Cyber Security and Applications 2 (2024) 100027

Contents lists available at ScienceDirect

Cyber Security and Applications

journal homepage: http://www.keaipublishing.com/en/journals/cyber-security-and-applications/

Efficient time-oriented latency-based secure data encryption for cloud storage

Shahnawaz Ahmad[*], Shabana Mehfuz

Department of Electrical Engineering, Jamia Millia Islamia (A Central University), New Delhi, 110025, India

ARTICLE INFO

Keywords:
Cloud security
Data encryption
Data security
Latency approximation
TLADE
QoSV
QoS

ABSTRACT

To ensure data security in the cloud, there exist several techniques proposed by various researchers. The most common method is the usage of data encryption techniques like profile, rule, and attribute-based encryption techniques. However, they struggle in achieving higher data security performance due to insufficient resistance to tampering. Also, the existing techniques are not good enough for achieving a higher quality of service performance. To handle this issue, an efficient time-oriented latency approximation-based data encryption technique (TLADE) has been presented in this article. The method focuses on the selection of optimal encryption techniques at different time stamps according to the latency approximation. Accordingly, the method would select an optimal technique for data encryption based on the QoS values. To perform this, different encryption schemes have been implemented and each of them is measured for their QoS support values (QoSV) based on latency. Based on the values of QoSV, an efficient approach for the current duty cycle has been selected and applied to the cloud service data. The proposed approach improves the performance of different QoS factors and also has reduced the latency factor.

Introduction

The cloud environment provides different services at various levels to support the organizational process. The cost-effectiveness of cloud services has helped organizations in maintaining their data at the least cost. Unlike traditional data management, organizations do not need to maintain their data servers which claim higher costs from the organizations which cannot be offered by all the units [1]. To support such organizations in maintaining their data at the least cost, the cloud service providers provide several services at various levels like presentation, network, and data layers. This has encouraged organizations to maintain their data at the least cost while allowing users to access his/her data through different services.

Like any other network and environment, the cloud faces several threats at various levels. The security threats could be directed toward data as well as services. Various types of cyber-attacks, such as Distributed Denial of Service (DDoS) attacks and brute force attacks, specifically aim at disrupting the functioning of services. However, all the threats target the service functionality and are targeted to degrade the service performance. If the service is targeted by a DDoS attack, then the malicious user would generate several service requests to the service provider which generates traffic at the service point. Also, the malicious user would keep the service handled without any data transfer or would place malformed data on the service. Both approaches would degrade the service throughput and affect the performance of the entire environment [2]. To handle this issue, there exist numerous approaches which monitor the service request and allow the service access based on the profile, trust, and so on. The trust-based approaches are becoming popular in this case, which maintains the traces of service access belonging to different users. According to the traces available, the method measures the trust of the users. The value of trust can be measured using service access behaviors, service protocol being followed, frequency of service access, completeness of service access, and so on. Still, the performance of access restriction schemes is not addressable. So, by enforcing a rigid security algorithm at the service level, the security performance of the model can be improved [3].

In terms of data security, it has been a huge threat to the cloud systems as the malicious user can post malicious data to the service which in turn would end up in failure and affect the performance [4–5]. Similarly, if the adversary is capable of fetching the service packets in the network, they can modify, or steal the data from the service packets. Both of these activities in turn would adversely affect the service performance. To address this issue there exist different data encryption schemes that use various encryption standards. All these approaches lack higher performance metrics in data security [6].

In this way, the service packets are transmitted through several routes and pass-through various routers, hubs, and so on. If there exists a malicious node, then it would involve different routing attacks to

[*] Corresponding author.
E-mail address: shahnawaz98976@gmail.com (S. Ahmad).

https://doi.org/10.1016/j.csa.2023.100027
Received 30 March 2023; Received in revised form 22 June 2023; Accepted 30 June 2023
Available online 30 June 2023

maximize the delay, minimize the throughput, and so on. However, the performance of the entire system depends on various factors like the selection of a routing scheme, the selection of an encryption scheme, and so on. Choosing an efficient data encryption scheme and routing scheme would support the performance achievement of various QoS parameters. This work is focused on improving the QoS of the environment by choosing an efficient latency-centric data encryption scheme. Any data encryption scheme takes a specific amount of time. Therefore, by choosing the least latency data encryption scheme, the performance of the entire system can be improved.

QoS of any environment is greatly dependent on how well the security performance is and how effective trust verification has been enforced [6]. By considering all this, an efficient TLADE is presented in this paper. The method is focused on the selection of efficient data encryptions scheme and route selection in such a way as to reduce the latency. The method performs latency approximation on the working of the data encryption scheme and routing. For each of them, QoSV is measured to perform an efficient selection of routes as well as security schemes. Both methods are focused on improving the QoS of the complete environment. The detailed approach has been discussed in this section.

The contributions of this paper are as follows:

- To enhance the Quality of Service (QoS) in the environment by selecting a data encryption scheme that prioritizes low latency.
- Choosing the most suitable encryption techniques at various time intervals based on an estimated latency.
- An effective method has been chosen and implemented to handle the cloud service data based on the QoSV values for the current operational cycle.
- To enhance the performance of various QoS aspects while simultaneously reducing latency.

The rest of the paper is structured as follows: Section 2 provides an overview of the relevant previous research. Section 3 introduces the Time Oriented Latency Approximation Based Data Encryption Technique. Section 4 presents the results and accompanying discussion. Section 5 elaborates on the findings, followed by the paper's conclusion.

Related previous work

The methods proposed by various researchers regarding the problem have been discussed in this section for the identification of the problem and understanding. The literature review which has been carried out in tabulated form (as shown in Table 1) provides a meta-analysis of available work and has established a ground for the proposed work.

A Markov decision-based SLA violation detection scheme has been presented in [7], which uses the requirements of various users in regulating the actions. In [8], an encryption scheme based on Non-Orthogonal Multiple Access (NOMA) and homomorphic encryption is introduced. This scheme incorporates a channel coding mechanism to address the presence of malicious nodes. In [9], a public key encryption scheme called Probabilistic Public Key Encryption (EPPKE) is introduced. This scheme leverages Covariance Matrix Adaptation Evolution Strategies (CMA-ES) to enhance data security. Furthermore, reference [10] presents a scheme called Revocable Storage Identity-Based Encryption (RS-IBE), which is designed to optimize data security. A Dynamic Data Encryption Strategy (D2ES) has been presented in [11], which performs selective encryption of data with the use of classified privacy.

Dynamic updatable searchable encryption cloud storage (DUSECS) scheme has been presented in [12], which uses characteristics of homomorphic encryption and uses the structure of a linked list. A chaotic fuzzy transformation method has been discussed in [13], which uses keyword indexing with fuzzy logic and ranked results are populated for mobile users. A time and attribute-based dual access control have been presented in [14], which performs attribute-based encryption with hierarchical identity-based encryption schemes. An encrypted data-sharing scheme has been presented in [15], which enables dynamic data sharing without the use of public keys and applies proxy re-encryption technology to maximize data security. In [16], a combined approach has been presented toward privacy preservation, which uses the Lanczos algorithm and the Nyström algorithm with additive homomorphic encryption. A user-side encrypted file system is presented in [17], which performs transparent encryption with structure-based information. A multi-Authority vector policy (MAVP) is presented in [18], which uses various access policies in the form of a matrix and uses a multi-authority vector policy function encryption scheme (MAVP-FE). In [19], an encryption transformation called Identity-Based Encryption Transformation (IBET) is introduced. This transformation combines Identity-Based Encryption (IBE) and Identity-Based Broadcast Encryption (IBBE) schemes with the goal of enhancing data security.

In [20], a public key searchable encryption scheme with forward security is introduced. This scheme utilizes a public key to enhance security. In order to safeguard medical data, reference [21] presents a Secure and Efficient Dynamic Searchable Symmetric Encryption (SEDSSE) scheme. This scheme aims to establish a dynamic searchable symmetric encryption system for the purpose of preserving privacy. A discrete wavelet transform-based selective encryption scheme has been presented in [22], which divides the data into three fragments and each has been enforced with different protection levels. In [23], a rekeying-aware encrypted deduplication scheme (RAEDS) has been presented, which handles various attacks by applying a convergent all-or-nothing transform (CAONT) on the data provided which challenges the adversaries. Reference [24] introduces a revocable storage ciphertext-policy attribute-based encryption scheme known as RS-CPABE-ASP. This scheme employs an arithmetic span program (ASP) access structure to address the challenges associated with policy maintenance costs. By leveraging the structure, access restrictions are enforced. On the other hand, in reference [25], a privacy-preserving aggregation scheme is presented. This scheme utilizes the Simulated Annealing Module Partition (SAMP) algorithm to divide the data into multiple blocks and applies the Differential Aggregation Encryption (DAE) scheme for encryption.

The state of the art in KMS security has primarily focused on traditional security mechanisms, leaving a gap for integrated technological solutions to handle complex cybersecurity issues. In this context, our research addresses a pressing need and contributes to the state of the art in KMS security by proposing a novel approach. By leveraging advanced cryptographic techniques, machine learning, deep learning, and IoT technologies, our framework enhances KMS security [47–49], filling this critical gap in the literature. This unique amalgamation of technologies promises a robust defense against evolving cyber threats, thereby paving the way for a more secure digital world.

The methods discussed in the literature have been analyzed and have been found to have poor performance. This is the motivation behind this work of designing an optimal efficient approach towards achieving higher QoS performance as compared to other approaches.

Time-oriented latency approximation-based data encryption: a proposed methodology

The proposed time-oriented latency approximation-based data encryption scheme has been remarked for and destination initially where the data is available. The method first determines the set of available routes to reach the desired source location or the location where the data needs to be delivered. The method performs route discovery which collects traffic data in different hops and routers present between the source and the service point. Such data has been maintained in the server and using the information, the method finds the routes available to perform latency approximation. Similarly, according to the service requested, the method finds the set of data encryption schemes available and their time complexity in encrypting the data. Using both of them, the method performs latency approximation to select an optimal route and an optimal encryption scheme to perform data transmission.

Table 1
Analysis of related work.

Reference	Focus	Description
Shanmuga Priya, et al. [26]	Authentication Service and Security	The researchers have proposed an enhanced approach for the information security model in cloud computing. In the proposed information security model, user authentication is achieved through the utilization of a group of HMAC-based OTP (One-Time Passwords). The study also includes a performance comparison between the MDS5 and SHA algorithms, aiming to improve the overall efficiency of the system.
Neela, K. L. et al. [27]	Maintaining the confidentiality of data and ensuring its security	The suggested paradigm relies on a decentralized architecture that operates without relying on any third-party scheme. The data security within this architecture can be enhanced through the implementation of the cyclic shift transposition method. To ensure secure data transmission and retrieval and mitigate real-time attacks, the authors utilized a quick response code and a timestamp based on hashing.
Wazid et al. [28]	Authenticated key management protocol	Wazid et al. introduced the AKM-IoV secure authenticated KMP, specifically designed for the placement of Internet of Vehicles (IoV) in fog computing scenarios. Once the IoV transmitting entities successfully authenticate within the configured AKM-IoV, they generate session keys to facilitate secure data transfers.
Miao et al. [29]	Hybrid Keyword-Field Search on encoded data	The researchers have presented a hybrid keyword-field search method for outsourced data, incorporating an efficient key management technique involving a keyed hash tree and a well-suited score function. This method enables simultaneous searching of keywords and fields on encoded data.
Park et al. [30]	key agreement mechanism for V2G in SIoT	The researchers have introduced a dynamic key agreement mechanism for Vehicle-to-Grid (V2G) in the context of the Secure Internet of Things (SIoT). This mechanism ensures user privacy while requiring minimal resources. The proposed protocol provides protection against various types of attacks, including trace attacks and impersonation. It also ensures anonymity, safeguards session key integrity against man-in-the-middle attacks, enables secure mutual authentication with protection against replay attacks and offline password guessing and ensures perfect forward secrecy.
Manish Kumar et al. [31]	ECC-based technique	The objective of this approach was to enhance the efficiency of DNA encoding by employing the proposed ECC algorithm. The RGB image undergoes two encoding stages: first, it is encoded using DNA encoding, and then it undergoes encoding using an asymmetric encryption technique based on the Diffie-Hellman key exchange. In order to assess the efficacy of the proposed method, it is applied to a standard set of test images for evaluation.
R. Balasubramanian et al. [32]	Data Security in Cloud	This study considers key spaces, key sensitivity, and statistical analysis as important factors. The authors have examined the result obtained by multiplying two real-valued multiplicative functions with distinct inputs.
Vijayakumar, V., et al. [33]	A planning-enabled intermediary encryption solution	Vijayakumar, V., et al. introduced an intermediary encryption solution with planning capabilities to mitigate security concerns. Under this approach, authorized individuals will have limited access to the documents for a predetermined duration. The proposed strategy combines searchable encryption and proxy re-encryption techniques to achieve the desired outcomes.
Pallavi et al. [34]	Ensuring the security and efficiency of managing databases that serve multiple tenants	This study introduces a database management system designed for a cloud computing environment, specifically catering to multi-tenancy requirements. Before suggesting a secureness weight metric for tenant worker selection, SEMTDBMS assesses the security requirements of tenant workers. A novel workload scheduler has then been presented for allocating workload among tenants.
Alassafi et al. [35]	A model for determining the security aspects relevant to the adoption of cloud computing	This research offers a significant contribution by investigating the impact of various independent security-related parameters on the selected security taxonomy. The analysis considers critical ratio, standard error, and significance levels, distinguishing it from previous studies. Information was gathered from IT and security professionals working for Saudi Arabian government agencies. For data analysis in [44], the Analysis of Moment Structures (AMOS) tool has been utilized.
Sharma et al. [36]	Optimized load balancing methods designed for multi-datacenter cloud environments	For load balancing and appropriate work scheduling, authors have presented two multi-datacenter two-phase load adjustment algorithms. Proposed approaches distribute user jobs to various virtual machines located in various data centers according to better makespan. Here, authors have taken into account the amount of bandwidth needed for inter-datacenter task transfer as well as communication delay.
Jain et al. [37]	Ensuring the secure exchange of healthcare data through cloud-based solutions	Reference [46] employs the SHA1 hash-based message authentication code and an Elliptic Curve Cryptography (ECC)-based mechanism to securely share sensitive data in the context of robotic healthcare.

The architecture of the proposed TLADE approach has been presented in Fig. 1. It involves several operations and each has been presented in detail in this section.

Preprocessing

The preprocessing process involves two different operations in this approach (as shown by Algorithm 1). First, the access trace set has been processed to eliminate the noise. It has been performed by finding a set of features of the trace and verifying each trace for the presence of all the features with values. If any of the traces have been found incomplete, then it has been removed from the data set. Second, the method finds the service requested and the user's location. Using these two values, a set of routes available to deliver the result has been identified. The routes available are identified and given to the next stage to perform approximation.

The preprocessing algorithm finds the features and performs noise removal from the access trace data set. Additionally, the method identifies and adds the list of routes to reach the source and destination to the set. The features extracted and preprocessed traces have been used to perform latency approximation.

Traffic-Throughput-Latency QoSV estimation

The quality of service of the cloud system depends on traffic and throughput factors. When the traffic is higher it introduces higher latency and affects the throughput performance. So, by choosing a route with the least traffic, the throughput performance can be improved

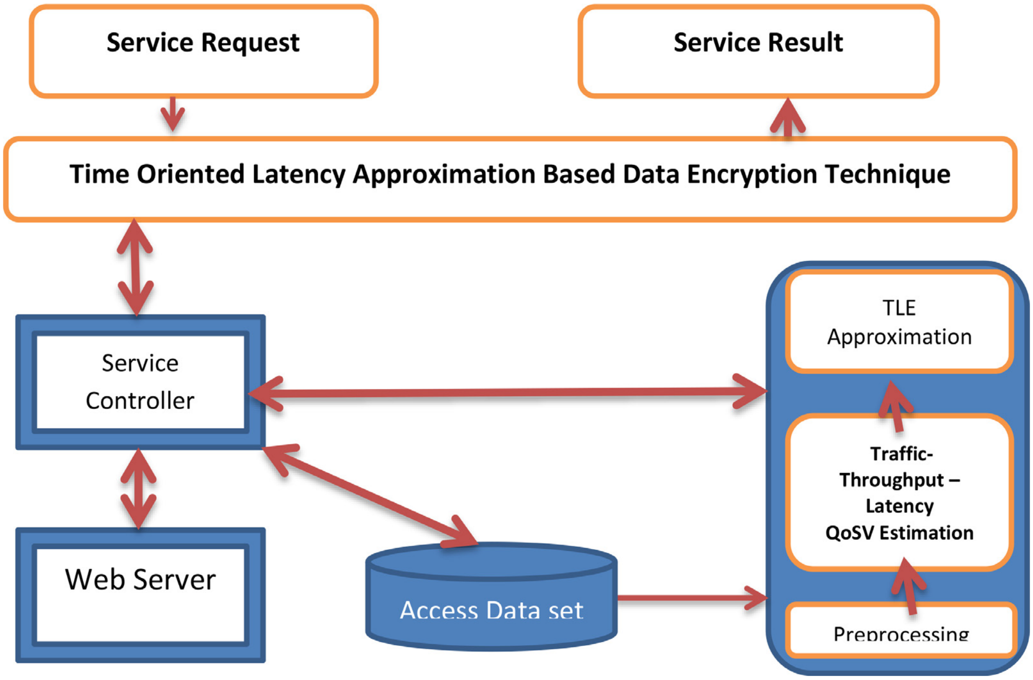
Service Request
Service Result
Time Oriented Latency Approximation Based Data Encryption Technique
Service Controller
Web Server
Access Data set
TLE Approximation
Traffic-Throughput – Latency QoSV Estimation
Preprocessing

Fig. 1. Proposed architecture for the TLADE approach .

Appendix A

Algorithm 1: Preprocessing.

```
Given: Service Access Trace SaT, Service Request SR
Obtain Preprocessed Trace PT, Route List RL.
Start
  Read SaT and SR.
  Fetch a list of features F_list = $\sum_{i=1}^{size(SaT)} SaT(i).Features \ni F\_List$
  For each tuple T
    If T∈ $\forall Features(F_{List})$ then
      For each feature F
                    $size(F_{List})$
        If $T(F_{List(i)})! = Null\ then$
                    $i = 1$
          Complete.
        Else
          Break;
        End
      End
      If Complete then
        PT = $\sum Traces(PT)T$
      End
    End
  End
  Route List Rl = $\sum Routes(S \rightarrow D) \in Network$
Stop
```

which in turn would improve the QoS performance. The value of Traffic QoSV (TQoSV) is measured according to the traffic present on the route in different time stamps and current time stamps. Similarly, the value of throughput QoSV is measured according to the throughput present in a specific route in the various time stamp. Similarly, the latency QoSV is measured based on the average latency in various time stamps and currents. The values estimated have been used in latency approximation.

The equations (i) to (vi) are presented in Algorithm 2 which calculates the average throughput, average latency, TQoSV, ThQoSV, and LQoSV respectively. The traffic-throughput-latency QoS estimation algorithm computes the QoS support value produced by different routes in terms of latency, throughput, and traffic. Such computed parameters are converted into feature sets and are used to perform latency Approximation.

Time-oriented latency (TLE) approximation

The time-oriented latency approximation technique estimates different QoS support values for given routes (As shown by Algorithm 3). Different routes would have different traffic, latency, and throughput according to different conditions. However, if the route is free of adversaries and traffic, then it would produce higher throughput and latency will be less. So, this approach estimates QoS support values on different parameters like traffic, throughput, and latency. To perform this, the method preprocesses the access traces and then traces from which noise has been removed have been used in estimating different QoS support values throughput Traffic-Throughput-Latency QoSV estimation scheme. The method returns values of TqosV, ThQoSv, and LQoSV sets which contains values of QoS support belonging to different factors at the different time stamp. The method calculates QoSV values for various routes using the provided values. By evaluating the QoSV value [21], the method selects a specific route for data transmission. Equation (vii) is employed to compute the QoSV, and equation (viii) is used to calculate the route R. The time complexity of the proposed method is determined by the encryption time, denoted as O(n).

The time-oriented latency approximation algorithm measures different QoS support values on latency, traffic, and throughput on different routes with various time stamps. The method calculates a distinctive Quality of Service Value (QoSV) and chooses the route with the highest QoSV value. This selected route is utilized for secure data transmission [23]. The general outline of Algorithm 3 is provided in Table 4.

TLADE data encryption

Cloud computing has gained immense popularity as a storage and data access solution, facilitated by the Internet and remote servers. Instead of owning physical resources, clients lease them from third-party providers. User management and key management are crucial aspects in cloud computing, encompassing tasks such as user setup, key generation, expiration, and destruction. The security of data is a significant

Appendix B

Algorithm 2: Traffic-throughput-latency QoSV estimation.

```
Given: Preprocessed Trace PT, Route List Rl
Obtain Feature Set Fs.
Start
  Read PT and RL.
  Find total time stamp Tts = $\sum_{i=1}^{Size(PT)} (TimeStamp \in PT(i))Tts$
  For each route R
    Find traces RT = $\sum_{i=1}^{size(Tts)} Tts(i).Route == R$
    For each timestamp Ti
      Compute average Traffic ATr = $\frac{\sum_{i=1}^{size(RT)} RT(i).Traffic \ \&\& \ RT(i).Time == Ti}{size(RT)}$ $(i)$
  Compute Average throughput AThr = $\frac{\sum_{i=1}^{size(RT)} RT(i).BytesTransfered \ \&\& \ RT(i).Time == Ti}{size(RT)}$ $(ii)$
      Compute average latency Alt = $\frac{\sum_{i=1}^{size(RT)} RT(i).latency \ \&\& \ RT(i).Time == Ti}{size(RT)}$ $(iii)$
      Compute TQoSv = $\frac{\sum_{i=1}^{size(RT)} RT(i).Traffic < ATr}{size(RT)}$ $(iv)$
      Compute ThQoSV = $\frac{\sum_{i=1}^{size(RT)} RT(i).BytesTransfered > Athr}{size(RT)}$ $(v)$
      Compute LQoSV = $\frac{\sum_{i=1}^{size(RT)} RT(i).Latency < AlT}{size(RT)}$ $(vi)$
      Generate Feature Vector Fv = {Ti,TQoSV, ThQoSV, LQoSV}
      Add to feature set Fs.
    End
  End
Stop
```

Appendix C

Algorithm 3: Time-oriented latency (TLE) approximation.

```
Given: Feature Set Fs, Route List Rl
Obtain Route R.
Start
  Read Fs and Rl.
  For each route R
  Compute QoSv = $\frac{\frac{\sum_{i=1}^{size(RT)} RT(i).Traffic \ \&\& \ RT(i).Time == Ci}{size(RT)}}{\frac{\sum_{i=1}^{size(TTs)} TTS(u\backslash i).TQoSv}{size(TTS)}} \times \frac{\frac{\sum_{i=1}^{size(RT)} RT(i).BytestTransfered \ \&\& \ RT(i).Time == Ci}{size(RT)}}{\frac{\sum_{i=1}^{size(TTs)} TTS(u\backslash i).ThQoSv}{size(TTS)}} \times \frac{\frac{\sum_{i=1}^{size(RT)} RT(i).Latency \ \&\& \ RT(i).Time == Ci}{size(RT)}}{\frac{\sum_{i=1}^{size(TTs)} TTS(u\backslash i).TQoSv}{size(TTS)}}$ $(vii)$
  End
  Route $R = \underset{i=1}{\overset{size(Rl)}{Max}}(Rl.QoSV)$ $(viii)$
Stop
```

concern for businesses due to the sensitive information transmitted over the Internet [38–43]. The term "key management" encompasses the creation, distribution, processing, usage, destruction, and replacement of cryptographic keys during the process of crypto shredding. It includes cryptographic protocols, key servers, client procedures, and other relevant protocols. Keys can be shared at the user or system level, and this concept differs from key preparation, which pertains to managing keys within the internal functioning of a cipher. The effectiveness of a cryptosystem's security relies on its ability to manage keys efficiently. Cryptography involves not only computational aspects but also social technology features such as system processes, user training, organizational interactions, and departmental collaborations.

The TLADE scheme maintains a set of encryption schemes to enforce data security. The method first identifies the set of schemes available and measures their average latency from the trace. According to the trace available, the method computes latency support according to the data size and calculated latency. Using these values, the method finds the most effective scheme for data encryption and hands over the data to be transmitted through the selected route. The latency support LS has been presented in Algorithm 4 and calculated by equation (ix).

The TLADE data encryption algorithm evaluates the latency support of various schemes and selects a route with the highest latency support. Subsequently, the data is encrypted and transmitted through the selected route.

Results and discussion

The time-variant latency approximation-based data encryption scheme, as proposed, has been implemented and thoroughly evaluated across various scenarios to assess its performance. The evaluation results are compared against the outcomes of alternative approaches.

The details of the evaluation considered for performance analysis of the proposed approach have been presented in Table 2. The proposed technique has been compared with techniques present in literature on the basis of encryption/decryption time with respect to varying data size. This has been presented in Table 3.

The encryption/decryption time introduced by various methods is measured and presented in Fig. 2. The proposed TLADE technique takes less time due to its efficient methodology that optimizes encryption and

Appendix D

Algorithm 4: TLADE data encryption.

```
Given: Route R, Service Access Trace SaT, Data D
Obtain; Null
Start
  Read R, D, and SaT.
  For each service S
    Compute Latency Support LS = $\frac{\sum_{i=1}^{SaT} SaT(i).Bytes \,\&\&\, SaT(i).Scheme = S}{\sum_{i=1}^{SaT} SaT(i).Scheme = S} \times \frac{\sum_{i=1}^{SaT} SaT(i).Time \,\&\&\, SaT(i).Scheme = S}{\sum_{i=1}^{SaT} SaT(i).Scheme = S} \quad (ix)$
  End
  Scheme s = Choose the most latency support scheme.
  Cipher text CT = perform Encryption (D, S)
  Send data in the selected route.
Stop
```

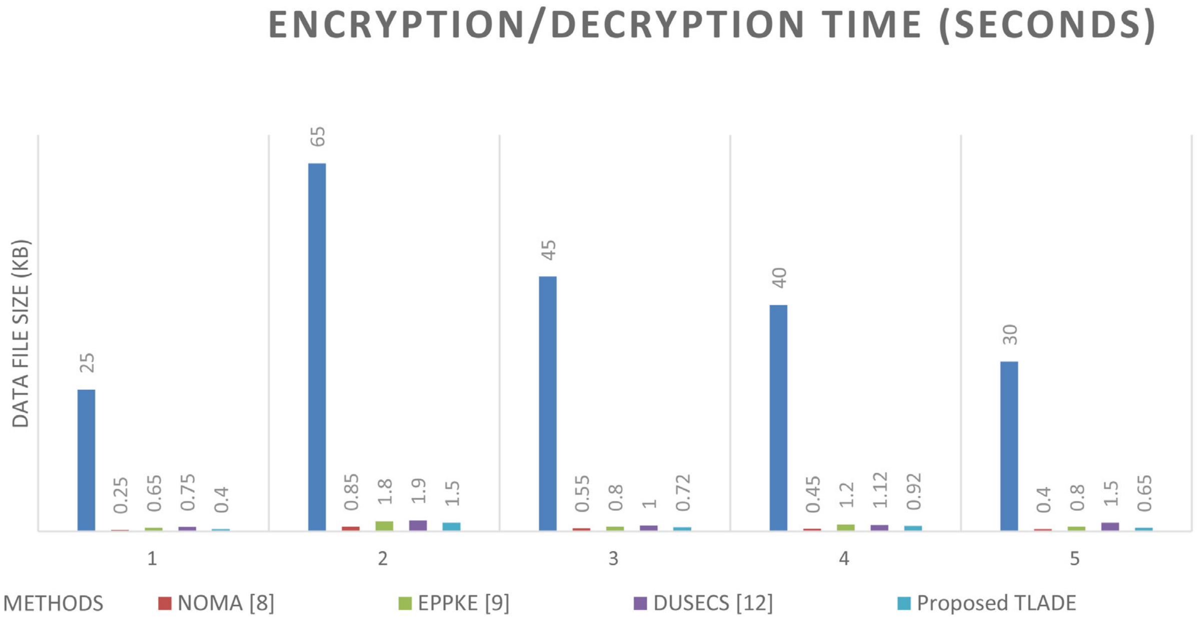

Fig. 2. Encryption/decryption time (Seconds).

Table 2
Evaluation details.

Details	Values
Tool Used	Advanced Java
No of services	100
Number of Users	500
Size of Data set	1 million tuples

decryption functions, reducing computational overhead and enhancing overall performance.

The average latency can be computed using Appendix B, as presented in Table 4, which provides a general representation of the algorithm.

The Table 5 represents the results for each timestamp (Ti) analyzed in the algorithm. For each timestamp, the average traffic, throughput, and latency values are computed based on the corresponding traces. The QoS values for traffic, throughput, and latency are have also been calculated using Appendix B. Table 5 shows the traffic-throughput-latency QoSV estimation.

To calculate the average latency (Alt) (see equation x) for each timestamp in the table, you need the sum of latencies and the number of traces available for each timestamp. Assuming you have the following data:

For T1:

$\sum Latency\ for$ T1: 50
Number of traces for T1: 5

For T2:

Table 3
Comparative analysis of TLADE with existing techniques.

Data File Size (kb)	Encryption/Decryption Time (In Seconds)			
	NOMA [8]	EPPKE [9]	DUSECS [12]	Proposed TLADE
25	0.25	0.65	0.75	0.4
65	0.85	1.8	1.9	1.5
45	0.55	0.8	1	0.72
40	0.45	1.2	1.12	0.92
30	0.4	0.8	1.5	0.65
Average Time	0.5	1.05	1.254	0.838

Table 4
General representation of algorithm 2.

Timestamp (Ti)	Traffic Average (ATr)	Throughput Average (AThr)	Latency Average (Alt)	Traffic QoS Value (TQoSV)	Throughput QoS Value (ThQoSV)	Latency QoS Value (LQoSV)
T1	ATr(T1)	AThr(T1)	Alt(T1)	TQoSV(T1)	ThQoSV(T1)	LQoSV(T1)
T2	ATr(T2)	AThr(T2)	Alt(T2)	TQoSV(T2)	ThQoSV(T2)	LQoSV(T2)
T3	ATr(T3)	AThr(T3)	Alt(T3)	TQoSV(T3)	ThQoSV(T3)	LQoSV(T3)
…	…	…	…	…	…	…

Table 5
Traffic-throughput-latency QoSV estimation.

Timestamp (Ti)	Traffic Average (ATr)	Throughput Average (Athr)	Latency Average (Alt)	Traffic QoS Values (TQoSV)	Throughput QoS Value (ThQoSV)	Latency QoS Value (LQoSSV)
T1	100	200	10	0.8	0.9	0.6
T2	150	180	15	0.7	0.8	0.7
T3	120	220	12	0.9	0.7	0.8
T4	180	190	18	0.6	0.8	0.5

$\sum \mathit{Latency\ for}$ T2: 60
Number of traces for T2: 4

For T3:

$\sum \mathit{Latency\ for}$ T3: 48
Number of traces for T3: 4

For T4:

$\sum \mathit{Latency\ for}$ T4: 36
Number of traces for T3: 2

The formula to calculate the average latency (Alt) for a specific timestamp (Ti) is:

$$\mathrm{Alt(Ti)}= \sum \mathit{Latency\ for}\ \mathrm{Ti/NumberoftracesforTi(x)}$$

Now, let's calculate the values for the table:

For T1: Alt(T1) = 50 / 5 = 10
For T2: Alt(T2) = 60 / 4 = 15
For T3: Alt(T3) = 48 / 4 = 12
For T4: Alt(T4) = 36 / 2 = 18

An evaluation of the latency ratio can be derived from Appendix D. To determine the Analysis of the Latency Ratio, we require the Latency Average (Alt) values obtained from the previous table (Table 5) and random Latency Support (LS) values. With these inputs, we can calculate the Latency Ratio and present the results in a new table (Table 6).

Table 6
Latency support for the different Schemes.

Timestamp (Ti)	Latency Average (Alt)	Latency Support (LS)
T1	10	8
T2	15	12
T3	8	10
T4	18	15

Now, we can calculate the Analysis of Latency Ratio using the equation (xi):

Latency Ratio (LR) = Latency Average (Alt) / Latency Support (LS) (xi)

LR for T1, T2, T3, and T4 can be calculated using equation (xi)

LR_{T1} = 10 / 8 = 1.25
LR_{T2} = 15 / 12 = 1.25
LR_{T3} = 8 / 10 = 0.8
LR_{T4} = 18 / 15 = 1.20

Based on the provided Table 5, the calculations for Traffic QoS Value (TQoSV) and Throughput QoS Value (ThQoSV) are as follows:

For Traffic QoS Value (TQoSV):

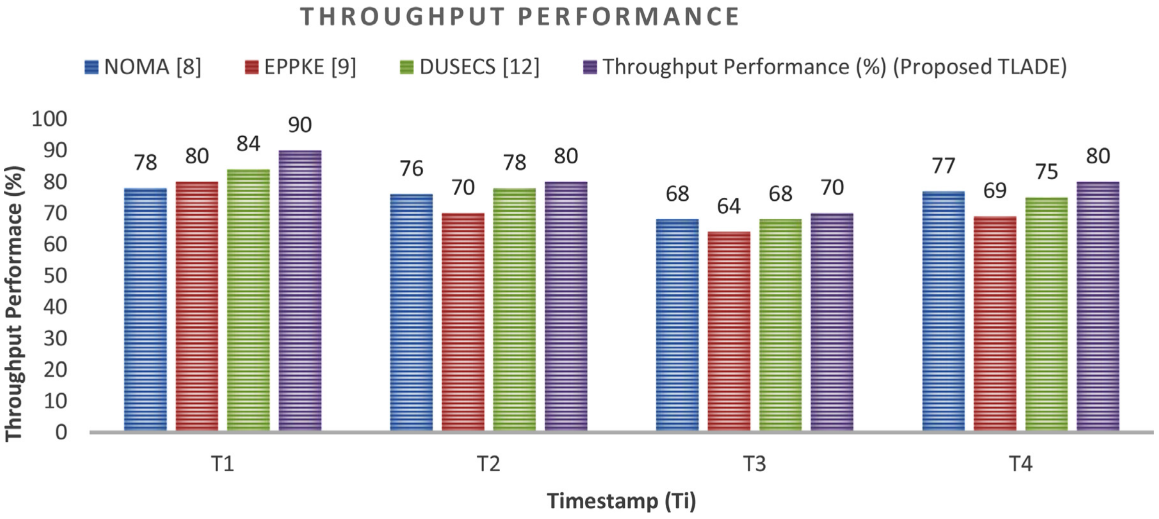

Fig. 3. Analysis of throughput performance.

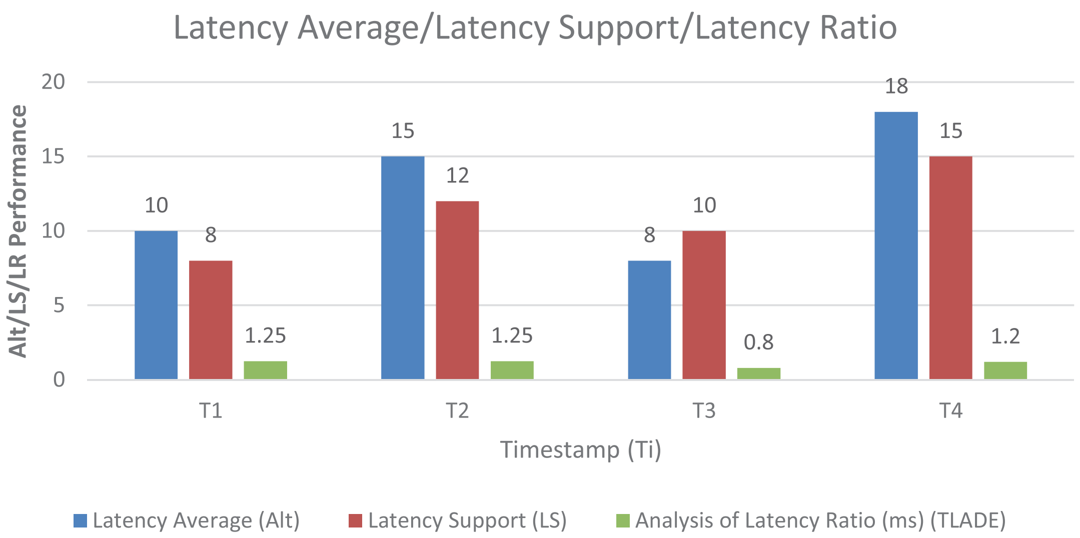

Fig. 4. Performance of Alt/LS/LR in terms of LR.

Table 7
Throughput performance.

Timestamp (Ti)	NOMA [8]	EPPKE [9]	DUSECS [12]	Throughput Performance (%) (Proposed TLADE)
T1	78	80	84	90
T2	76	70	78	80
T3	68	64	68	70
T4	77	69	75	80

TQoSV at T1: 0.8
TQoSV at T2: 0.7
TQoSV at T3: 0.9
TQoSV at T4: 0.6

For Throughput QoS Value (ThQoSV):

ThQoSV at T1: 0.9
ThQoSV at T2: 0.8
ThQoSV at T3: 0.7
ThQoSV at T4: 0.8

Table 7 and Fig. 3 present the measured throughput performance attained by various approaches. The proposed TLADE technique achieves higher throughput performance by implementing a streamlined methodology for encryption and decryption functions, minimizing processing delays and maximizing data transfer rates [44–46].

Using the provided values, the updated Table 8 with the Analysis of Latency Ratio will be as follows:

Table 8
Analysis of latency ratio.

Timestamp (Ti)	Latency Average (Alt)	Latency Support (LS)	Analysis of Latency Ratio (ms)
T1	10	8	1.25
T2	15	12	1.25
T3	8	10	0.8
T4	18	15	1.20

The details of the Comparative Analysis of LR have been presented in Table 9 and Fig. 5. The proposed technique has been compared with techniques present in literature.

Fig. 4 illustrates the measurement and presentation of the latency ratio introduced by different methods, namely Alt and LS. The proposed TLADE technique yields lower values of latency ratio compared to other existing methods due to its optimized methodology for encryption and decryption functions, reducing processing overhead and minimizing delays in data transmission, resulting in improved latency performance.

Conclusion

A novel approach to TLADE has been presented in the papers. The method identifies the routes and services. For the routes identified, the support of the route on latency, throughput, and traffic toward QoS maximization is measured. Based on the QoS support values, the method selects an optimal route for data transmission. Also, towards data encryption, the method estimates latency support according to the bytes encrypted and the time taken. Using the value of latency support an op-

Table 9
Comparative analysis of TLADE with existing techniques.

Timestamp (Ti)	NOMA [8]	EPPKE [9]	DUSECS [12]	Analysis of Latency Ratio (in ms) (Proposed TLADE)
T1	1.72	1.88	1.35	1.25
T2	1.4	1.31	1.61	1.25
T3	1	1.2	1.11	0.8
T4	1.35	1.45	1.63	1.2

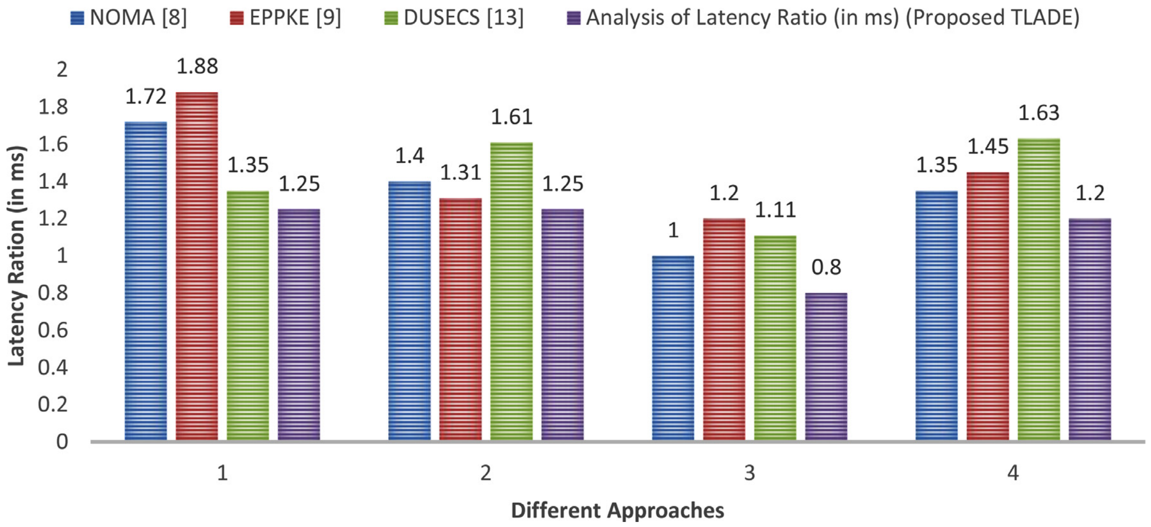

Fig. 5. Comparative analysis of LR.

timal service is selected and data has been encrypted. Encrypted data has been forwarded through the selected route. The proposed approach improves the performance of different QoS factors with reduced latency. As an extension of this work, multi-key homomorphic encryption techniques can be implemented to further increase secrecy and address cloud security challenges.

Declaration of Competing Interest

The authors declare that they have no competing interests.

CRediT authorship contribution statement

Shahnawaz Ahmad: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Shabana Mehfuz:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Supervision, Validation, Visualization, Writing – review & editing.

Funding

Not applicable.

Acknowledgments

The authors acknowledge the financial support received own, for their support and encouragement in carrying out his college work. The authors also would like to acknowledge the administration of Jamia Millia Islamia, which the authors represent.

References

[1] S. Ahmad, S. Mehfuz, F. Mebarek-Oudina, et al., RSM analysis-based cloud access security broker: a systematic literature review, Clust. Comput. (2022), doi:10.1007/s10586-022-03598-z.

[2] Shahnawaz Ahmad, Shabana Mehfuz, Javed Beg, Assessment on potential security threats and introducing novel data security model in a cloud environment, Mater. Today: Proc. 62 (2022) Part 7Pages 4909-4915, ISSN 2214-7853, doi:10.1016/j.matpr.2022.03.536.

[3] Shahnawaz Ahmad, Shabana Mehfuz, Javed Beg, Cloud security framework and key management services collectively for implementing DLP and IRM, Mater. Today: Proc. 62 (Part 7) (2022) 4828–4836 ISSN 2214-7853, doi:10.1016/j.matpr.2022.03.420.

[4] S. Ahmad, S. Mehfuz, J Beg, Fuzzy TOPSIS-based cloud model to evaluate cloud computing services, in: G. Manik, S. Kalia, S.K. Sahoo, T.K. Sharma, O.P. Verma (Eds.), Advances in Mechanical Engineering. Lecture Notes in Mechanical Engineering, Springer, Singapore, 2021, doi:10.1007/978-981-16-0942-8_4.

[5] S. Ahmad, S. Mehfuz, J. Beg, Enhancing security of cloud platform with cloud access security broker, in: M.S. Kaiser, J. Xie, V.S. Rathore (Eds.), Information and Communication Technology For Competitive Strategies (ICTCS 2020). Lecture Notes in Networks and Systems, Vol 190, Springer, Singapore, 2021, doi:10.1007/978-981-16-0882-7_27.

[6] Y.M. Mahaboob John, G. Ravi, Real-time regional mobility energy feature approximation-based secure routing for improved quality of service in MANET, Int. J. Commun. Syst. (2021), doi:10.1002/dac.4713.

[7] S. Zhou, L. Wu, C. Jin, A privacy-based SLA violation detection model for the security of cloud computing, China Commun. 14 (9) (Sept. 2017) 155–165, doi:10.1109/CC.2017.8068773.

[8] F. Hu, B. Chen, Channel coding scheme for relay edge computing wireless networks via homomorphic encryption and NOMA, in: IEEE Transactions on Cognitive Communications and Networking, 6, Dec. 2020, pp. 1180–1192, doi:10.1109/TCCN.2020.3023724.

[9] M.G. Aruna, K.G. Mohan, Secured cloud data migration technique by competent probabilistic public key encryption, China Commun. 17 (5) (May 2020) 168–190, doi:10.23919/JCC.2020.05.014.

[10] K. Lee, Comments on "Secure data sharing in cloud computing using revocable-storage identity-based encryption", in: IEEE Transactions on Cloud Computing, 8, 1 Oct.-Dec. 2020, pp. 1299–1300, doi:10.1109/TCC.2020.2973623.

[11] K. Gai, M. Qiu, H. Zhao, Privacy-preserving data encryption strategy for big data in mobile cloud computing, in: IEEE Transactions on Big Data, 7, 1 Oct. 2021, pp. 678–688, doi:10.1109/TBDATA.2017.2705807.

[12] L. Cao, Y. Kang, Q. Wu, R. Wu, X. Guo, T. Feng, Searchable encryption cloud storage with dynamic data update to support efficient policy hiding, China Commun. 17 (6) (June 2020) 153–163, doi:10.23919/JCC.2020.06.013.

[13] A. Awad, A. Matthews, Y. Qiao, B. Lee, Chaotic searchable encryption for mobile cloud storage, in: IEEE Transactions on Cloud Computing, 6, 1 April-June 2018, pp. 440–452, doi:10.1109/TCC.2015.2511747.

[14] Q. Zhang, S. Wang, D. Zhang, J. Wang, Y. Zhang, Time and attribute based dual access control and data integrity verifiable scheme in cloud computing applications, IEEE Access 7 (2019) 137594–137607, doi:10.1109/ACCESS.2019.2942649.

[15] L. Jiang, D. Guo, Dynamic encrypted data sharing scheme based on conditional proxy broadcast re-encryption for cloud storage, IEEE Access 5 (2017) 13336–13345, doi:10.1109/ACCESS.2017.2726584.

[16] S. Sharma, J. Powers, K. Chen, PrivateGraph: privacy-preserving spectral analysis of encrypted graphs in the cloud, IEEE Trans. Knowl. Data Eng. 31 (5) (1 May 2019) 981–995, doi:10.1109/TKDE.2018.2847662.

[17] O.A. Khashan, Secure outsourcing and sharing of cloud data using a user-side encrypted file system, IEEE Access 8 (2020) 210855–210867, doi:10.1109/ACCESS.2020.3039163.

[18] J. Wang, C. Huang, K. Yang, J. Wang, X. Wang, X. Chen, MAVP-FE: multi-authority vector policy functional encryption with efficient encryption and decryption, China Commun. 12 (6) (June 2015) 126–140, doi:10.1109/CC.2015.7122471.

[19] H. Deng, et al., Identity-based encryption transformation for flexible sharing of encrypted data in public cloud, in: IEEE Transactions on Information Forensics and Security, 15, 2020, pp. 3168–3180, doi:10.1109/TIFS.2020.2985532.

[20] M. Zeng, H. Qian, J. Chen, K. Zhang, Forward secure public key encryption with keyword search for outsourced cloud storage, in: IEEE Transactions on Cloud Computing, 10, 1 Jan.-March 2022, pp. 426–438, doi:10.1109/TCC.2019.2944367.

[21] H. Li, Y. Yang, Y. Dai, S. Yu, Y. Xiang, Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data, in: IEEE Transactions on Cloud Computing, 8, 1 April-June 2020, pp. 484–494, doi:10.1109/TCC.2017.2769645.

[22] H. Qiu, H. Noura, M. Qiu, Z. Ming, G. Memmi, A user-centric data protection method for cloud storage based on invertible DWT, in: IEEE Transactions on Cloud Computing, 9, 1 Oct.-Dec. 2021, pp. 1293–1304, doi:10.1109/TCC.2019.2911679.

[23] H. Yuan, X. Chen, J. Li, T. Jiang, J. Wang, R.H. Deng, Secure cloud data deduplication with efficient re-encryption, in: IEEE Transactions on Services Computing, 15, 1 Jan.-Feb. 2022, pp. 442–456, doi:10.1109/TSC.2019.2948007.

[24] X. Huang, H. Xiong, J. Chen and M. Yang, "Efficient revocable storage attribute-based encryption with arithmetic span programs in cloud-assisted Internet of Things," in IEEE Transactions on Cloud Computing, doi:10.1109/TCC.2021.3131686.

[25] J. Wu, X. Sheng, G. Li, K. Yu, J. Liu, An efficient and secure aggregation encryption scheme in edge computing, China Commun. 19 (3) (March 2022) 245–257, doi:10.23919/JCC.2022.03.018.

[26] S. ShanmugaPriya, A. Valarmathi, D. Yuvaraj, The personal authentication service and security enhancement for an optimal strong password, Concurr. Comput.: Pract. Exper. (2019) e5009.

[27] K.L. Neela, V. Kavitha, Enhancement of data confidentiality and secure data transaction in the cloud storage environment, Cluster Comput 21.1 (2018) 115–124.

[28] M. Wazid, P. Bagga, A.K. Das, S. Shetty, J.J. Rodrigues, Y. Park, AKM- IoV: authenticated key management protocol in fog computing-based Internet of vehicles deployment, IEEE Internet Thing J. 6 (5) (2019) 8804–8817.

[29] Y. Miao, X. Liu, R.H. Deng, H. Wu, H. Li, J. Li, D. Wu, Hybrid keyword- field search with efficient key management for the industrial internet of things, IEEE Transact. Indus. lnfom., I 5 (6) (2018) 3206–3217.

[30] K. Park, Y. Park, A.K. Das, S. Yu, J. Lee, Y. Park, A dynamic privacy- preserving key management protocol for V2G in social internet of things, IEEE Access 7 (2019) 76812–76832.

[31] Manish Kumar, Akhlad Iqbal, Pranjal Kumar, A new RGB image encryption algorithm based on DNA encoding and elliptic curve Diffie-Hellman cryptography, Signal Process. 125 (August 2016) 187–202.

[32] DI George Amalarethinam, H.M. Leena, Asymmetric addition chaining cryptographic algorithm (ACCA) for data security in cloud, in: Advances in Big Data and Cloud Computing, Springer, Singapore, 2018, pp. 331–340.

[33] A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm, Comput. Struct. 169 (2016) 1–12.

[34] G.B. Pallavi, P. Jayarekha, Secure, and efficient multi-tenant database management system for the cloud computing environment, Int. j. inf. tecnol. 14 (2022) 703–711, doi:10.1007/s41870-019-00416-5.

[35] M.O. Alassafi, H.F. Atlam, A.A. Alshdadi, et al., A validation of security determinants model for cloud adoption in Saudi organizations' context, Int. j. inf. tecnol. 14 (2022) 1075–1085, doi:10.1007/s41870-019-00360-4.

[36] S.C.M. Sharma, A.K. Rath, B.R Parida, Efficient load balancing techniques for multi-datacenter cloud milieu, Int. j. inf. tecnol. 14 (2022) 979–989, doi:10.1007/s41870-020-00529-2.

[37] S. Jain, R. Doriya, Security framework to healthcare robots for secure sharing of healthcare data from the cloud, Int. j. inf. tecnol. 14 (2022) 2429–2439, doi:10.1007/s41870-022-00997-8.

[38] C. Song, et al., Hierarchical edge cloud enabling network slicing for 5 G optical fronthaul, J. Opt. Commun. Network. 11 (4) (April 2019) B60–B70, doi:10.1364/JOCN.11.000B60.

[39] Y. Yao, Z. Zhai, J. Liu, Z. Li, Lattice-based key-aggregate (searchable) encryption in cloud storage, IEEE Access 7 (2019) 164544–164555, doi:10.1109/ACCESS.2019.2952163.

[40] S. Wang, R. Pei, Y. Zhang, EIDM: a ethereum-based cloud user identity management protocol, IEEE Access 7 (2019) 115281–115291, doi:10.1109/ACCESS.2019.2933989.

[41] Y. Miao, et al., Hybrid keyword-field search with efficient key management for industrial Internet of Things, IEEE Trans. Ind. Inf. 15 (6) (June 2019) 3206–3217, doi:10.1109/TII.2018.2877146.

[42] M. Ma, G. Shi, F. Li, Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario, IEEE Access 7 (2019) 34045–34059, doi:10.1109/ACCESS.2019.2904042.

[43] F. Wang, L. Xu, W. Gao, Comments on "SCLPV: secure certificateless public verification for cloud-based cyber-physical-social systems against malicious auditors, IEEE Transact. Comput. Soc. Syst. 5 (3) (Sept. 2018) 854–857, doi:10.1109/TCSS.2018.2858805.

[44] N. Babu, T. Suresh, Real time continuous behavior class based secure routing for improved QoS in industrial networks, in: 2022 International Conference on Electronic Systems and Intelligent Computing (ICESIC), 2022, pp. 111–116, doi:10.1109/ICESIC53714.2022.9783571.

[45] S. Muruganandam, J.A. Ranjit, Real-time reliable clustering and secure transmission scheme for QoS development in MANET, Peer-to-Peer Netw. Appl. 14 (2021) 3502–3517, doi:10.1007/s12083-021-01175-6.

[46] P. Sathyaraj, D.R. Devi, Retraction note to designing the routing protocol with secured IoT devices and QoS over Manet using trust-based performance evaluation method, J. Ambient Intell. Human. Comput. (2022), doi:10.1007/s12652-022-03956-0.

[47] S. Ahmad, S. Mehfuz, J. Beg, An efficient and secure key management with the extended convolutional neural network for intrusion detection in cloud storage, Concurr. Comput. Pract. Exper. (2023), doi:10.1002/cpe.7806.

[48] S. Ahmad, S. Mehfuz, J. Beg, Hybrid cryptographic approach to enhance the mode of key management system in cloud environment, J. Supercomput. (2022), doi:10.1007/s11227-022-04964-9.

[49] S. Urooj, S. Lata, S. Ahmad, S. Mehfuz, S. Kalathil, Cryptographic data security for reliable wireless sensor network, Alexand. Eng. J. 72 (2023) 37–50 ISSN 1110-0168, doi:10.1016/j.aej.2023.03.061.