE Trans. Serv. Comput.* **2017**, *11*, 430–447.
22. Hosam, O.; Ahmad, M.H. Hybrid design for cloud data security using combination of AES, ECC and LSB steganography. *Int. J. Comput. Sci. Eng.* **2019**, *19*, 153.
23. Shantha, A.; Renita, J.; Edna, E.N. Analysis and implementation of ECC algorithm in lightweight device. In Proceedings of the 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 4–6 April 2019; pp. 305–309.
24. Varghese, S.; Vigila, S.M.C. A varied approach to attribute based access model for secure storage in cloud. In Proceedings of the 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 17–18 March 2017; pp. 1–4.
25. Hodowu, D.K.M.; Korda, D.R.; Ansong, E.D. An enhancement of data security in cloud computing with an implementation of a two-level cryptographic technique, using AES and ECC algorithm. *Int. J. Eng. Res. Technol.* **2020**, *9*, 639–650.
26. Lee, B.-H.; Dewi, E.K.; Wajdi, M.F. Data security in cloud computing using AES under HEROKU cloud. In Proceedings of the 2018 27th Wireless and Optical Communication Conference (WOCC), Hualien, Taiwan, 30 April–1 May 2018; pp. 1–5.
27. Zhu, Y.; Fu, A.; Yu, S.; Yu, Y.; Li, S.; Chen, Z. New algorithm for secure outsourcing of modular exponentiation with optimal checkability based on single untrusted server. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
28. Bhardwaj, K.; Chaudhary, S. Implementation of elliptic curve cryptography in 'C'. *Int. J. Emerg. Technol.* **2012**, *3*, 38–51.
29. Ogiela, U. Cognitive cryptography for data security in cloud computing. *Concurr. Comput. Pr. Exp.* **2019**, *32*, e5557.
30. Sood, S.K. A combined approach to ensure data security in cloud computing. *J. Netw. Comput. Appl.* **2012**, *35*, 1831–1838.
31. Mendonca, S.N. Data security in cloud using AES. *Int. J. Eng. Res. Technol.* **2018**, *7*.
32. Suresha, R.G. Enhancing security in cloud storage using ecc algorithm. *Int. J. Sci. Res.* **2013**, 2–8. Available online: **https://www.ijsr.net/archive/v2i7/MDIwMTM3NA==.pdf** (accessed on 22 October 2021).
33. Abbas, S.; Maryoosh, A.A. Improving data storage security in cloud computing using elliptic curve cryptography. *IOSR J. Comput. Eng.* **2015**, *17*, 48–53.

34. Barati, M.; Aujla, G.S.; Llanos, J.T.; Duodu, K.A.; Rana, O.F.; Carr, M.; Rajan, R. Privacy-Aware cloud auditing for gdpr compliance verification in online healthcare. *IEEE Trans. Ind. Inform.* **2021**, *1*.
35. Mahto, D.; Yadav, D.K. RSA and ECC: A comparative analysis. *Int. J. Appl. Eng. Res.* **2017**, *12*, 9053–9061.
36. Vidakovic, D.; Parezanovic, D. Generating keys in elliptic curve cryptosystems. *arXiv* **2013**, arXiv:1309.0245.