Journal of Engineering and Applied Sciences 13 (15): 6043-6052, 2018
ISSN: 1816-949X

ECDH-ECC: A Combination of Elliptic Curve Cryptography and Diffie Hellman Based Cryptography Technique for Big Data Security

Adilakshmi Kameswari Vadavalli and R. Subhashini
Department of Computer Science Engineering, School of Computing,
Sathyabama University, Chennai, India

Abstract: In recent days, big data is one of the widely used technology in many cloud based applications. So, providing security and ensuring privacy for the data that stored in cloud is an essential task due to the distributed storage. For this purpose, different security mechanisms based on encryption and decryption are developed in the traditional works. But it mainly lacks with some major limitations such as increased computational complexity, reduced security and high overhead. Thus, this research aims to develop a new mechanism for providing security to the big data that stored in cloud. At first, the data owner selects their data and generating the corresponding base point to the data based on the elliptic curve. Then, the corresponding key is generated for the data by implementing an Elliptic Curve Diffie Hellman (ECDH) algorithm. Further, it is encrypted with the use of Elliptic Curve Cryptography (ECC) based encryption technique. After that, the encrypted data is send to the Cloud Service Provider (CSP) and its corresponding signature is generated at the data owner side and is forwarded it to the Third Party Auditor (TPA). When, the data user sends the request to the CSP which sends the encrypted data and the TPA sends the generated signature to the data user. After that, the Message Authentication Code (MAC) generation and verification processes are performed at the data user side. If the generated signature and the received signature are same, the data is decrypted by the data user. During experimentation, the performance results of existing and proposed security mechanisms are validated based on different measures for proving the superiority of the proposed ECDH-ECC technique.

Key words: Cloud Service Provider (CSP), Third Party Auditor (TPA), Elliptic Curve Diffie Hellman (ECDH), Elliptic Curve Cryptography (ECC), Message Authentication Code (MAC), data owner, data user

INTRODUCTION

Big data is the most widely used technology in many cloud based applications such as business, e-Learning, marketing and healthcare (Hashem *et al*., 2015). It contains a large amount of information as shown in Fig. 1, so, it is highly difficult to process those data. The major characteristics of big data are variety, volume, variability, velocity and complexity (Assuncao *et al*., 2015; Chen *et al*., 2014). The significant benefits of using big data (Fernandez *et al*., 2014) are as follows:

- It offers a rich set of tools and options to map the entire data landscape
- It allows the individual to analyze the internal threats in the cloud
- It has the ability to integrate both the unstructured and structured data
- Also, it can easily extract the value from varying data sources such as web, mobile devices and radio frequency identification

But providing security to the big data is one of the demanding and critical task in the recent days. The major security issues associated with big data storage and retrieval in cloud are as followed:

Privacy and confidentiality: The data user must be secured with authorization during the data hosting on cloud. Also, the assurance must be provided to the data users for increasing the data security.

Fig. 1: Big data

Corresponding Author: Adilakshmi Kameswari Vadavalli, Department of Computer Science Engineering, School of Computing, Sathyabama University, Chennai, India

Data integrity: The cloud service providers ensure both the data integrity and security. When the data integrity requirements exists, the source and protection of the data must be prevented.

Data location and relocation: High degree of mobility is provided to the data by the cloud. So, the provider has the responsibility to ensure the security of the systems and to safeguard the information for robust authentication.

Data availability: The cloud users store their data in various locations, so, the data availability becomes a relatively tough task.

Storage, backup and recovery: When the user wants to transfer the data in the cloud, the data resilience storage must be ensured.

Problem identification So, the existing works develop various encryption and decryption techniques for providing security to the big data (Jin *et al*., 2015). In which the cryptography technique (Wei *et al*., 2014; Chu *et al*., 2014) is mostly used for protecting the data against the unauthorized users by providing authenticity integrity and access control. The cryptographic techniques is used to encrypt the original data into unknown format which is stored into the cloud (Prakash *et al*., 2014; Yakoubov *et al*., 2014). The CSP and TPA are known as the investigators (Zhu *et al*., 2013; Rizvi *et al*., 2015; Tahilyani *et al*., 2015) in cloud they validates the signature of the corresponding data for ensuring the security. Moreover, the data must be protected before outsourcing it to the user, so, enabling the privacy preservation is also an important consideration (Daniel and Vasanthi, 2016). The traditional techniques (Shimbre and Deshpande, 2015) mainly focuses on the block level operation, overhead reduction, confidentiality and integrity maintenance. But it failed to reduce the computational complexity, time consumption and memory consumption.

Objectives: Based on the problem identification, this research work has the following objectives:

- To securely store and share the Big data in a cloud environment
- To generate the key for the data owner's data by creating the base point, an Elliptic Curve Diffie Hellman (ECDH) algorithm is implemented
- To encrypt the data based on the generated key, an Elliptic Curve Cryptography (ECC) technique is used
- To decrypt the data at the user side, the Message Authentication Code (MAC) generation and validation processes are performed

Organization: The remaining sectors in this study are structured as follows: the existing security mechanisms and algorithms used for big data storage in cloud are surveyed.

Literature review: In this study, the existing techniques and algorithms related to big data security are surveyed with its benefits and demerits.

Hongbing *et al*. (2015) suggested an alternative approach to provide a secure data sharing by dividing the big data into small parts. Here, the tenant's big data was used which does not required any additional security requirements before storing it into the cloud. Also, a cryptographic virtual mapping was employed to ensure the security and confidentiality of the big data. Here, the efficiency of this mechanism was evaluated based on the overhead and efficiency of the suggested scheme. However, this study required to increase the level of security before storing the data into cloud. Sookhak *et al.* (2017) developed a Remote Data Auditing (RDA) mechanism for ensuring the security and integrity of the data that stored in cloud. Here, a Divide and Conquer Table (DCT) was utilized to efficiently support the data operations such as insert, append, update and delete. The main intention of this study was to reduce the required communication and computation cost with increased efficiency. The main limitation that observed from this study was, it has increased communication overhead in a cloud storage systems. Puthal *et al*. (2015) introduced a new multi-sharing mechanism for providing the privacy preservation to the data that stored in cloud. In this study, the benefits of the proxy re-encryption was integrated with the anonymous technique for increasing the security of the data. The key properties that focused in this research were as follows:

- Anonymity
- Multiple receiver update
- Conditional sharing

Moreover, the researchers of this study stated that these primitives were highly applicable for the applications of electronic encrypted data sharing and secure email forwarding. Youssef (2014) designed a new framework based on the big data analytics for Healthcare Information Systems (HISs). The properties that mainly focused in this research were availability, data integrity

interoperability and secure data sharing. This study stated some of the important security issues in cloud based HIS which includes:

- Authorization
- Authentication
- Non-repudiation
- Availability
- Integrity
- Confidentiality

Inukollu *et al.* (2014) investigated various security issues in cloud computing for processing the big data in many small and large scale organizations. Here, some security measures such as file encryption, logging, network encryption, node authentication and testing were studied. Moreover, the access control mechanism was utilized to control the policies of the sensitive data. Wu *et al.* (2017) introduced a New Extensive Data Access Control Scheme for Multi-Authority Cloud Storage System (NEDAC-MACS) to improve the level of security in a cloud systems. The main focus of this technique was to perform the cipher-text based communication between the server and Attribute Authorities (AAs) and to reduce the overall overhead of storage. In this research, the following processes were performed during data storage:

- The AAs generated the secret key
- The data encryption was performed by the owners
- The users performed data decryption with the use of cloud servers

However, this research failed to increase the efficiency of the data storage. Chang and Ramachandran (2016) developed a new framework, namely, Cloud Computing Adoption Framework (CCAF) to increase the security of the cloud data. It includes the following stages:

- Firewall and access control
- Identity management and intrusion prevention
- Convergent encryption

Furthermore, the risk assessments were performed to analyze the risk by identifying the security goal and the artefacts were developed based on the specifications. Then, the eliction technique was utilized to categorize the requirements. Liu *et al.* (2016) recommended a two-factor data security protection mechanism to improve the security of cloud storage system. The suggested system was developed based on the Identity Based Encryption (IBE) mechanism in which the cipher text was generated by the sender and downloaded from the cloud by the receiver. The entities that involved in this design were sender, private key generator, cloud server receiver and Security Device Issuer (SDI). This technique has the main advantages of increased efficiency and reduced complexity. But this research failed to present the security proof of the suggested system. Li *et al.* (2015) designed an efficient tool, namely order Preserving Encryption (OPE) for preserving the order of relevance in the cloud storage system. Here, the data traffic was reduced by retrieving the data based on the ranking from the large amount of documents. Also, the relevance between the searching query and files was estimated by using the ranking function during the ranked search. The dummy document IDs were added into the inverted index for estimating the relevance scores.

Ali *et al.* (2017) introduced a new model, namely, Data Security for Cloud Environment with semi trusted third party (DaSCE) mechanism for providing security to the cloud data system. The motive of this technique was to provide access control and secure key management. Also, a File Assured Deletion (FADE) protocol was developed to ensure the privacy integrity and access control of the outsourced cloud data. But this study required to enable the security for both data sharing and data forwarding. Kelbert *et al.* (2017) formed a layered architecture with secure creation and integration of micro-services for increasing the security of big data processing. The main focus of this study was it provided an effective mechanism for improving the state-of-the-art of cloud. Here, the dependability features were integrated with the cloud stack for migrating the critical applications of the cloud. The drawback of this study was it required to reduce the computational complexity of the data storage. Jain and Khan (2017) recommended some access control policies for providing secure access to the big data that stored in a cloud environment. The requirements that guaranteed the cloud security were completeness, correctness and privacy. Here, the Attribute Based Encryption (ABE) was used to construct the access control policies of the cloud system. Maheswari *et al.* (2016) implemented a three planning mechanism to handle the privacy and security issues in the cloud environment. Here, different encryption standards such as Advanced Encryption Standard (AES), Message Digest 5 (MD5) and Data Encryption Standard (DES) were used for high security. The processes that involved in this work were, anonymity, multiple receiver communication, conditional sharing and privacy preservation.

However, this study required to prove the efficiency of the suggested technique by using different performance measures. Terzi *et al.* (2015) investigated various security and privacy issues in cloud for processing the big data. Typically, key generation and distribution between the servers and users was a challenging task in a cloud. Here, the Condition Proxy Re-Encryption (CPRE) mechanism was used to securely share the data. The benefit of this study was it efficiently reduced the overhead by implementing an enhanced encryption and decryption techniques. Katal *et al.* (2013) investigated the challenges issues and tools of the big data technology which includes the followings:

- Privacy and security
- Data sharing
- Data processing and storage

Moreover, the technical issues such as fault tolerance, scalability, quality of data and heterogeneous data were also discussed in this study. Sharma and Navdeti (2014) reviewed various security issues, threats and its corresponding solutions for providing security to the big data. Here, a new mechanism, namely, Authentication Authorization Auditing and Encryption (3ADE) was developed to solve the big data security issues. It includes distributed computing, data fragmentation, data access control, node to node communication, client interaction and no security. Wei *et al.* (2014) introduced an auditing protocol, namely, SecCloud for providing secure data storage and computation auditing. Here, the batch verification process was performed to handle the requests of separate users for improving the efficiency of cloud. Moreover, uncheatable cloud computation and privacy cheating discouragement were defined for modeling the security problems.

From the survey it was analyzed that the existing security mechanisms have both benefits and drawbacks but it mainly lacks with the following issues:

- Increased computational complexity
- Reduced response time
- Increased overhead
- Minimized confidentiality and security
- Not highly efficient
- Same key is used for both encryption and decryption
- Inexpensive key management

To solve these issues, the proposed research aims to develop a new security mechanism with the use of enhanced key generation and encryption mechanisms.

MATERIALS AND METHODS

Proposed method: In this sector, the detailed description about the proposed big data storage and security mechanism is provided with the clear flow illustration. The main aim of this study is to increase the security of big data by performing the secure key generated based encryption process. This research includes the stages of base point generation, secret key generation, encryption, signature generation, verification and decryption, then its flow is depicted in Fig. 1. Initially, the data owner selects the data and its corresponding base point by the use of Elliptic Curve Diffie Hellman (ECDH). Then, the key is generated for the data based on the generated base point then the data is encrypted by using the Elliptic Curve Cryptography (ECC) technique then it is forwarded to the Cloud Service Provider (CSP). Then, the data owner generates the corresponding signature for the encrypted data and forwards it to the Third Party Auditor (TPA) for verification. Here, the authorization is performed between the data owner and user based on the signature verification process. When the user sends the request to the CSP, it sends the encrypted data and TPA sends the signature to the user. Furthermore, the use generates the Message Authentication Code (MAC) for the encrypted data, then the user matches it with the received signature from the TPA. If both signature are same, the data is decrypted by the user for further use.

Data owner: Typically, the data owner contains a collection of data that is outsourced to the user with the help of CSP. Before outsourcing it to the user, the data must be converted into some unknown format for providing security against the unauthorized users. For this purpose, the processes such as base point generation, key generation and data encryption are performed which increases the security of the data. So, the data owner uses an efficient key generation and encryption techniques such as ECDH and ECC for data security. Also, the data owner required to distribute the secret between multiple authorized users by implementing a cryptographic mechanism. Moreover, the data owner maintains the record for calculating the relevance score which reduces the computational overhead. The major tasks that performed by the data owner are as follows: base point generation, secret key generation and signature generation (Fig. 2).

Base point and key generation: Initially, the base point is generated for the particular data by implementing the ECDH technique which establishes a secret share by exchanging the data for a secure communication. The major reason for using this technique is it provides an efficient solution for creating a common secret between

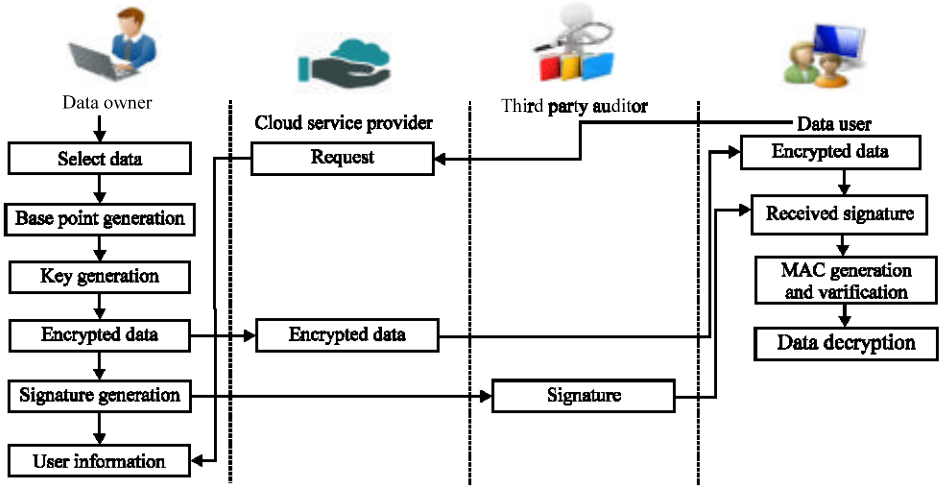

Fig. 2: Overall flow of the proposed big data storing and sharing system

two users. Also, it provides a dynamic key for ensuring the confidentiality of the communication. Moreover, the ECDH is a kind of anonymous technique that allows two authenticated parties for a secure communication. It establishes a shared secret for exchanging the data in a public cloud. In this technique, the procedure of elliptic curve is integrated with the standard DH algorithm. After generating the base point, the keys such as private key, public key and secret key are generated for the appropriate data. The working procedure of base point generation and key generation using ECDH technique are illustrated as follows (Algrothim 1).

Algorithm 1; Base point generation:

Initialize a = 11, b = 22, u = 29, x = 3, y = 1, s = 0

$a=\left(\left(x^3+ax+b\right)\%u\right)$

If a != 0 then
  For i = 1: u do
    $s=\left(\left(s+i\right)\%u\right)$
    If a == s then
      Break
    Else
      Y = y+1
      I =i+2
    End if
  End for
End if
If $y^2 = x^3+ax+b$ then
  g = x
  p = y
End if
//Key Generation
a ← private key (Sender)
b ← private key (Receiver)
Pb_s ← Public key (Sender)
Pb_r← Public key (Receiver)
Sk_s← Secret key (Sender)
Sk_s← Secret key (Receiver)
Compute public key, $Pb_s = ((g^a)\%p)$
Compute the public key, $Pb_r = ((g^b)\%p)$
Compute the secret key, $Sk_s = ((B^a)\%p)$
Compute the secret key, $Sk_s = ((A^a)\%p)$

Encryption: Encryption is defined as the process of converting the original data into an unknown format for protecting the data that stored in the cloud server. Typically, there are four different ways are used to perform the data encryption which includes file level, directory level, full disk level and application level. In this research, the ECC technique is used to encrypt the data owner's data by using the generated secret key. It is an emerging and attractive public key cryptosystem when compared to the existing Rivest Shamir Adleman (RSA) technique. The ECC technique is highly depends on the alphabetical table and key table which is the strength of this algorithm. Here, varying characteristics are symbolized as the coordinates of the curves in which the group structure is formed with a finite number of integer points. Furthermore, it creates the complexity during encryption, so, the unauthorized person cannot easily access it. The major benefits of using ECC are as follows (Algrothim 2).

- Faster computation
- Bandwidth saving
- Low power consumption

Algorithm 2; ECC encryption and decryption:

//ECC Encryption
A ← Select data
Xd ← Encrypted data
P1 ← Private key
P2 ← Public key
k ← Private Variable
P2 = P1×p
for i = 0:TP do
  while TP.i.data
    Xd = A.data+(k×P2)
  end while
end for
// ECC Decryption
CX←Cipher text 1

```
Yd←Decrypted data;
CX = k * p
for i = 0:TP do
      while Xd.i.data
          Yd = Xd.data-(P1*CX)
      end while
end for
for i = 0: TP do
          M.i ← MAC verification (TP.i)
end for
DA ← Merged Data
for i = 0:TP do
        DA += Ds
End for
```

Signature generation: After encrypting the data, the data owner generates the corresponding signature for the encrypted data. Then, the data owner forwards the signature to the TPA for verifying the authenticity of the data user. Also, the TPA validates the secret of both sender and receiver by checking secret keys. The main motive of generating the signature is to validate the authorization of the user in cloud. Moreover, this process enables an on-demand, convenient and reliable data access with better security (Algrothim 3).

Algorithm 3; Secret Validation by TPA:

```
Sks←Secret key (Sender)
SkR←Secret key (Receiver)
V←Verification
If Sks = SkR then
        V = Valid secret
Else
        V = Invalid secret
End if
```

Cloud service provider: In cloud, the CSP has some significant resources and knowledge for building and managing the computing systems. In which, the data owner stores their data into a set of cloud servers that are executing in a distributed manner. In this environment, the CSP receives the encrypted data from the data owner and sends it to the corresponding user based on their request. The CSP is responsible for the secure service provisioning to the data users.

Third party auditor: The TPA performs two kind of auditability such as private and public, in which an increased scheme efficiency was attained by using the private auditability. Consequently, the public auditability allows anyone for the correctness of data storage and it does not maintain any private information. The TPA facilitates the interaction between the two parties based on the trust. In this research, the TPA obtains the generated signature from the data owner, then forwards it to the data user who request the particular data to the CSP. For better efficiency and security, the TPA performs multiple auditing tasks with strong authentication and authorization. Moreover, it verifies the integrity of the data user by checking the authenticity. Also, it verifies with the CSP to check whether the user is authenticated or not. Furthermore, the TPA is responsible for improving the storage accuracy, group auditing, reliability and prevention.

Data user: Here, the data user sends the request to the CSP for accessing the particular data. After sending the request, it receives the encrypted data from the CSP and its corresponding signature from the TPA. Then, it separately generates the MAC for the encrypted data and it matches both the generated signature and the received signature. If both are the data user can decrypt the data by using the signature.

RESULTS AND DISCUSSION

MAC generation: Typically, authentication is defined as the process of validating the authenticity of the user based on the authenticity code. It uses some cryptographic hash functions for generating the MAC value with the use of secret key. During this process, the hash key is generated for the encrypted data. After that, the tag is generated with the use of hash key, based on this value, the data can be further decrypted (Algorithm 4).

Algorithm 4; MAC generation:

```
Input:  Encrypted data
Output:     MAC Code
Step 1:     Generate the hash value for the encrypted data
            H(B)-Hash value of the encrypted file
            H (B)-Hash value of the encrypted file
            HK-Hash Key
Step 2:     Tag (H) = T (Tag generation)
            H-File
            T-Tag of the file H
Step 3:     MAC = T
```

Data decryption: After generating the MAC, the generated signature and received signature are validated. If both are same, the data is decrypted by using the ECC decryption technique. Decryption is defined as the process of converting the encrypted data into the original data that is uploaded by the data owner into a cloud.

Performance analysis: In this sector, the experimental results of both existing and proposed security mechanisms are evaluated by using various performance measures which includes key generation time, MAC generation time, encryption time, decryption time, computational overhead and memory consumption.

Key generation time: Key generation time is defined as the amount of time that ECDH-ECC taken for generating the key for the particular data. It is calculated as follows:

$$\text{Key generation time} = \text{Information transferring time} + \text{Execution time} \quad (1)$$

Here, the key generation time of ECDH-ECC is calculated with respect to both number of users as shown in Fig. 3. From this analysis, it is observed that the proposed ECDH-ECC requires reduced key generation for varying number of users.

MAC generation time: MAC generation time is defined as the amount of time taken by a data user to generate the MAC code for the received encrypted data which is graphically illustrated in Fig. 4. Here, the MAC generation time is calculated with respect to the different partitions. In this analysis, it is proved that the proposed ECDH-ECC requires reduced MAC generation time for the encrypted data.

Encryption time: Encryption time is defined as the amount of time taken by the data owner to encrypt the original data into an encrypted data. Figure 5 shows the encryption time of the proposed ECDH-ECC with respect to varying data size. Typically, the encryption time is represented in terms of milliseconds and is calculated as follows:

$$\text{Encryption time} = \text{Ending time} - \text{Starting time} \quad (2)$$

From this illustration, it is analyzed that the encryption time can be increased with the linear increase in the data size.

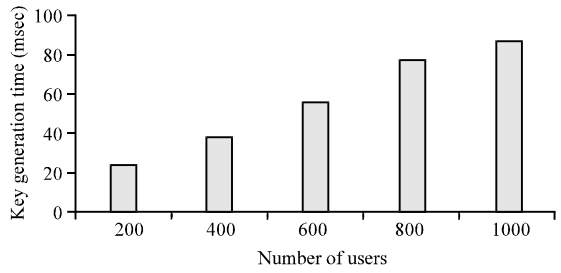

Fig. 3: Key generation time

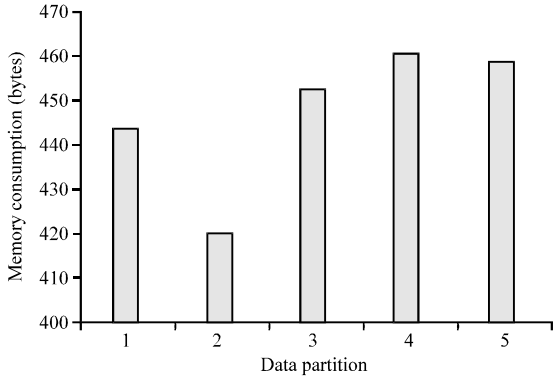

Fig. 4: MAC generation time

Decryption time: Decryption time is defined as the amount of time taken by the data user to decrypt the encrypted data which is expressed in terms of milliseconds. It is calculated as follows:

$$\text{Decryption time} = \text{Ending time} - \text{Starting time} \quad (3)$$

Figure 6 shows the decryption time of the proposed ECDH-ECC technique with respect to varying data size (kb). From this, it is evident that the decryption time can be increased with the linear increase the data size.

Memory consumption: Memory consumption is defined as the amount of memory that is utilized for data storage in cloud and also, it is defined as an occupied capacity of CPU. Figure 7 shows the memory consumption of the proposed ECDH-ECC technique with respect to the partitions of the encrypted data. From this illustration, it is evaluated that the proposed ECDH-ECC requires the reduced memory space for storing the data into the cloud server.

Comparative analysis: Table 1 compares the existing (Maheshwari *et al*., 2016) and proposed encryption techniques based on the execution time for analyzing its efficiency. The existing techniques considered in this research are Key Policy Attribute Based Encryption

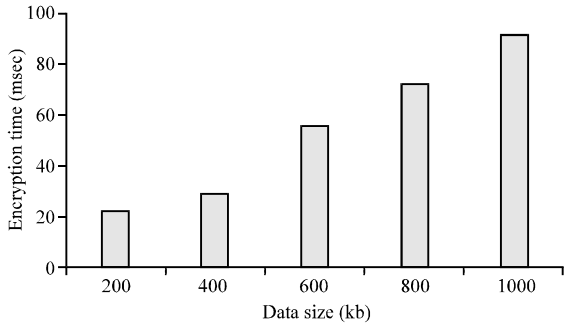

Fig. 5: Encryption time

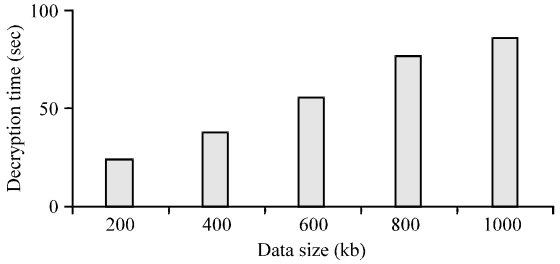

Fig. 6: Decryption time

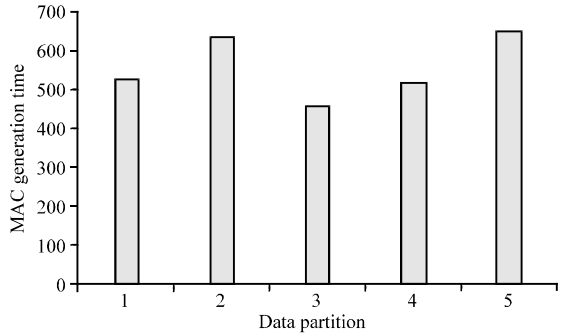

Fig. 7: Memory consumption

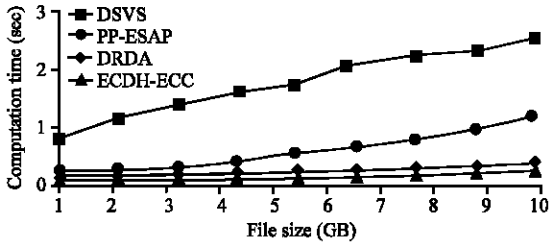

Fig. 8: Computation time of existing and proposed techniques

Table 1: Comparison between existing and proposed techniques

Algorithms	Number of user records	Execution time (sec)	Efficiency
KP-ABE	20	80	70
PROXY	20	120	60
ECDH-ECC	20	50	90

(KP-ABE) and PROXY. In which the efficiency of the algorithm is evaluated based on the execution time of these techniques.

Figure 8 shows the computation time of existing (Sookhak *et al*., 2017) and proposed security mechanisms with respect to varying file size (GB). The existing mechanisms considered in this analysis are Distributed Storage Verification Scheme (DSVS), Privacy Preserving-Efficient Storage Auditing Protocol (PP-ESAP) and Dynamic Remote Data Auditing (DRDA).

CONCLUSION

This study proposed a new security mechanism, namely, ECDH-ECC for enabling a secure data sharing and storing in a cloud environment. This framework contains four entities such as data owner, CSP, TPA and data user. Here, the data owner uploads their data in a cloud before that it performs the key generation and encryption processes by using the ECDH-ECC technique. The main reason of implementing ECDH is it efficiently generates the key with increased complexity, so, the unauthorized user cannot easily access the key. Moreover, the ECC is a highly secured encryption technique which encrypts the data based on the generated signature. After that, the data owner sends the encrypted data to the CSP and generates the corresponding the signature for the data. When the data user sends the request to the CSP for access the particular data, the CSP sends the encrypted data to the user. Correspondingly, the TPA forwards the signature to the data user which performs the MAC verification process. If both the generated and received signatures are same, the data is decrypted by the user. In experimental evaluation, the performance results of existing and proposed security mechanisms are validated by using the measures of key generation time, MAC generation time, encryption time, decryption time, computational overhead and memory consumption.

RECOMMENDATIONS

In future, this research will be enhanced by implementing this technique for the real time application scenarios. Also, the group key generation process can be performed to improve the level of security.

ACKNOWLEDGEMENTS

We wish to acknowledge the Department of Science and Technology india and School of Computing, Sathyabama University, Chennai for providing the facilities to do the research under the DST-FIST Grant Project No.SR/FST/ETI-364/2014.

REFERENCES

Ali, M., S.U. Malik and S.U. Khan, 2017. DaSCE: Data security for cloud environment with semi-trusted third party. IEEE. Trans. Cloud Comput., 5: 642-655.

Assuncao, M.D., R.N. Calheiros, S. Bianchi, M.A.S. Netto and R. Buyya, 2015. Big data computing and clouds: Trends and future directions. J. Parallel Distrib. Comput., 79-80: 3-15.

Chang, V. and M. Ramachandran, 2016. Towards achieving data security with the cloud computing adoption framework. IEEE. Trans. Serv. Comput., 9: 138-151.

Chen, M., S. Mao and Y. Liu, 2014. Big data: A survey. Mobile Networks Applic., 19: 171-209.

Chu, C.K., S.S. Chow, W.G. Tzeng, J. Zhou and R.H. Deng, 2014. Key-aggregate cryptosystem for scalable data sharing in cloud storage. IEEE. Trans. Parallel Distrib. Syst., 25: 468-477.

Daniel, E. and N.A. Vasanthi, 2016. An efficient continuous auditing methodology for outsourced data storage in cloud computing. Proceedings of the 2015 International Conference on Computational Intelligence, Cyber Security and Computational Models, December 19, 2015, Springer, Singapore, ISBN:978-981-10-0250-2, pp: 461-468.

Fernandez, A., S.D. Rio, V. Lopez, A. Bawakid and M.J.D. Jesus *et al*., 2014. Big data with cloud computing: An insight on the computing environment, MapReduce and programming frameworks. Interdiscip. Rev. Data Min. Knowl. Discovery, 4: 380-409.

Hashem, I.A.T., I. Yaqoob, N.B. Anuar, S. Mokhtar and A. Gani *et al*., 2015. The rise of big data on cloud computing: Review and open research issues. Inf. Syst., 47: 98-115.

Hongbing, C., R. Chunming, H. Kai, W. Weihong and L. Yanyan, 2015. Secure big data storage and sharing scheme for cloud tenants. China Commun., 12: 106-115.

Inukollu, V.N., S. Arsi and S.R. Ravuri, 2014. Security issues associated with big data in cloud computing. Intl. J. Network Secur. Appl., 6: 45-56.

Jain, T. and A.P.J.A. Khan, 2017. Secure big data access control policies for cloud computing environment. Intl. J. Innovative Res. Comput. Sci. Technol., 5: 253-256.

Jin, X., B.W. Wah, X. Cheng and Y. Wang, 2015. Significance and challenges of big data research. Big Data Res., 2: 59-64.

Katal, A., M. Wazid and R.H. Goudar, 2013. Big data: Issues, challenges, tools and good practices. Proceedings of the 6th International Conference on Contemporary Computing (IC3) 2013, August 8-10, 2013, IEEE, Dehradun, India, ISBN:978-1-4799-0191-3, pp: 404-409.

Kelbert, F., F. Gregor, R. Pires, S. Kopsell and M. Pasin *et al*., 2017. SecureCloud: Secure big data processing in untrusted clouds. Proceedings of the Conference on Design, Automation and Test in Europe, March 27-31, 2017, European Design and Automation Association, Lausanne, Switzerland, pp: 282-285.

Li, K., W. Zhang, C. Yang and N. Yu, 2015. Security analysis on one-to-many order preserving encryption-based cloud data search. IEEE. Trans. Inf. Forensics Secur., 10: 1918-1926.

Liu, J.K., K. Liang, W. Susilo, J. Liu and Y. Xiang, 2016. Two-factor data security protection mechanism for cloud storage system. IEEE. Trans. Comput., 65: 1992-2004.

Maheswari, M.I., S. Revathy and R. Tamilarasi, 2016. Secure data transmission for multisharingin big data storage. Indian J. Sci. Technol., 9: 1-6.

Prakash, G.L., M. Prateek and I. Singh, 2014. Efficient data security method to control data in cloud storage system using cryptographic techniques. Proceedings of the 2014 Conference on Recent Advances and Innovations in Engineering (ICRAIE'14), May 9-11, 2014, IEEE, Jaipur, India, ISBN:978-1-4799-4041-7, pp: 1-6.

Puthal, D., S. Nepal, R. Ranjan and J. Chen, 2015. DPBSV-an efficient and secure scheme for big sensing data stream. Proceedings of the 2015 IEEE Conference on Trustcom/BigDataSE/ISPA Vol. 1, August 20-22, 2015, IEEE, Helsinki, Finland, ISBN:978-1-4673-7951-9, pp: 246-253.

Rizvi, S., A. Razaque and K. Cover, 2015. Third-Party Auditor (TPA): A potential solution for securing a cloud environment. Proceedings of the 2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing (CSCloud'15), November 3-5, 2015, IEEE, New York, USA., ISBN: 978-1-4673-9299-0, pp: 31-36.

Sharma, P.P. and C.P. Navdeti, 2014. Securing big data hadoop: A review of security issues, threats and solution. Intl. J. Comput. Sci. Inf. Technol., 5: 2126-2131.

Shimbre, N. and P. Deshpande, 2015. Enhancing distributed data storage security for cloud computing using TPA and AES algorithm. Proceedings of the 2015 International Conference on Computing Communication Control and Automation (ICCUBEA'15), February 26-27, 2015, IEEE, Pune, India, ISBN: 978-1-4799-6892-3, pp: 35-39.

Sookhak, M., A. Gani, M.K. Khan and R. Buyya, 2017. Dynamic remote data auditing for securing big data storage in cloud computing. Inf. Sci., 380: 101-116.

Tahilyani, M., A. Dutta and D. Parwani, 2015. Cloud computing storage security and various protocols: A survey. Intl. J. Adv. Eng. Technol., 8: 762-769.

Terzi, D.S., R. Terzi and S. Sagiroglu, 2015. A survey on security and privacy issues in big data. Proceedings of the 10th International Conference on Internet Technology and Secured Transactions (ICITST'15), December 14-16, 2015, IEEE, London, England, UK., ISBN: 978-1-9083-2051-3, pp: 202-207.

Wei, L., H. Zhu, Z. Cao, X. Dong and W. Jia *et al*., 2014. Security and privacy for storage and computation in cloud computing. Inf. Sci., 258: 371-386.

Wu, X., R. Jiang and B. Bhargava, 2017. On the security of data access control for multiauthority cloud storage systems. IEEE. Trans. Serv. Comput., 10: 258-272.

Yakoubov, S., V. Gadepally, N. Schear, E. Shen and A. Yerukhimovich, 2014. A survey of cryptographic approaches to securing big-data analytics in the cloud. Proceedings of the 2014 IEEE Conference on High Performance Extreme Computing (HPEC'14), September 9-11, 2014, IEEE, Waltham, Massachusetts, USA., ISBN:978-1-4799-6232-7, pp: 1-6.

Youssef, A.E., 2014. A framework for secure healthcare systems based on big data analytics in mobile cloud computing environments. Intl. J. Ambient Syst. Appl., 2: 1-11.

Zhu, Y., G.J. Ahn, H. Hu, S.S. Yau and H.G. An *et al*., 2013. Dynamic audit services for outsourced storages in clouds. IEEE. Trans. Serv. Comput., 6: 227-238.