Received: January 6, 2018


International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

# Enhanced-Elliptic Curve Diffie Hellman Algorithm for Secure Data Storage in Multi Cloud Environment

**Anitha Patil**[1*]

[1]*Pillai HOC College of Engineering and Technology, India*
* Corresponding author's Email: panitha243@gmail.com

**Abstract:** Cloud computing service is one of the most emerging research area in the field of cloud environment. It also emerged as a new platform for managing, deploying and provisioning large scale data. However, the access of user in cloud node or data storage point is restricted to a certain condition. Whereas, several encryption methodologies are utilized in the cloud for improving the security and also to reduce the computation time, but most of the methodologies requires high processing unit. To overcome this concern, an effective cloud storage system is developed in this research paper. Here, an Enhanced-Elliptic Curve Diffie Hellman (E-ECDH) approach is utilized for encrypting and decrypting the data with low computational time. The E-ECDH method generates the key without any complex program that helps in limited use of resources. The encryption and decryption time of the E-ECDH is decreased by using the less complex value. Finally, the experimental outcome showed that the proposed approach improved the security of the cloud system up to 0.11-0.03% of success rate compared to the other existing methodologies.

**Keywords:** Cloud computing service, Cloud security, Cloud storage model, Enhanced-elliptic curve diffie hellman, Success rate.

## 1. Introduction

Cloud computing is the delivery of host services over the internet for accessing the data. Cloud computing services are generally classified into three phases: public cloud, private cloud and hybrid cloud [1, 2]. The development of the cloud database enables the enterprises to enjoy an elastic and effective cloud storage for their private data and also takes the advantage of cloud computing for storing and processing the large amount of data [3]. Many companies provide cloud services such as google, amazon, e-commerce companies like e-bay and social medias like Facebook, Twitter etc. In recent decades, several organizations tend to store their database in the cloud, which increases the demand of security and availability of the data [4, 5]. The current methodologies used in cloud does not render the both demands for the users, which causes increased threat for the private data of an organization like misusing of data by competitor organizations, various attack used by hackers to obtain the data [6]. To overcome these concerns, researches used multi cloud approaches instead of single cloud method. The multi-cloud method is the process of using several cloud computing services in a single heterogeneous construction [7, 8].

Multi-cloud providers prevent the loss of data, due to localized component failure in the cloud. Cloud storage offers a huge amount of space to store the user data that helps to reduce the cost of server maintenance [9, 10]. In this experimental research, Enhanced-ECDH methodology is utilized in multi cloud computing services for encrypting and decrypting the data using TPC-H benchmark dataset. E-ECDH is one of the public key cryptography system, which is used to secure the data transmission. The E-ECDH methodology reduces the encryption and decryption time of the data by developing two keys namely public and private key. The public key is visible to the user that is utilized for encrypting the system and the private key is used for hiding the data, which is used for decryption process. The proposed method increases the speed of the encryption and decryption process and also reduces the

computational overheads compared to the existing approaches. Whereas, the proposed system utilizes very low resources for security development and the remaining resources are used for other computational purpose.

This paper is composed as follows. Section 2 presents a broad survey of several recent papers on cloud security strategies. In section 3, E-ECDH methodology is presented with an effective cloud storage model. In Section 4, comparative analysis of proposed (E-ECDH) and existing methodology is presented. The conclusion is made in Section 5.

## 2. Literature review

Several techniques are suggested by researchers in the cloud security system. In this scenario, a brief evaluation of some important contributions to the existing literatures are presented.

F. Amato, F. Moscato, V. Moscato, and F. Colace, [11] developed a cope model driven engineering methodologies for monitoring cloud infrastructures and security analysis. This literature provides a formal profile of host's thermal behaviours for reducing overhead. Then, forecast and identify malicious actions by relating with real time data based on services input workloads. The experimental outcome confirmed that the proposed methodology was more significant than existing approaches by means of work load. In a few cases, cope models need data owner for computation to perform encryption and decryption process.

T. Halabi, and M. Bellaiche, [12] proposed an effective methodology named as goal-question-metric paradigm for performing valuation and quantification of cloud security services based on a set of performance metrics. In this literature, practicability and the efficiency of the proposed approach was evaluated by means of computational time. Also, compared to other existing methodologies, the proposed scheme has low computational time. The proposed methodology was only applicable for horizontal dataset, not for all the databases.

V.P. Binu, and A. Sreekumar, [13] developed a methodology for secret sharing, which was the combination of shamir's scheme and elliptic curve pairing. In this proposed approach, the consistency of the shares was chosen by the participants in an ensured manner. In reconstruction phase, the combiner shares the participant's data to verify the cheaters using bilinear pairing. The proposed scheme reduces the verifiability problem, because of participant's share. The proposed approach has a concern of secure key storage in a large data storage.

H.I. Kim, H.J. Kim, and J.W. Chang [14] proposed a new secure query processing algorithm in the cloud data named as K-Nearest Neighbour (KNN). This methodology was designed to protect the both user data and query records. In addition, the index approach was utilized to increase the efficiency of the system without accessing the important data. The experimental result shows that the proposed approach reduces the query processing cost than the existing methodologies along with the privacy preserving of data. This literature does not focus on computational time, which was considered as one of the major concern in cloud security.

M. Ahmadian, F. Plochan, Z. Roessler, and D.C. Marinescu [15] illustrated a secure proxy service to carry out transformation for cloud server modification. This technique was applicable to all the NoSQL data model and also it was applied to the document-store data model. The proposed method was designed to the descriptive language based on a subset of JSON notations and a tool to create and analyse the security plans over a cryptographic model with query and data validation. This technique was very useful for NoSQL database query processing, but it was not applicable for big data analysis.

To overcome the above mentioned drawbacks, an effective cloud storage model is implemented with enhanced-ECDH algorithm for improving the security of the cloud system.

## 3. Proposed methodology

The general architecture of multi-cloud servicing is represented in the Fig. 1. The multi cloud servicing consists of three parts such as users, Data Owner (DO) and Cloud Database (CDB).

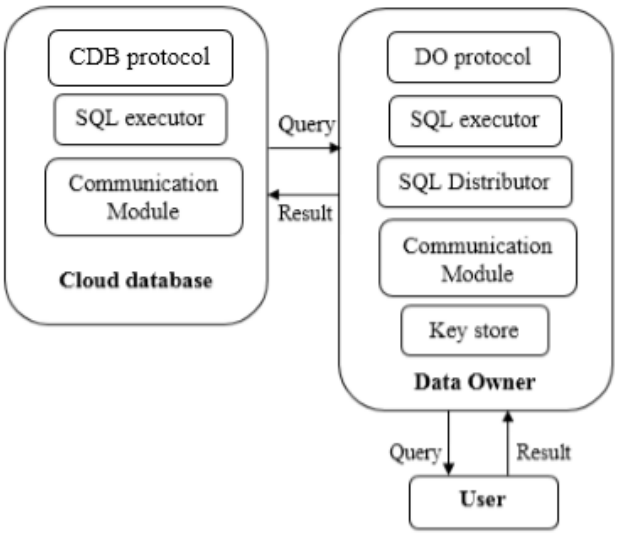


Figure.1 General structure of multi cloud service

### 3.1 Data owner and cloud database

Data owner is an individual or enterprise that consists of enormous private data. Data owner's servers are low configuration, which contains only limited storage space, because most of the storage requirements and computations are transferred into cloud side. The communication module is designed to connect with all cloud databases, which delivers a unified database management for data owners.

If the user submits query request, the SQL analyser analyses the queries and categorize the query into two classes such as read/write queries and read only queries. The SQL distributor is employed for choosing the most suitable cloud for load balancing methodologies to do the query. While modifying the query request, database management system synchronize all the database utilized in the query processing. Each CBD protocols comes from a dissimilar service provider. If it receives the queries, the SQL executor accomplishes the queries with the received key and carry out pre-designed protocol to determine the result.

### 3.2 Encryption and decryption process

In this scenario, the encryption and decryption process is performed by using enhanced-ECDH method. Initially, ECDH is a fully homomorphic encryption scheme that contains a pair of keys namely public and private key for encrypting and decrypting the data within an insecure channel. Message sharing is directly derived by using public key and the decryption process is done using private key. The derived keys are used as the key for successive data transactions that happens between the committed parties in the channel. The work flow of ECDH scheme is executed in the following steps for transaction of data between the sender $S$ and receiver $R$.

Initially, all the elliptic curve parameters are generated. In the next step, each party should choose a pair of keys, one is the private key $d$ , a random unique point chosen in the curve and another one is public key, which is derived from Eq. (1).

$$Q = dG \tag{1}$$

Where, $G$ is the generator of curve.

Let, the sender key is $(d_A, Q_A)$ and the key of receiver is denoted as $(d_B, Q_B)$. Whereas, the public key $Q$ is shared with others during communication, based on that only a person can able to decrypt the message sent by the sender. While transmission of a message in the ECDH system, a message or data denoted as a point in the elliptic curve $(x, y)$. The point $(x, y)$ is calculated by the receiver and decrypt the message via the product $Q_B d_A$ or $Q_A d_B$ .The ECDH encrypted messages always possess the symmetric property that is denoted in Eq. (2).

$$Q_B = d_A d_B G = d_B d_A G = d_B G \tag{2}$$

The ECDH encrypts the data efficiently with symmetrical encryption property, but it is more difficult to chosen plain text and cipher text in attacks. Its larger key size is another major issue in encryption and decryption time. To resolve these issues, improve the speed of encryption and decryption using Enhanced-ECDH that is explained below.

#### 3.2.1. Enhanced-Elliptic Curve Diffie-Hellman

The limitations that found in the ECDH is solved with the help of integrating the encryption and hashing scheme directly into the elliptic curve module. By reducing the key size, the time taken for encryption and decryption is lessened and the communication overhead is also reduced. Enhanced-ECDH is an improved encryption scheme, which is capable of provisioning semantic privacy confidentiality over external attacks like cipher text attacks and plain text attacks. The performance of Enhanced-ECDH algorithm is described below.

In Enhanced-ECDH, the sender learns receiver's public key $g^x$ , where $x$ is the private key of the receiver. Then, sender generates a new epithermal value $y$ and its associated value $g^y$. After generating epithermal value, sender calculates the symmetric key $k$ with the help of Key Generation Function (KGF), which is denoted in Eq. (3).

$$k = KGF(g^{xy}) \tag{3}$$

Now, the sender encrypts the data with the help of $k$ to generate the cipher text $c$ of the message $c = E\ (k: message)$. Then, the sender transmits the both cipher text and public key $(c; g^y)$. If the receiver has both $x$ and $g^y$ values, it can able to decrypt the cipher text and retrieve the original message.

The Enhanced-ECDH algorithm encrypts the data with fewer complexities and lesser time, which also provides a secured hashing method to withstand external attacks. The Enhanced-ECDH algorithm is integrated with the key generation parameters for reducing the communication cost or communication overhead. The encryption and decryption using Enhanced-ECDH is detailed as follows.

#### 3.2.2. Encryption process

To encrypt the message, the proposed method (E-ECDH) performs the following steps.

Manipulate a random integer $r$ within the interval $r \in [1, n-1]$ and it compute $R = rG$, where $G$ is the generator of elliptic curve. Then, E-ECDH computes the shared secret $sec = pub_x$, where $pub = (pub_x, pub_y) = rK_R$, where $K_R$ is the public key of receiver $R$ and $pub \neq \emptyset$. Next, derive the symmetric encryption keys and Message Authentication Code (MAC) keys using Eq. (4).

$$k_E \parallel k_M = KGF(S \parallel S_1) \tag{4}$$

Encrypt the message with the help of encryption key, which is represented in Eq. (5).

$$K_E c = encryption\ (K_E; message) \tag{5}$$

Then, calculates the MAC of the encrypted message $S_2 : d = MAC(k_M; c \parallel S_2)$. Finally, E-ECDH sends the cipher text output $c$ to the receiver, which encrypts with the help of private key $d$, it is represented as $R \parallel c \parallel d$.

#### 3.2.3. Decryption process

In order to decrypt the cipher text message the receiver $R$ performs the following steps.

Initially, the receiver $R$ derives the shared secret $S = P_{ubx}$, where $pub = (pub_x, pub_y) = K_R R = rK_R G$, $K_R$ is the public key of receiver and output fails if $pub = \emptyset$.

Then, computes the decryption key from $k_E \parallel k_M = KGF(S \parallel S_1)$. After computing the key, E-ECDH calculates the MAC to verify the tag, whether output fails or not $d \neq MAC(k_M; c \parallel S_2)$.Finally, E-ECDH utilizes the symmetric encryption key to decrypt the original message $m = E^{-1}(k_E : c)$.

### 3.3 Mutli cloud architecture

E-ECDH requires at least two different cloud service providers, it seems that the total investment will get increase, due to extra costs of cloud resources. Commonly, user will collect the query result from cloud server, E-ECDH ensures that each and every query is executed with a load balancing methodology, so computation resources of each cloud is utilized well. The graphical representation of load balancing architecture in database is denoted in the Fig. 2. E-ECDH reduces the bandwidth requirement of each and every cloud. Also, E-ECDH implements the data

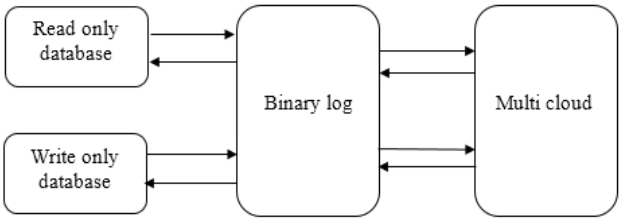


Figure.2 Loading balancing architecture in database

integrity protection for multi-cloud architecture. It has a complete data encryption solution, which does not require any other chargeable secure services from vendors to encrypt the data. To explain the proposed multi cloud architecture, an effective approach E-ECDH is developed for several cloud servers from different vendors.

After developing the load balancing algorithm, the cloud database is categorized into two phases: read only database and write only database. If the database is small there is no need for classification. In case, it is a large database, classification is necessary. E-ECDH builds own database module using database replication. In multi-cloud service architecture, once the query performed write operation, the respective data is synchronized between CDBs. The proposed E-ECDH approach maintains data consistency between the CDBs. To ensure the synchronization efficiency, SQL queries are applied for synchronizing measures. Two types of synchronization measure are carried out in E-ECDH: row based synchronization and statement based synchronization. If the query is synchronized using statement based synchronization, E-ECDH sacrifice much performance of the cloud. Compared to statement based synchronization, the row based synchronization is very slow in measuring.

## 4. Experimental outcome

In this experimental investigation, cost analysis, execution time and the perceived availability of the proposed methodology are verified. The proposed methodology is designed with C# on the data side and MySQL is installed on the cloud side with protocols written by UDFs. The processor of two Intel Xeon CPU E5-4603 v2@2.20GHZ and 32GB RAM with running Cent OS 6.5 in the 64-bit processor and MySQL 5.5. In order to estimate the efficiency of the proposed technique, the performance of the proposed approach was compared with the famous cloud servers like Crypt-DB, MONOMI, SDB, Amazon RDS and SHAMC [16], because these cloud servers are elastic in nature and also adjusted for any kind of environments. In Table 1, the proposed system is

Table 1. Evaluation results of database workload (for 30 minutes)

| Functions | SHAMC2 [16] | SHAMC3 [16] | SHAMC4 [16] | Crypt -DB [16] | MONOMI [16] | SDB [16] | Amazon RDS [16] | E-ECDH |
|---|---|---|---|---|---|---|---|---|
| Read Queries (107) | 1.1955 | 1.3518 | 1.4212 | 0.7826 | 0.9856 | 0.7521 | 1.3996 | **1.4424** |
| Write Queries (107) | 0.2152 | 0.2549 | 0.2864 | 0.1056 | 0.1685 | 0.1041 | 0.4665 | **0.6908** |
| Transaction (105) | 7.6512 | 7.9691 | 8.1251 | 6.0519 | 6.5849 | 6.0551 | 9.9882 | **10.284** |
| Deadlock | 1265 | 1143 | 1095 | 1521 | 1582 | 1438 | 846 | **795** |
| TPS | 7468 | 7823 | 8512 | 5125 | 5816 | 5049 | 10659 | **10856** |
| **Success Rate (%)** | **99.84** | **99.86** | **99.87** | **99.75** | **99.76** | **99.76** | **99.92** | **99.95** |

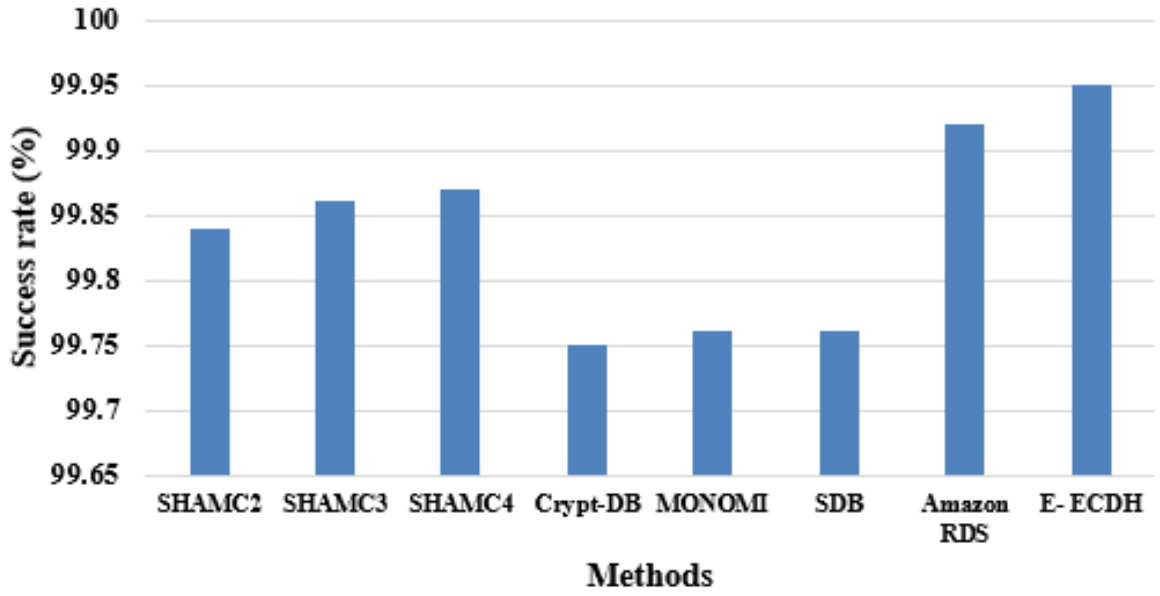


Figure.3 Performance comparison of existing and proposed approach by means of success rate

compared with the existing systems [16] based on workload evaluated for 30 minutes.

The parameters like read queries, write queries, transaction, deadlock, Transition per Second (TPS) are considered for identifying the success rate of the proposed system. The proposed methodology (E-ECDH) is compared with the existing methodologies such as SHAMS2, SHAMS3, SHAMS4, CryptDB, MONOMI, SDB, and Amazon RDS. The Table 1 clearly shows that the proposed methodology (E-ECDH) outperforms all the existing system by means of success rate. The E-ECDH method generates the public and private key in a simpler manner. So, it doesn't consume more resources for the system and it utilizes the most of the resources for executing other tasks like read queries, write queries, transaction, deadlock, TPS, etc. So, the success rate of the proposed system automatically gets increased. The E-ECDH method is also utilized to encrypt the data before upload it to the cloud. The graphical comparison of existing and proposed methodologies by means of success rate is shown in the Fig. 3.

The success rate measures the successful transmission of the proposed methodology. The deadlock may occur in the request of queries to the cloud, which is common in the large database that cannot be fully avoided. Fig. 3 clearly shows that the proposed system has higher transmission than the existing systems. The amazon RDS [16] is the second highest transmission rate with higher success rate compared to other cloud servers.

The proposed E-ECDH system outperforms all the existing cloud servers by means of success rate, due to its design and multi cloud architecture. The performance of the system is improved by increasing the number of cloud in the methodology. Development cost is the second most important factor in system development and product cost. Evaluate the typical scenario of the enterprise 1TB hard disk, 24 GB of RAM, 80 Mbps bandwidth and 8-core processor for investigating the cost of the proposed methodology. In Table 2, the query based cost comparison is performed between SHAMC2 and E-ECDH. Whereas, the query numbers 1, 8, 16 are collected from the database TPC-H benchmark.

Table 2. Query based cost comparison

| Methodology | Query number | DO protocol cost | Transmission | CDB protocol cost |
|---|---|---|---|---|
| SHAMC2 [16] | Q1 | 16 | 21 | 17 |
| | Q8 | 12 | 28 | 32 |
| | Q16 | 25 | 27 | 31 |
| E-ECDH | Q1 | 6 | 12 | 14 |
| | Q8 | 4 | 9 | 17 |
| | Q16 | 8 | 15 | 23 |

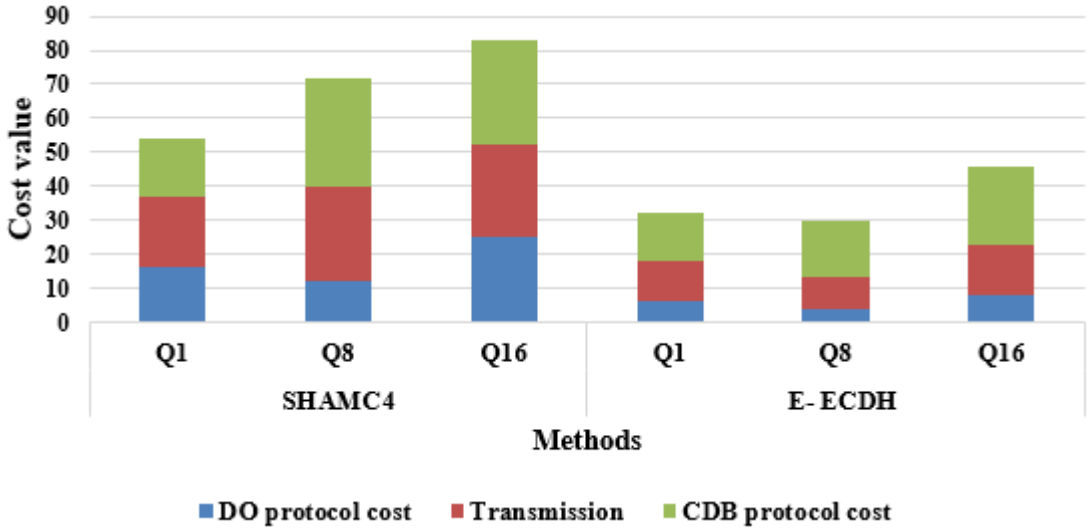


Figure.4 Query based cost comparison of existing and proposed methods

Table 3. Execution time comparison for both the existing and proposed approach

| Methods | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **Crypt DB [16]** | **32** | **29** | **21** | **23** | **12** | 17 | 15 | 11 | 13 | 17 |
| **MONOMI [16]** | **6** | **13** | **11** | **10** | **9** | 14 | 8 | 7 | 8 | 13 |
| **SHAMC [16]** | **5** | **6** | **4** | **8** | **5** | 7 | 7 | 8 | 10 | 7 |
| **E-ECDH** | **2** | **3** | **1** | **2** | **1** | **1** | **2** | **4** | **4** | **6** |

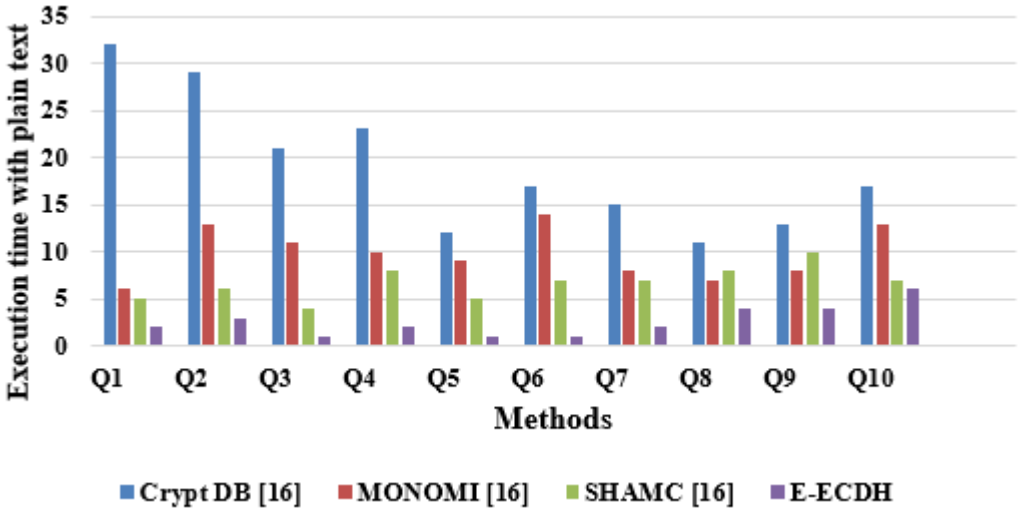


Figure.5 Comparative analysis based on execution time

The proposed approach processes the cloud data for encrypting and decrypting with less computational time. The TPC-H benchmark plain text database is evaluated with the both existing and proposed systems, then the execution time is calculated for each query. The graphical representation of E-ECDH process and their execution time with the threat of plain text attack is given in the Fig. 5 and Table 3.

## 5. Conclusion

Cloud computing is a distributed computing infrastructure, which has the ability to render a variety of internet based services for the customers. Due to several benefits, the cloud computing infrastructure has gained more popularity in recent years. The application of cloud computing is increased in commercial fields like online application and data storage. In this experimental research, E-ECDH methodology is utilized in the cloud for encrypting the data with less computational time. The proposed approach reduces the encryption and decryption time of the data by developing two keys namely public and private key. The public key is visible to the user, which is utilized for encrypting the system, whereas the private key is hidden that is utilized for decryption process. Associated to the other obtainable approaches in cloud security, the advanced scheme delivered an effective performance by means of execution time, cost and success rate. Also, the proposed approach achieves around 0.11-0.03% of enhancement in success rate than the existing methods (Crypt-DB, MONOMI, SDB, Amazon RDS and SHAMC). In the future work, develop an appropriate cloud storage model for further reducing the computational time and also withstand threats to secure the data with more confidentiality.

## References

[1] F. Sabahi, "Secure virtualization for cloud environment using hypervisor-based technology", *International Journal of Machine Learning and Computing*, Vol.2, No.1, pp.39, 2012.

[2] G. Di Modica and O. Tomarchio, "Matchmaking semantic security policies in heterogeneous clouds", *Future Generation Computer Systems*, Vol.55, pp.176-185, 2016.

[3] K. Feng and J. Zhang, "Improving availability and confidentiality of shared data under the multi-cloud environment", In: *Proc. of 2nd International Conf. On* Cloud Computing and Big Data Analysis (ICCCBDA), pp.6-10, 2017.

[4] J. Li, D. Lin, A.C. Squicciarini, J. Li, and C. Jia, "Towards privacy-preserving storage and retrieval in multiple clouds", *IEEE Transactions on Cloud Computing*, Vol.5, No.3, pp.499-509, 2017.

[5] Q.G.K. Safi, S. Luo, C. Wei, L. Pan, and G. Yan, "Cloud-based security and privacy-aware information dissemination over ubiquitous VANETs", *Computer Standards & Interfaces*, Vol.56, pp.107-115, 2017.

[6] T. Xiang, X. Li, F. Chen, S. Guo, and Y. Yang, "Processing secure, verifiable and efficient SQL over outsourced database", *Information Sciences*, Vol.348, pp.163-178, 2016.

[7] K. Kritikos, T. Kirkham, B. Kryza, and P. Massonet, "Towards a security-enhanced PaaS platform for multi-cloud applications", *Future Generation Computer Systems*, Vol.67, pp.206-226, 2017.

[8] T. Zhang and R.B. Lee, "Monitoring and Attestation of Virtual Machine Security Health in Cloud Computing", *IEEE Micro.*, Vol.36, No.5, pp.28-37, 2016.

[9] S. Lallali, N. Anciaux, I.S. Popa, and P. Pucheral, "Supporting Secure Keyword Search in the Personal Cloud", *Information Systems*, Vol.72, pp.1-26, 2017.

[10] Q. Huang, Y. Yang, and M. Shen, "Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing", *Future Generation Computer Systems*, Vol.72, pp.239-249, 2017.

[11] F. Amato, F. Moscato, V. Moscato, and F. Colace, "Improving security in cloud by formal modelling of IaaS resources", *Future Generation Computer Systems*, In Press, 2017.

[12] T. Halabi and M. Bellaiche, "Towards quantification and evaluation of security of Cloud Service Providers", *Journal of Information Security and Applications*, Vol.33, pp.55-65, 2017.

[13] V.P. Binu and A. Sreekumar, "Secure and Efficient Secret Sharing Scheme with General Access Structures Based on Elliptic Curve and Pairing", *Wireless Personal Communications*, Vol.92, No.4, pp.1531-1543, 2017.

[14] H.I. Kim, H.J. Kim, and J.W. Chang, "A secure kNN query processing algorithm using homomorphic encryption on outsourced database", *Data & Knowledge Engineering*, In Press, 2017.

[15] M. Ahmadian, F. Plochan, Z. Roessler, and D.C. Marinescu, "Secure NoSQL: An approach for secure search of encrypted nosql databases in the public cloud", *International Journal of*

*Information Management*, Vol.37, No.2, pp.63-74, 2017.
[16] L. Wang, Z. Yang and X. Song, "SHAMC: A Secure and highly available database system in multi-cloud environment", *Future Generation Computer Systems*, In Press, 2017.