electronics

MDPI

Article

Hybrid AES-ECC Model for the Security of Data over Cloud Storage

Saba Rehman [1], Nida Talat Bajwa [1], Munam Ali Shah [1,*], Ahmad O. Aseeri [2] and Adeel Anjum [1]

[1] Department of Computer Sciences, COMSATS University, Islamabad 45550, Pakistan; sabarehmanciit@gmail.com (S.R.); nidatalat01@gmail.com (N.T.B.); adeel.anjum@comsats.edu.pk (A.A.)
[2] Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia; a.aseeri@psau.edu.sa
* Correspondence: mshah@comsats.edu.pk

Abstract: A cloud computing environment provides a cost-effective way for the end user to store and access private data over remote storage using some Internet connection. The user has access to the data anywhere and at any time. However, the data over the cloud do not remain secure all the time. Since the data are accessible to the end user only by using the interference of a third party, it is prone to breach of authentication and integrity of the data. Moreover, cloud computing allows simultaneous users to access and retrieve their data online over different Internet connections, which leads to the exposure, leakage, and loss of a user's sensitive data in different locations. Many algorithms and protocols have been developed to maintain the security and integrity of the data using cryptographic algorithms such as the Elliptic Curve Cryptography (ECC). This paper proposes a secure and optimized scheme for sharing data while maintaining data security and integrity over the cloud. The proposed system mainly functions by combining the ECC and the Advanced Encryption Standard (AES) method to ensure authentication and data integrity. The experimental results show that the proposed approach is efficient and yields better results when compared with existing approaches.

Keywords: cloud computing; data security; authentication; data integrity; Elliptic Curve Cryptography (ECC); Advanced Encryption Standard (AES)

Citation: Rehman, S.; Talat Bajwa, N.; Shah, M.A.; Aseeri, A.O.; Anjum, A. Hybrid AES-ECC Model for the Security of Data over Cloud Storage. *Electronics* **2021**, *10*, 2673. https://doi.org/10.3390/electronics10212673

Academic Editors: Amir Mosavi and Manohar Das

Received: 4 September 2021
Accepted: 26 October 2021
Published: 31 October 2021

1. Introduction

Cloud computing has been one of many distinguished computing models that provides numerous services irrespective of the location and medium, easily accessible at any location, such as unlimited database storage, networks and communications, and others. Its attractive features have led to an ever-increasing reliance on the cloud, leading to a massive volume of data and raising privacy and security concerns. Significant drawbacks of cloud services, especially data security and data breaching, could be originated by cloud users either intentionally or unintentionally. Therefore, restrictions on data access should be applied to unauthenticated and unauthorized sources. Devices can also be a source of data breaching since users may be allowed to reuse the APIs and data, causing data loss. Hence, cryptographic methods are primarily employed to secure the data over the cloud by applying encryption/decryption techniques with the help of different keys. The two types of techniques that are used for the encryption of data with the help of keys are [1]:

- Asymmetric key encryption of data and
- Symmetric key encryption of data.

Asymmetric key encryption is also known as public-key cryptography. It contains a pair of keys, i.e., public and private keys for encryption and decryption of the message, respectively. However, symmetric key encryption is also used for the security of the data, where data are encrypted with the help of a single private key and then decrypted. The

private key is used to encrypt the message and avoid the various activity of the hackers. It has been identified that the difficulty in adopting the symmetric cryptographic algorithms stems from the key size which needs to be large enough to ensure proper security. List of abbreviations used in this paper are shown in Table 1.

Table 1. Acronym Table.

Abbreviation	Explanation
AES	Advanced Encryption Standard
ECC	Elliptic Curve Cryptography
RSA	Rivest–Shamir–Adleman
DES	Data Encryption Standard
PHECC	Polynomial-based hashing elliptic curve cryptography
NIST	National Institute of Standards and Technology
EAP-CHAP	Extensible Authentication Protocol—Challenge Handshake Authentication Protocol
GDLP	Generalized Discrete Logarithm Problem
API	Application Programming Interface
CSP	Cryptographic Service Provider

Advanced Encryption Standard (AES) is based on the symmetric key encryption algorithm [2], which was developed for the replacement of Data Encryption Standard (DES) and is much faster than the DES because it contains 128-bit key size whereas DES contains 64-bit key size. Accordingly, Elliptic Curve Cryptography (ECC) is based on asymmetric key encryption [3], and it outperforms other cryptographic methods with its primary feature of providing higher security with smaller key size compared to other existing schemes like the Rivest–Shamir–Adleman (RSA) algorithm [2], which is based on asymmetric cryptography algorithm leading to smaller latency and lesser computational/hardware complexities.

Using AES along with ECC was proposed by Chen et al. [4], who used AES with ECC for increasing the system security. However, they also used Shamir secret sharing key for the transfer of data. In this paper, we proposed a hybrid model of AES and ECC for the security of data over cloud storage without involving a third party. The proposed hybrid model (AES-ECC) is used to maintain the security of the system over cloud storage efficiently. The main reason for using this hybrid approach is to reduce the key size of the data while maintaining the security of the system in less time. Although many authentication methods exist, computational cost and time is not reduced.

Cloud computing provides efficient services for the storage of data on the remote server. In general, it provides three services which are used at different levels and different purposes [3]:

- SAAS (Software as a service) $\geq$ End Users
- PAAS (Platform as a service) $\geq$ Application Developers
- IAAS (Infrastructure as a service) $\geq$ Network Architects

However, different cloud systems are classified in different deployment models which are as follows [5]:

- Public cloud service (For all)
- Private cloud service (For the encryption purpose and security)
- Hybrid cloud service (Both as a mixture of public and private cloud systems)
- Community cloud service (Specifically for the communities)

1.1. Cloud Computing Characteristics

Cloud computing is a promising technology which helps organizations expand and safely transfer data from physical locations to the 'cloud' server that can be accessed from anywhere at any time. Different characteristics of cloud computing provides services to the users irrespective of their type but depend on their work. Some of the characteristics are [6]:

1.1.1. Network Access

The network is accessible easily to every user that there will be no difficulty for the users to access their data at any place or any machine.

1.1.2. Expeditious Elasticity

User friendly and elastic in manner. It allows cloud storage as per the need of every user.

1.1.3. Self Service on Demand

Self-services are provided by cloud computing as many of them are provided without the consent of the service provider.

1.1.4. Pooling Resources

Redistribution and allocation of resources are dependent on the user's need in a unique manner

1.1.5. Managed and Measured Services

Services are controlled by the service providers like the arrangement of data and security management of the cloud storage for each user. Figure 1 represents the services mainly provided by cloud computing.

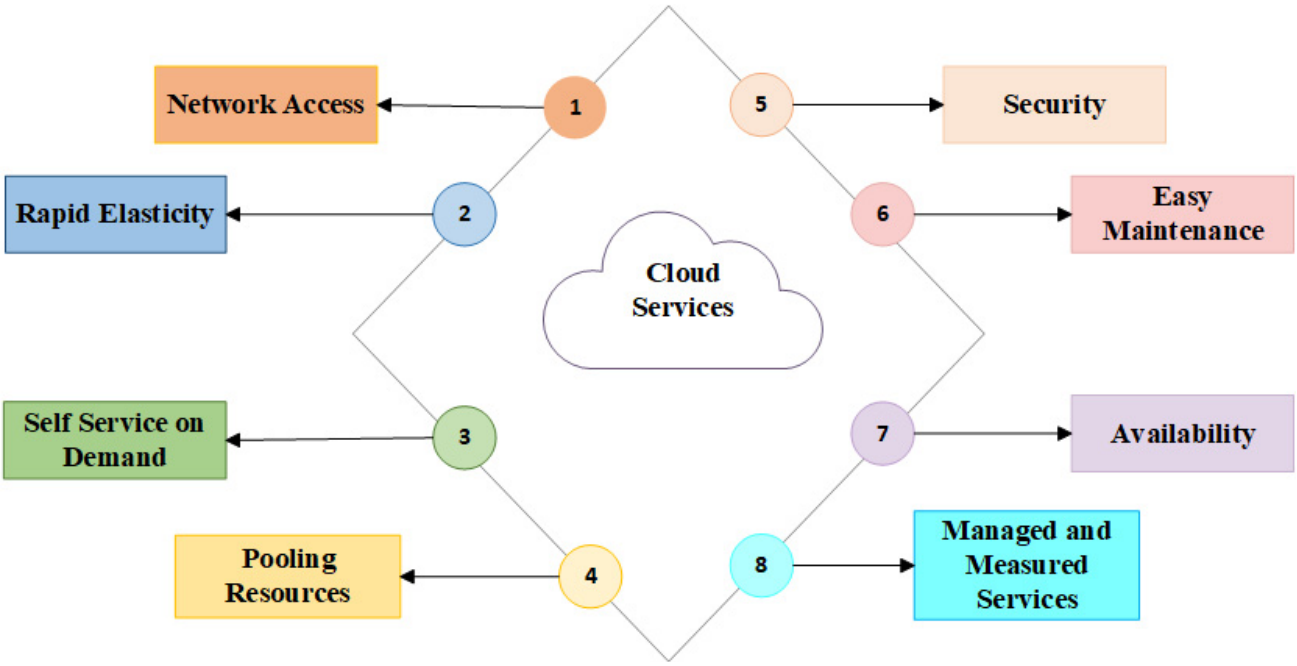

Figure 1. Cloud services.

1.2. Cryptographic Cloud Services

Recent enhancements in cloud storage have provided different services to the users to get the encryption and decryption of data done in an experienced manner without the involvement of a third party. This enhances the system for increasing the security of the system as well as the efficiency of the storage for secure accessibility and fast retrieval of data. The sharing of cloud resources by different types of users can be easily done through this service and enhance the capability of the system by securing more storage after the implementation of the cryptographic techniques [7].

Figure 2 presents the different forms of the cloud storage services. It can be seen that the data between sender and receiver are being encrypted and decrypted and stored at the cloud server. It also represents the secure transmission of data over the cloud storage.

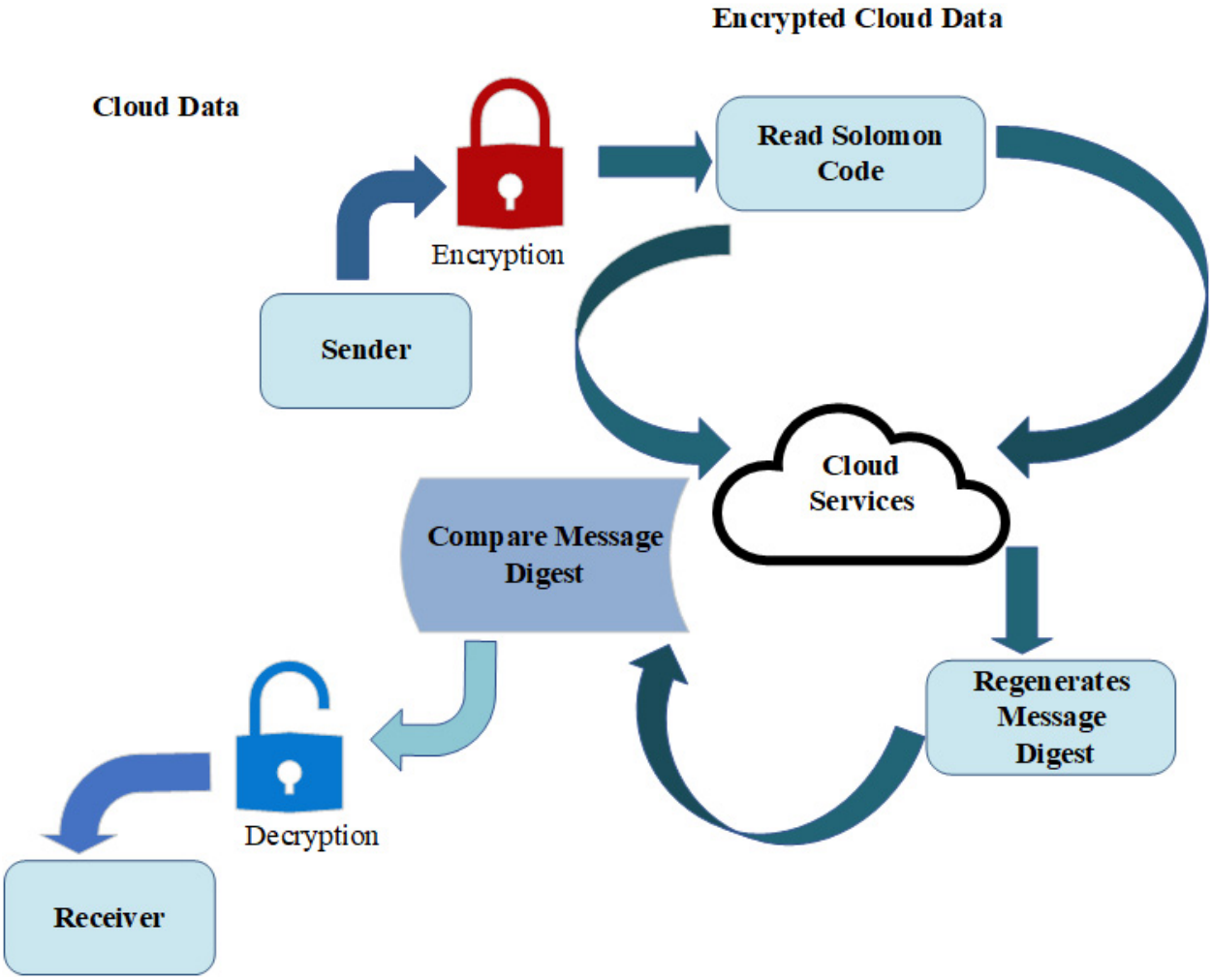

Figure 2. Cloud storage services.

1.3. Elliptic Curve Cryptography (Cloud Computing Service)

Efficient types of cloud services are provided in the form ECC which is used for the encryption and decryption of data. These services use asymmetric encryption and decryption of data. It is more efficient than others as it uses the RSA encryption and decryption of data; the key size is lower compared to the symmetric encryption and decryption of data. For example, if the size of the data for the key is 1020 bits, it will be reduced to 163 bits by using the RSA scheme in ECC. Moreover, ECC is more appropriate for network access using smartphones. Many hackers are in a hurry to breach the user's personal information or some data to blackmail them. That is why ECC is specifically made for users who are accessing their data with least secured devices that can be easily hacked [8].

Figure 3 represents the user and data stored on cloud server; it shows the process of how a user request of data has been processed at the server and how the data are being accessed securely.

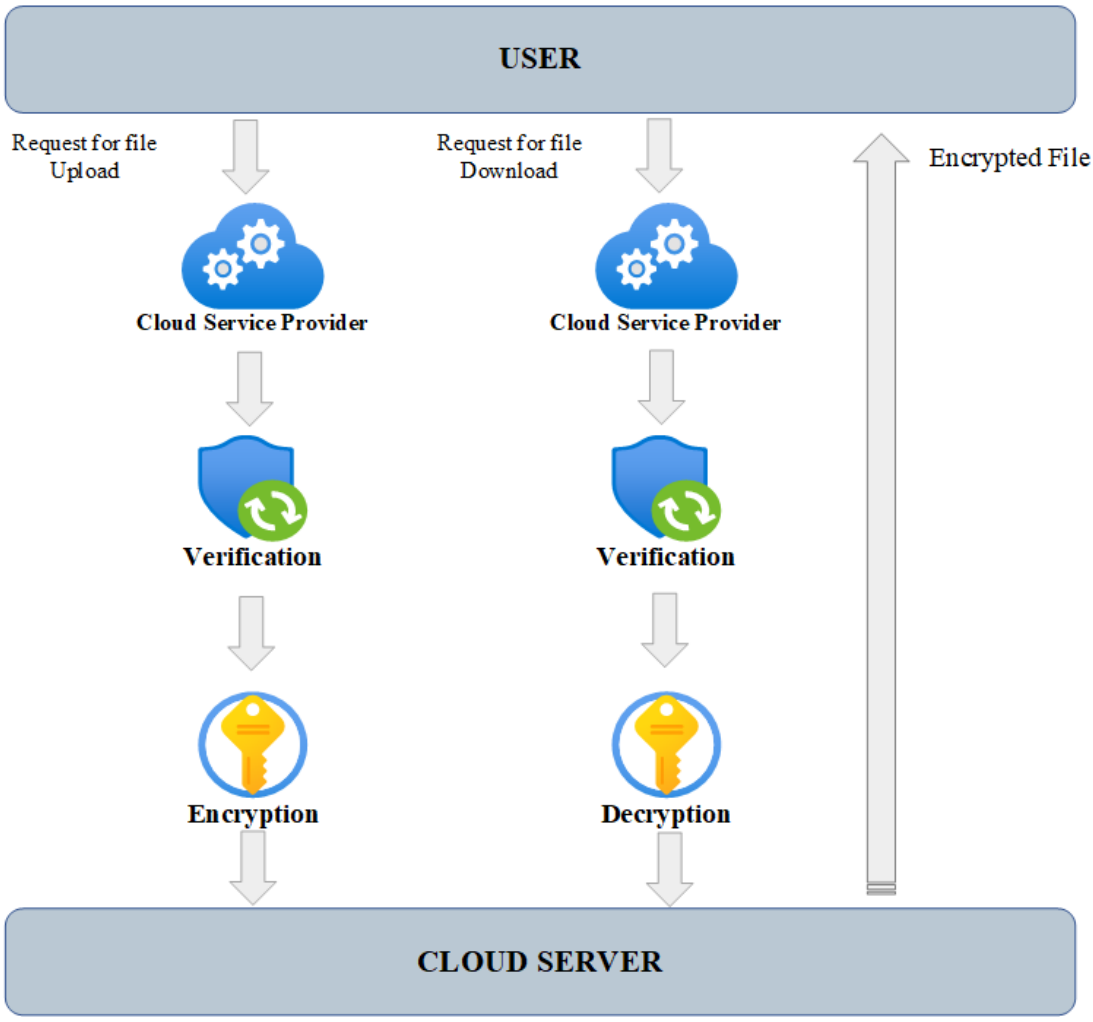

Figure 3. Cloud storage server.

1.4. Problem Statement

Currently, the security of data is a major issue which can be compromised in different ways using either external or internal means. To protect the transmission of data over the Internet, different encryption techniques are used. The problem with these techniques is that they need large key size, large memory size, and require a lot of computation power to protect the data. As in AES encryption, a key is generated soon after the input file is uploaded, and we know AES uses symmetric key encryption method in which a single key is used for encryption as well as decryption. Hence, if the single key is known by the third party, then the input file will easily be decrypted and again encrypted so that the user does not know that the input file has been read by someone else. Although AES is one of the secure algorithms, its security can be compromised by knowing the single key. On the other hand, ECC uses asymmetric key encryption in which there are two keys for encryption and decryption, public and private key, respectively. This is the reason that its security level is higher, and it is not easy for hackers to crack both keys at once. ECC is also mainly known for its smaller key size. ECC can provide the same security level as compared to other algorithms in a smaller key size. There is a need to develop a system which provides the security of the data over the cloud with less computational cost and less time for the encryption/decryption process. We combine the properties of both algorithms and utilize them in our proposed model.

1.5. Contributions

The following are the main contributions of this paper:

- ❖ We propose a hybrid model by combining two algorithms AES and ECC, in which key generation of AES is done with the help of ECC. In simple words, we are not using the key generated by the AES algorithm; ECC is used for generating the key so that its key size is reduced.
- ❖ As in symmetric/asymmetric encryption, a public key or private key is used for encryption and decryption of data. Thus, this process needs a large key size and requires a lot of computational power. The proposed hybrid algorithm (AES-ECC) is used to enhance the security of system in less time by solving the problem of key size and it helps to reduce the computational power for memory optimization.
- ❖ We also present an algorithm for our proposed framework which describes how the public key is generated using ECC algorithm and how encryption/decryption is done using AES.

The remaining sections of this paper are organized as follows. Section 2 provides the related studies. In Section 3, the research methodology is explained in detail. The proposed hybrid AES-ECC model is presented in Sections 4 and 5 is about experimental results and discussions. In Section 6, the conclusion of the work is presented.

2. Related Work

Cloud storage is gaining popularity day by day due to its synchronous resource sharing among all users. Data owners prefer cloud storage over other services due to its access-all time nature. For this purpose, data integrity and data preservation should be verified in order to increase security of the system.

In ref. [1], AES along with ECC is proposed for increasing security of the system. Shamir secret sharing is used for distributing and managing the system without the trusted center. Although the proposed combined approach increases system security, it still takes huge computational cost as well as taking much time.

AES, DES, and Blowfish methods are used along with the proposed method which uses the relatable algorithms for secure cloud services [2]. These algorithms provide efficiency and integrity to the data storage to avoid conflict among bulk users and secure the data of each user separately. Moreover, data accessibility is expeditious and managed appropriately by the service provider. The avalanche impact of plain text and data block size is also measured by the cloud computing data services.

In paper [3] by Madhavi et al., the security strength of ECC and RSA are combined by using the data over 264 bits while 256 bits of data follow the guidelines of NIST. This working of algorithms shows that the ECC works more effectively as compared to the RSA algorithm as it provides more secure services over reduced data size and contains less storage for the accessibility of data. Experimentation can be done on different platforms of JAVA.

ECC is used for the encryption and decryption of data to provide secure and efficient services to different users [4,9]. The encryption and decryption of data are done in the layered method comprising of two sections. The first section contains the small sections for the addition of the bits for the data encryption process and reduces the size of the keys for efficient accessibility. Meanwhile, the second layer consists of a partition of the elliptical curves which are used for the encryption of data such as P_0, P_1, P_2, P_3, P_4 … P_n. These steps are used for the encryption and decryption process and these two layers provide the security of the data. In the previous techniques, data losses and security issues arise. To overcome the effect of these issues, ECC is used to secure the data and avoid the breakdown of the data for unethical reasons. In this asymmetric technique of cryptography, data security and enhancement of larger datasets can be done easily to provide security services most readily. ECC simultaneously provides the two operations to access and secure the data over cloud computing.

Polynomial-based hashing elliptical curve cryptography (PHECC) is proposed in [10], which has the implementation of polynomial-based hashing elliptical curve cryptography by using the hybrid algorithm of cloud computing service. This technique ensures the supportive following and usage of clients/users by using the service. Cloud data security can benefit from the hybrid cloud algorithm which is suitable for the current situation and provides high security. The polynomial-based hashing along with the elliptical curve following the hybrid algorithm has provided essential security measures to both the system and the users for enhanced and efficient services. The integrity of data is maintained through this technique as data are encrypted and decrypted; once it uploads to the cloud, a specific hash value is assigned using PHECC. Data are encrypted and decrypted in PHECC using the elliptical curve while the hash value is generated by polynomial hashing and hybrid algorithms. Thus, the integrity of data over cloud services for uploading and downloading is achieved.

Hybrid algorithms for RSS and ECC are used in paper [11]. Once the data are reduced, some elements which are signature are assigned to the elliptical curve authorities for the signing and digestion of the message. Sometimes, the encrypted data is used for this purpose by ECC. The encryption and encryption process are done in the same manner. Hybrid algorithms for RSS and ECC analysis are done based on their excellence.

Data encryption and decryption over the transmission are secured through different mediums [12]. The confidentiality, integrity, and integrity of the data are mentioned in that paper. The technology used for the authentication and confidentiality of the data is the Irondale encryption algorithm with the EAP-CHAP for the usage of cloud computing services over the internet.

Authentication of data is important in different latest devices such as IoT devices. High-capacity data with more confidential information is stored in these storage spaces, which is why authentication is important with these devices. Processor power must be high to perform the cryptographic operations on that device. Clouds are used by these devices to achieve the authentication of data and execution of protocols [13,14].

A model was made for the smart grid named the wide area measurement system key management proposed by Yee Wei Law et al. [15]. For the secure communication of data and authentication with different devices to implement the protocols, public key infrastructure is used. The problem can be solved by the traditional public key infrastructure also.

Storage cloud is IaaS that gives buyer associations an adequate degree of adaptability and flexibility in utilizing the virtualized climate for its storage requirements. IaaS is the essential level for different models and administrations. The other two categories of installment are factors and ward on the measure of capacity limit required and the data transmission required. In the vast majority of the cases, the first prerequisite, which is the capacity limit, has various classes of installments that are reliant upon the limit. As such, the expense of 1 GB when the entire prerequisite does not surpass 1 TB is not the same as the expense of 1 GB when the complete necessity surpasses 1 TB [16,17].

Security and protection privacy [18,19] are a vital challenge for cloud computing services. Since the CSP is a non-confided in outsider, we cannot store crude information without encryption in view of secrecy issues. In the proposed work, a solid information transmission and capacity in cloud by the help of cross breed cryptosystem is talked about. The AES and ECC are applied at the same time to enhance the trustworthiness and classification of the framework, by which we can apply both symmetric and deviated encryption to add more security to the cloud information. Accordingly, the expected model manages a well-organized AES- and ECC-based encryption strategy, which is safer and has more computational power.

Paper [20,21] introduced the capacities of cryptography in terms of giving security in distributed storage. This was done by exploring the normal cryptography techniques including AES, ECC, and RSA. In any case, with respect to the variations in the exhibition of these procedures, this study resolved the issue of recognizing an effective and secure encryption technique. Some encryption methods can ensure security, yet they take quite a

while for encryption and decryption. In the other way around, different strategies might give an efficient encryption, yet they experience the ill effects of the need for security.

The author [22] introduced a configuration that depends on different key parts that allow the security model to put away and share touchy information utilizing public cloud innovation. AES with a 256-bit encryption decoding motor is utilized to scramble the information that will be transferred on the cloud. AES has conveyed incredible outcomes for scrambling mass information at high paces. Steganography measures are utilized for information security. Steganography stores delicate information inside other information, in this way expanding the information size to be put away on cloud. These result in increment transfer speed and cloud diminish capacity use, which is not practical.

8-bit ECC processor possesses just 11 cuts used in [23,24]. A review on the duplication strategies were done, from which Karatsuba, Stall, and Montgomery's particular increase techniques were observed to be effective. The investigation of the three augmentation techniques was performed and among the three increase strategies, Karatsuba duplication was picked as a space-effective duplication strategy. The Karatsuba augmentation strategy possesses a smaller number of cuts in contrast with other strategies. It will be helpful in decreasing for higher request bits since lower request bits as of now involve a smaller number of cuts.

Paper [25–27] introduced a two-level cryptographic procedure and a model for the improvement of information security in cloud processing. The model utilizes both symmetric and uneven encryption calculation (AES and ECC) to improve the security of information against intruders, denying them from approaching the genuine information, thus empowering privacy, respectability of the information, and time taken to perform cryptographic tasks, and further developing the trust level of clients in cloud computing and speeding up the utilization of more modest keys of ECC in the cryptographic interaction.

Table 2 represents the comparative analysis of related work in detail of papers [1–13].

Table 2. Comparative analysis of related work.

Refs.	Tools/Technology	Approach	Limitations
[1]	JAVA in Eclipse	AES-ECC along with Shamir secret key	CSP does not contain any information about the private key of any user.
[2]	MATLAB	Comparison of different existing algorithms	Large key size required for encryption.
[3]	Euler's Phi function	Two-layered approach	ECC with GDLP in which group operations cannot be limited to multiplication.
[5]	JAVA	Polynomial-based hashing elliptical curve cryptography	PHECC increases the size of the encrypted messages because they have to encrypt the hash values as well.
[6]	PYTHON	Irondale encryption algorithm with EAP-CHAP	EAP-CHAP consumes high computational power.
[7]	MATLAB	Smart grid model named WAM (Wide Area Measurement)	Smart grid needs different devices to secure communication of data in local area measurements.
[8]	XSS	Model-based approach	Smaller security layer around the security wrapper.
[9]	iFogSim	Point Multiplication in Hybrid approach	Less data security in cloud computing.
[10]	JAVA in Netbeans	Hybrid algorithm and experimental time evaluation	Large amount of data on cloud is at risk because the user authentication process in cloud storage is very slow, so the bulk of users need different mediums to process their authentication.
[11]	Python	Holistic Security Model (AES 256-bit)	Data are vulnerable to sniffing and prying eyes during transmission from user to cloud storage. Attackers can compromise cloud segmentation and gain access to victim's data.
[12]	Verilog Programming Language	8-bit Elliptic Curve Crypto-processor	ECC-based lightweight devices are less secured because Karatsuba multiplier is without bi-linear pairing. It cannot uphold the decryption process and it has less scalability.
[13]	AESCrypt and OpenSSL in the Kali Linux	AESCrypt Two-layered approach	Limited field available for ECC operation.

Different levels of security and threats are discussed by different authors [5] based on the services which are discussed below in Table 3.

Table 3. Levels of security threats in a cloud environment.

Threats/Dangers	Description	Restitution
Breaching of data	Performance of different vulnerable attacks on the sensitive data to use it for the unauthorized purpose or stealing of sensitive information for blackmailing.	Well managed and monitored infrastructure, protection policies implementation in the system for securing data by setting protocols.
Management and access credentials	Less protection or negligibility of a person gives access to the unauthorized user.	Two-factor authentication can be used to minimize the unauthorized access of malicious people.
Insecure APIs and interfaces	Many applications are used which are insecure and use of these APIs in the system gives access to the unauthorized person and may infect the sensitive information by applying and reusing IDs and passwords.	Methods of encryption can help to overcome this issue by applying strong API frameworks and security models for limited access.
Vulnerability of the system	Sensitive information can be hacked through different loopholes and disrupts the system by accessing the sensitive data.	Limit the access of the people who are not part of the team or the system like customers and other people. Integrating the system credibility for enhancement.
Malicious employees	Some such people are known as threat employees and may infect the system for unauthorized use.	Monitoring of the activities which are suspicious on a daily basis must be done. Proper checking and balancing of data can help to reduce the threat and also catch malicious insiders red-handed with monitoring.
Loss of data	Crashing of the system can cause data loss or if there are no backups of data, sensitive information can be lost or accessed by unauthorized people.	Control security checks must be done as well as restoration of data along with secure backups can help to avoid data loss.

Different levels of security, attacks and their impact on system can be seen in Table 4.

Table 4. Security issues and system attacks [16].

Issues of Security	Attacks by Vendors	System Attacks	Impacts on System
At Virtualization Level	Vulnerability of storage Engineering by social behavior	VM Escape, DDoS, DM attacks	Modification of the system, interruption by system or outsiders
At Application Level	Management of session Authentication policy	Injections of SQL, Scripting by cross-site	Hijacking of the system, confidentiality of the system
At Network Level	Misconfiguration of firewalls	Reuse of IP, sniffing, and DNS system	Exposure of network and flaws of traffic
At Physical Level	Power control losses	System malware and phishing	Theft and hardware modification

3. Research Methodology

In this section, we provide the research design that has been followed by the paper. The research design for the proposed method is also discussed in the next section. Figure 4 shows the conventional research methodology followed in general, from reviewing existing schemes to proving the proposed one.

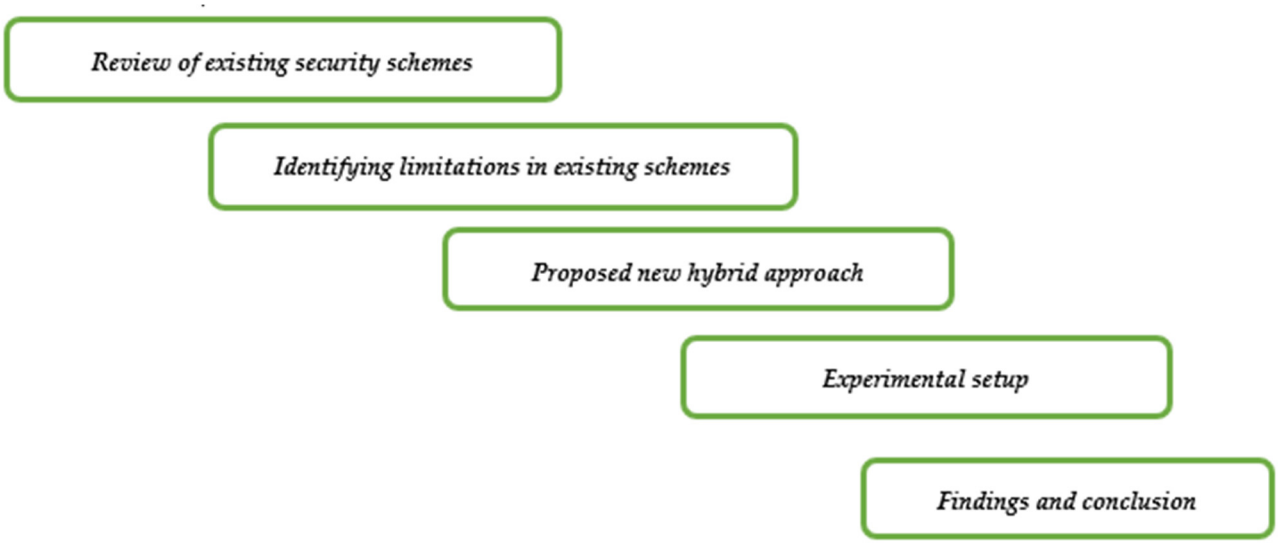

Figure 4. Methodology used in the paper.

In Figure 4, we can see that paper follows the conventional research methodology flow. Firstly, we studied already existing schemes in the literature. After studying, we identified several limitations in these existing schemes. Mainly, we found that computation overhead cost for existing schemes is larger as is the time required to compute them. Several other limitations were found; however, we solve only a few of the main limitations by proposing our new hybrid approach (AES-ECC), which secures the data over the cloud storage. Experimental setup is created for our proposed hybrid scheme and comparison can be done with other methods and hybrid scheme as well. We found that our proposed hybrid scheme outperforms other security schemes in terms of performance and efficiency.

4. Proposed Framework

In this section, we provide the design details of the proposed approach. We highlight the significance of combining ECC and AES and focus on the algorithm used in this approach.

4.1. Defining ECC and AES

ECC is a well-known cryptographic technique that uses following asymmetric key encryption and secures the data from unauthorized access. It uses pairs of public and private keys which ensures the security of the ECC. Two dimensional fields are used by ECC as binary and prime fields. Hacking is not easy if we are using this cryptography technique because it uses the enhanced operations and creates a relation among binary and primary fields and this relation in ECC cannot be read by unauthorized people. Small key size is the main factor of ECC. Maximum number points can help to find the appropriate field for the cryptographic implementation on data for the security measures. The first operation of the field selects the first number and then produces the large number based on the data which is between 0 to Z. Specifically, for the generation of the key, ECC is used and reduced the complexity of the operations. Due to the low-key size, the enhancement of ECC is much more than other cryptographic techniques. This paper uses the ECC techniques to optimize memory and space enhancement [28].

AES is one of the types of cipher text which uses the block cipher. This uses only one key for the encryption and decryption process to secure the data. It has many performance operations which are limiting on cloud storage like statistical analysis, searching on cloud storage, and others like these. It is the widely used strategic algorithm over cloud computing to improve the security rules over cloud storage. This paper uses the AES algorithm because it is easily implemented as well as being time compatible with the cloud storage accessibility of data [29,30].

4.2. ECC and AES—A Combined Hybrid Proposed Approach

ECC and AES create the most advanced and efficient cryptographic technique over the cloud storage. We can say that single AES is little bit slower than the hybrid (ECC-AES) method due to its larger key size, while the hybrid method allows reduced key size as well as a faster security mechanism for securing the data. As small key size is the main property of ECC, when AES uses ECC for encryption, key size is reduced, and performance is increased [31]. ECC uses encryption and decryption key standards to reduce the key size and create the secured key system. ECC is the most appropriate technique to use along with AES to get the data secured from unauthorized use. Once the key size is set, then ciphertext will generate the encryption and decryption of data. The key generated by ECC is used by AES. The combined effect of both ECC and AES is suitable for the proposed technique at cloud storage to get the secured system. This helps to reduce the storage size with secure data. Below, in Figure 5, is the block diagram of the proposed algorithm.

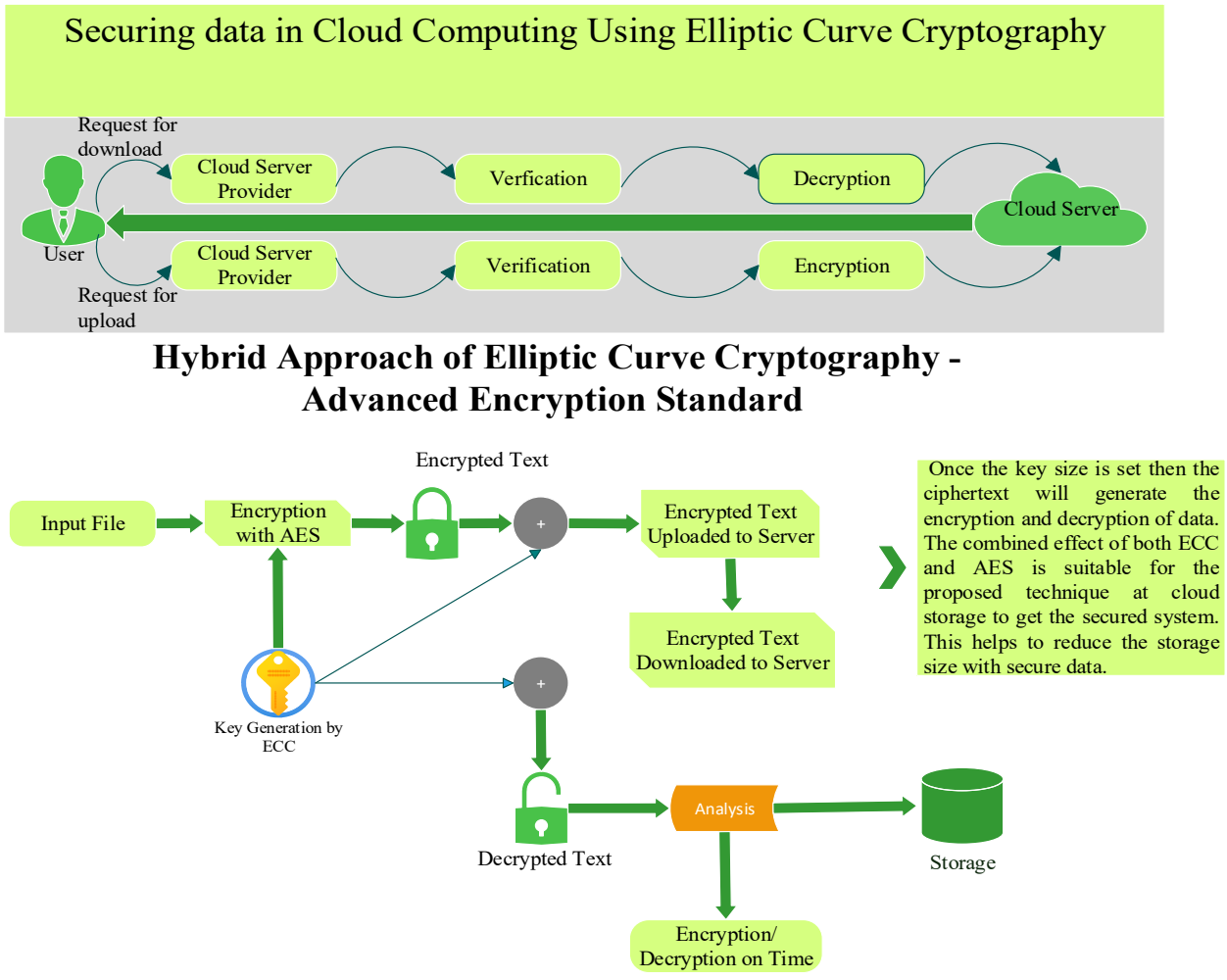

Figure 5. Representation of ECC and AES algorithm.

In the above figure, it can be clearly seen that AES along with ECC effectively secures data over cloud storage. The novelty of the proposed method can be clearly seen in the new proposed diagram, in which there is secure transmission of user data to server and then storage mechanism is even secured due to encrypted data. Moreover, novelty can be determined in terms of computational cost and time. However, attack prevention can be done in the following way: for example, if an attacker wants to attack on the user side in order to gain user personal information or for some other purpose, in the proposed approach, once the user uploads the input file, the file is converted into encrypted text

with the help of AES encryption, so the text is fully encrypted. Therefore, if an attacker performs an attack and somehow get the user-uploaded file, then it is useless because the information was already encrypted upon uploading. Similarly, on the other end, if an attack is performed, the attacker is not able to decrypt the encrypted file and hence the data are secured from attacks.

4.3. Algorithm for the Proposed Framework

4.3.1. Generation of a Public Key by Elliptic Curve Cryptography

Public key generation using ECC

Step I. Select any number n as the prime number.
Step II. Select any number for the generation of the public key as n (a)
Where n (a) < n
Step III. Compute the point on the curve as G
Where G > n
Step IV. Calculation of public key is:
P = n (a) ∗ G
Step V. Return the public key P after calculation.

4.3.2. Encryption and Decryption by Using Advanced Encryption Standard

Encryption/Decryption using AES

Step I. Take the input file
Step II. Now add the key generated by ECC which is the public key.
Step III. AES encryption is performed on the input file by using the public key which is generated by ECC.
Step IV. The encrypted file is uploaded on the server after the encryption by AES.
Step V. Once the file is uploaded then it will be downloaded at the server, and then file is translated by using the public key given by ECC so that the original file is decrypted.
Step VI. The performance of the system depends on the combined effect of ECC and AES, such as the storage space optimization and enhancement of the security services over the cloud server.

5. Experimental Results and Discussion

The uniqueness and efficiency of the system by the combination of ECC and AES is enhanced in the different way that many of the data over cloud storage are secured and get the secured connections for the encryption and description of the data. This means that the user can get the original message easily by applying these two techniques. Some advantages of ECC and AES over RSA are presented in the following subsection.

5.1. Advantages of ECC and AES

ECC is used to ensure the security of the data over cloud storage. The maintenance of the data with reduced key size can help to optimize the storage space as well as get the desired results. It uses the same 3072 bits as the RSA. The most beneficial thing about the ECC is the reduced key size and the encryption of data through a public key, which is in an optimized manner [32]. ECC is more beneficial compared to the RSA for using the latest algorithmic techniques applying to the encryption and decryption of data as well as the accuracy of the decrypted data. AES has many performance operations which are limiting on cloud storage like statistical analysis, searching on cloud storage, and others like these. It is the widely used strategic algorithm over cloud computing to improve the security rules over cloud storage. The public key is known by every person while the encryption and decryption process can also be done by the public key [33]. Many advantages and key sizes of ECC and AES over RSA are given in the Table 5 below.

Table 5. ECC and AES over RSA.

ECC	RSA	Key Size Comparison
160 bits	1024 bits	1:6 bits
256 bits	3024 bits	1:12 bits
384 bits	7068 bits	1:20 bits
512 bits	16360 bits	1:20 bits

Explanation: As you can see in the Table 5 and Figure 6, a medium key size is required for ECC as compared to RSA. Because of this, it provides better security than RSA. ECC-AES also provides better security with a smaller key size as compared to other cryptographic algorithms. It optimizes memory space and reduces computational complexity because of the smaller key size. Thus, by using a medium key size, a high level of data security can be obtained.

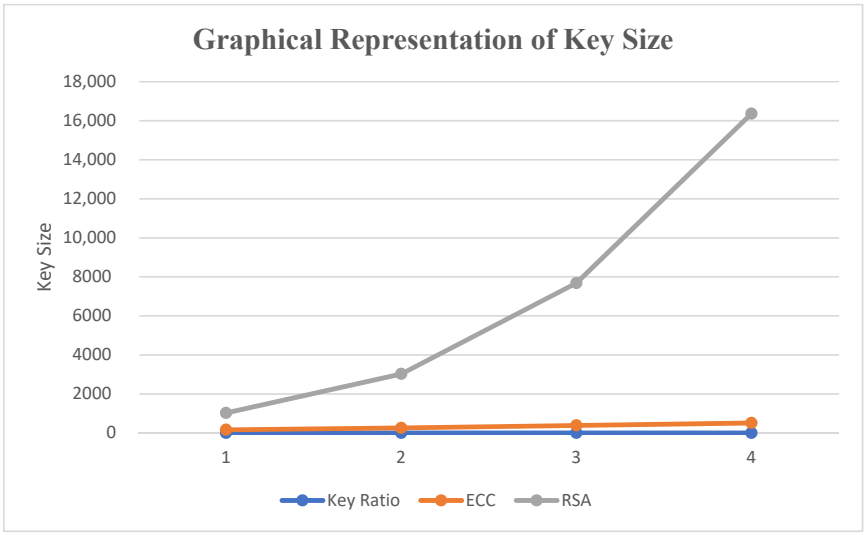

Figure 6. Graphical representation of key sizes.

5.2. Various Algorithms Comparisons

The analysis of different algorithms is compared to check the functionality and space optimization over cloud storage [34]. The comparison of different algorithms is given below in Table 6.

Table 6. Comparison of different Cryptographic Algorithms.

Factors	Proposed Hybrid	DES	3DES	Blowfish	RSA	Diffie-Hellman
No. of key	1	1	1	1	2	Key Exchange
Key Length in bits	64–256	56	112–168	32–448	1024	Key Exchange
Rounds	10	16	48	16	1	56
Limitation	Brute force	Brute force	Computational power	Key frequently changing	Key generation week	Cannot encrypt data

As you can see in the above Table 6, it shows the analysis of different cryptographic techniques based on different factors. Cryptographic algorithms have been compared with each other for performance evaluation on basis of number of keys used, keys in bits, rounds, and limitations.

5.3. Materials and Methods

This section provides details about the materials that we used in implementing our research paper. The following are some of the general details about the implementation:

Software	VS Code Editor
Language	Python
Python Library	SimpleCV
Image Format	Portable Network Graphics
Simulation tool	iFogSim
Graphs	Microsoft Excel
System Specs	Windows 10, SSD 500 GB, 1TB HARD DRIVE, 16 GB RAM

The Python code along with its complete guideis available at the following URL: https://github.com/sabarehmanciit/Hybrid-AES-ECC-Model (accessed on 27 September 2021).

5.4. Setup

We used Python to implement our proposed algorithm and to validate the efficiency and uniqueness of our proposed system. We used SimpleCV library for Python to read the images. The image format used by our code was Portable Network Graphics. The uniqueness and efficiency of the system by the combination of ECC and AES was enhanced in the different way that many of the data over cloud storage were secured and achieved the secured connections for the encryption and description of the data.

5.5. Data Size for Proposed Scheme

We took three different image datasets for the comparison of our proposed method with other existing schemes because images usually take more time than text data, so we wanted to check the computational cost as well as the time required to complete the encryption and decryption of the images. The reason for taking differentimages size is to compare performance in multiple scenarios. The size of the image datasets on which the experiments were performed is 3233, 4830, and 6308. Figure 7 represents the encryption time comparison of different algorithms.

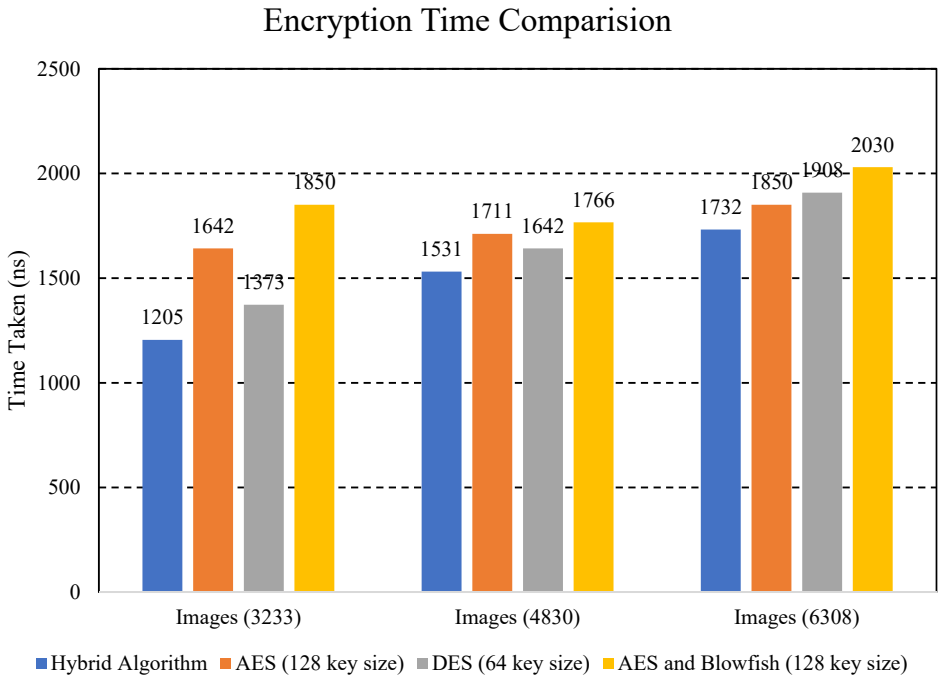

Figure 7. Encryption time comparison of different algorithms.

Explanation: Figure 7 shows comparison of the encryption time of the data with different cryptographic algorithms. The hybrid algorithm is the proposed algorithm. Results are prominent in that the hybrid ECC-AES approach took less time to encrypt data than the existing approaches due to its smaller key size. Moreover, the hybrid ECC-AES algorithm has characteristics of both algorithms which provides higher security by increasing the complexity and making the system strong against attacks. It can also be clearly seen that encryption time for the proposed h algorithm is much less as compared to other algorithms. As the time for encryption is reduced, our computational cost is also reduced, which is very effective. Hence, our proposed approach works more efficiently than others. Figure 8 represents the decryption time comparison of various algorithms.

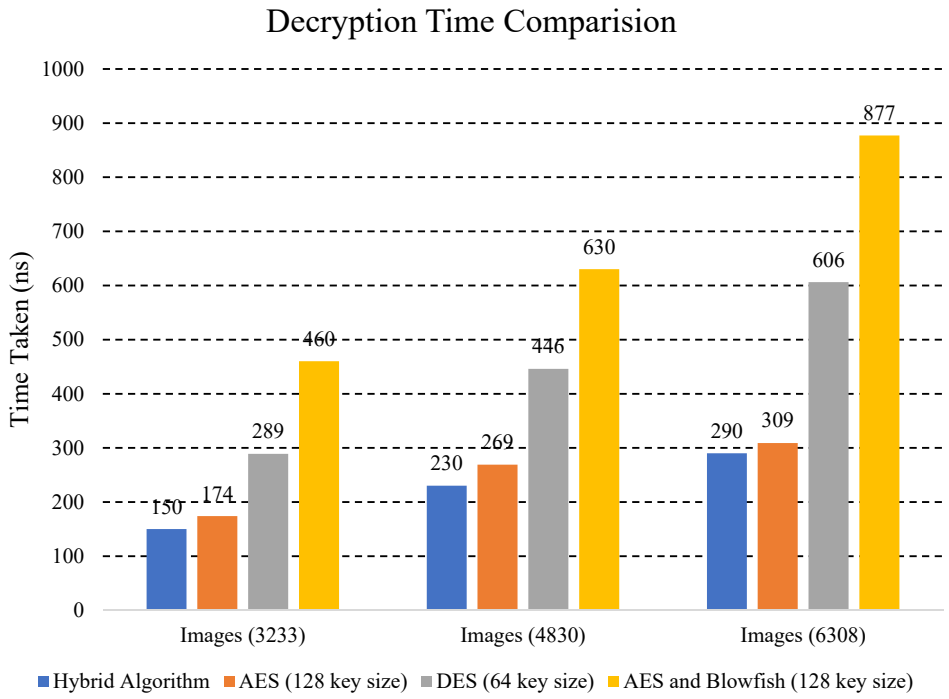

Figure 8. Decryption time comparison of different algorithms.

Explanation: Figure 8 shows comparison of the decryption time of the data with different cryptographic algorithms. The hybrid algorithm is the proposed algorithm. Results are prominent in that the hybrid ECC-AES approach takes less time to decrypt data than the existing approaches due to its smaller key size. Moreover, the hybrid ECC-AES algorithm has characteristics of both algorithms which provides higher security by increasing the complexity and making the system strong against attacks. It can also be clearly seen that decryption time for proposed hybrid algorithm is much less as compared to other algorithms. As the time for decryption is reduced, our computational cost is also reduced, which is very effective. Hence, our proposed approach works more efficiently than others.

The evaluation of our proposed scheme is based on encryption time and decryption time.

5.6. Performance Analysis

All the results and comparisons were performed on a system with the following specifications: Windows 10, Processor Intel(R) Core (TM) i7-8550U CPU @ 1.80GHz, 1992 MHz, 4 Core(s), 8 Logical Processor(s). The proposed algorithm was tested on Python environment.

5.6.1. Encryption Time

We also compared the encryption and decryption time with different key sizes for our proposed hybrid algorithm and with the existing algorithms (AES, DES, and Blowfish) for

final confirmation. The tests were performed using different keys, namely, 64 bits, 128 bits, 192 bits, and 256 bits. The proposed and existing encryption algorithms were tested for text data using different keys. In Tables 7 and 8, the encryption and decryption time of all the values in seconds are shown. Furthermore, we can see the visual analysis of these values for better understanding.

Table 7. Encryption time calculated using different key sizes.

Key Sizes	Hybrid	AES	DES	Blowfish
64	2.40	3.42	3.89	4.31
128	2.47	3.59	4.02	4.47
192	2.54	3.48	4.05	4.43
256	2.60	3.60	4.34	4.70

Table 8. Decryption calculated using different key sizes.

Key Sizes	Hybrid	AES	DES	Blowfish
64	1.64	2.69	3.11	3.89
128	1.77	2.82	3.30	3.94
192	1.88	2.93	3.38	4.10
256	2.10	3.08	3.58	4.25

In Figure 9, encryption time of proposed and existing algorithms have been compared.

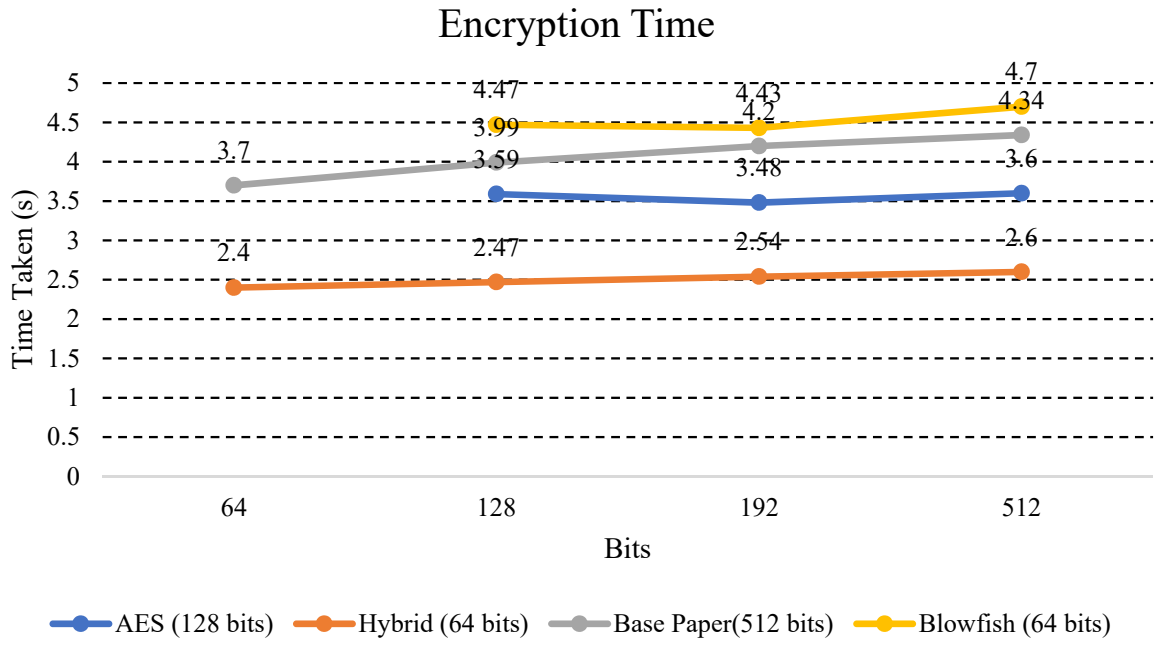

Figure 9. Comparison graph for encryption time in hybrid and other existing algorithms using different key sizes.

Explanation: The above Figure 9 shows the time taken by other existing algorithms (AES and Blowfish), Base Paper, and hybrid models for the same text data for the encryption process. It was found that the hybrid ECC-AES model took less time on key size in bits (64, 128, 192, and 256), while on the other hand, the other existing algorithms (AES, DES, and Blowfish) took more time on key size in bits (64, 128, 192, and 256).

5.6.2. Decryption Time

The decryption time calculated for the hybrid and other existing schemes is presented in Table 8.

In Figure 10, encryption time of proposed and existing algorithms have been compared.

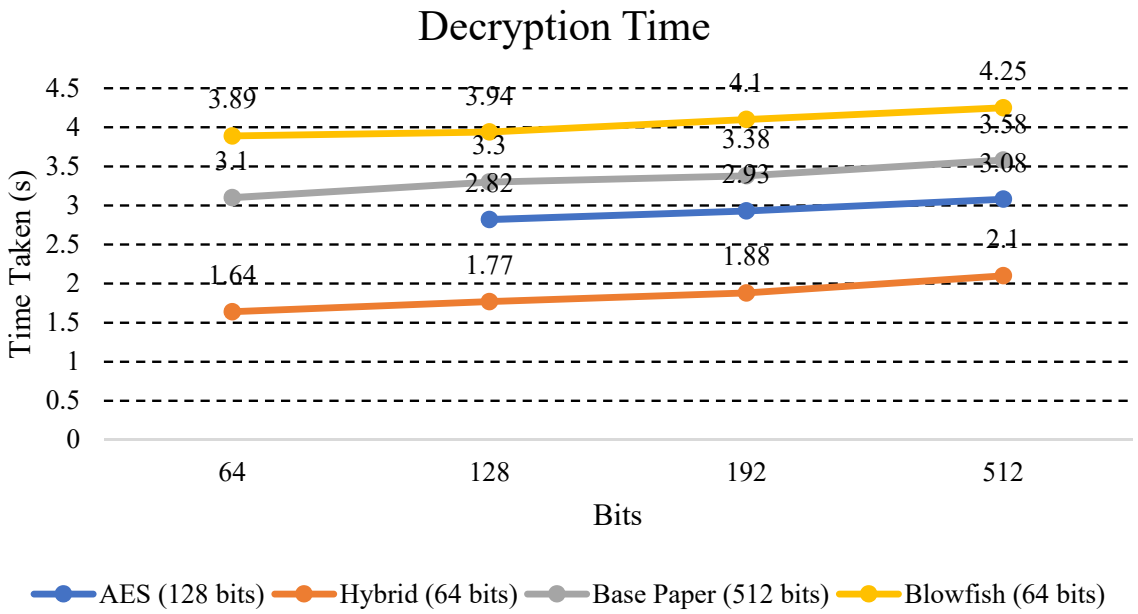

Figure 10. Comparison graph for decryption time in hybrid and other existing algorithms using different key sizes.

Explanation: Figure 10 shows the time taken by other existing algorithms (AES and Blowfish), Base Paper, and hybrid models for the same text data for the decryption process. It was found that the hybrid model took less time on key size in bits (64, 128, 192, and 256), while on the other hand, the other existing algorithms (AES, DES, and Blowfish) took more time on key size in bits (64, 128, 192, and 256).

Figures 9 and 10 show that our scheme will help in the optimization of the memory space and reduce the computational complexity as compared to other cryptographic algorithms. A medium memory size is required for the hybrid algorithm and less time is required for the encryption and decryption process as compared to other cryptographic algorithms.

5.6.3. Avalanche Effect

In general, the Avalanche Effect (Av) is the measure of change in an algorithm. It simply means that a smaller change in input can cause significant change in output of the text. We can calculate A_v by calculating changed bits in cipher bits and dividing this number by total number of cipher bits [35]. The formula by which we calculated Avalanche Effect of our proposed algorithm and base paper is:

$$Av = \text{Number of changed bits} / \text{Total number of bits}$$

We compared the Avalanche effect of our proposed algorithm with other encryption algorithms [1], in Table 9. An Avalanche effect greater than 50% indicates that a certain

algorithm has more security power as compared to others. Our proposed algorithm has the highest Avalanche effect showing that the system is more secure than others.

Table 9. Avalanche effect comparison.

Encryption Algos	1-bit Key Change	Avalanche Effect	1-bit Plain Text Change	Avalanche Effect
Base Paper	68	0.55	70	0.56
Blowfish	37	0.29	23	0.18
AES	64	0.51	71	0.55
Proposed	73	0.61	83	0.64

Table 9 shows the Avalanche effect for the various encryption algorithms. The more the change occurs in the cipher due to a single bit change in key or the plain text, the greater the avalanche effect. In Figure 11, we can see that the higher the avalanche effect, as with our proposed algorithm, the more it becomes difficult to break the algorithm easily. Hence, we can see that our algorithm has higher security in terms of its Avalanche effect. Below is the visual representation of the above table, where we can easily see that our proposed algorithm has higher security than other encryption algorithms.

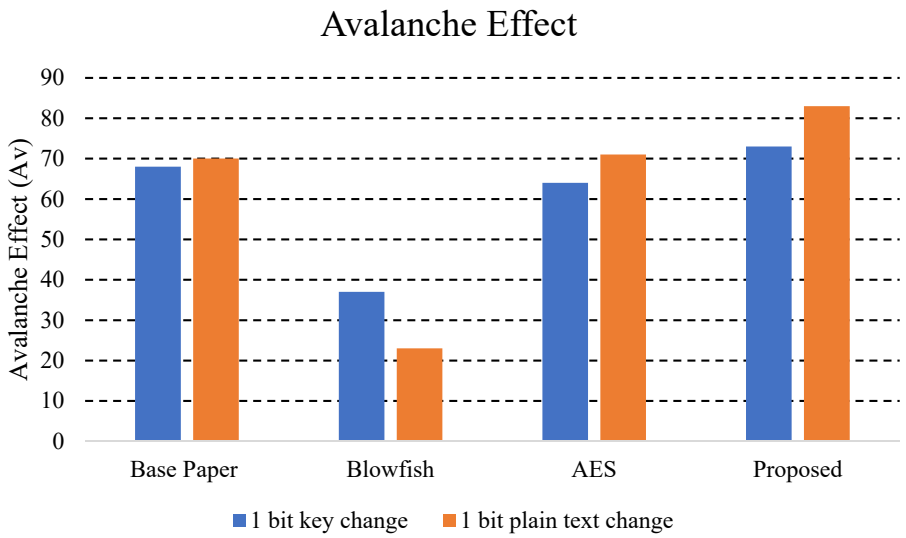

Figure 11. Comparison of Avalanche effect with other schemes.

5.6.4. Power Consumption

Encryption algorithms are the main part of the security systems. However, these algorithms consume a significant amount of computing resources such as CPU time, memory, and battery power. Moreover, resources in the cloud environment are specific to certain tasks [36]. Power can simply be defined in terms of Watts. In Table 10, the power consumption for relevant schemes is mentioned. It shows the power consumption of encryption time in Watts, here we compared our proposed algorithm with Base Paper.

Table 10. Power consumption of encryption time (Base Paper).

Power Consumption (Watts)	Proposed Hybrid	AES	DES	Blowfish
64-bit key size	2.40 W	3.42 W	3.89 W	4.31 W
128-bit key size	2.47 W	3.59 W	4.02 W	4.47 W
192-bit key size	2.54 W	3.48 W	4.05 W	4.43 W
256-bit key size	2.60 W	3.60 W	4.34 W	4.70 W

Table 11 shows the power consumption of decryption time in Watts, here we compared our proposed algorithm with Base Paper.

Table 11. Power consumption of Decryption time.

Power Consumption (Watts)	Hybrid	AES	Base Paper	Blowfish
64-bit key size	1.64 W	2.69 W	3.1 W	3.89 W
128-bit key size	1.77 W	2.82 W	3.3 W	3.94 W
192-bit key size	1.88 W	2.93 W	3.38 W	4.10 W
256-bit key size	2.10 W	3.08 W	3.58 W	4.25 W

6. Conclusions

IT-related services such as cloud computing provide efficient services regardless of the user's knowledge about technology. Data can be stored, managed, improved, and accessed through a cloud interface provided by third party cloud service providers via the Internet with minimal effort regardless of the location of the user. Cloud services provide many facilities to the user. Users are the people who are using the services depending on the kind of cloud services. Services are provided at low cost as well as being a beneficial thing for so many users to access their data from any place. Cloud services can be made available on any machine as there is no need to take your device along with you. However, the drawback of cloud services is the low data security which may be overcome through special strategies and must be secured. Specifically, for the generation of the key, ECC is used to reduce the complexity of the operations. Due to the low-key size, the enhancement of ECC is much better than other cryptographic techniques. AES in combination with ECC can do a lot better with the optimization and security of the data. However, much security is still needed in the future to expand the concept of cloud computing through cryptographic techniques. In future, this research can be improved by increasing the security of the hybrid approach. Multiple security layers can be added to enhance the productivity and efficiency of the system.

Author Contributions: Conceptualization, methodology and initial writeup, S.R. and N.T.B.; research supervision and results validation, M.A.S.; formal analysis, A.A.; review, editing and funding acquisition, A.O.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is available on the github on the following link: https://github.com/sabarehmanciit/Hybrid-AES-ECC-Model (accessed on 27 September 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shukla, D.K.; Dwivedi, V.K.; Trivedi, M.C. Encryption algorithm in cloud computing. *Mater. Today Proc.* **2020**, *37*, 1869–1875. [CrossRef]
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. [CrossRef]
3. Qazi, R.; Khan, I.A. Data security in cloud computing using elliptic curve cryptography. *Int. J. Comput. Commun. Netw.* **2019**, *1*, 46–52. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
13. Awad, W.S. A framework for improving information security using cloud computing. *Int. J. Adv. Res. Eng. Technol.* **2020**, *11*, 264–280. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
20. Saeed, Z.R.; Ayop, Z.; Azma, N.; Rizuan Baharon, 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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
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. [CrossRef]
30. Sood, S.K. A combined approach to ensure data security in cloud computing. *J. Netw. Comput. Appl.* **2012**, *35*, 1831–1838. [CrossRef]
31. Mendonca, S.N. Data security in cloud using AES. *Int. J. Eng. Res. Technol.* **2018**, *7*. [CrossRef]
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*. [CrossRef]
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.