Research Article

Security And Privacy Using Two Fold Encryption Protocol Techniques In Edge Computing

K.S.Mohanasathiya[1] and Dr.S.Prasath[2]

[1]Ph.D. Research Scholar (Part-Time), Department of Computer Science, Nandha Arts and Science College, Erode, Tamil Nadu, India
[E-Mail id: sathyaanandh08@gmail.com]
[2]Assistant Professor & Research Supervisor, Department of Computer Science,
Nandha Arts and Science College, Erode, Tamil Nadu, India
[E-Mail id: softprasaths@gmail.com]

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

Abstract : With deployments of IoT devices and the arrival of 5G fast wireless, placing compute and analytics close to where data is created is making the case for edge computing. Edge computing is transforming the way data is being handled, processed, and delivered from millions of devices around the world. The explosive growth of internet-connected devices – the IoT – along with new applications that require real-time computing power, continues to drive edge-computing systems. Due to several advantages like low latency, Real-Time Availability. Real-time Data Transmission, brings Company and customer together, Edge Computing is used. To overcome several problems regarding security and privacy concerns, this paper provides solution by introducing Secured Two Fold Encryption Protocol in Edge Computing (STFC) based on two encryption schemes namely proxy re-encryption and searchable encryption. This paper particularly focus on securing the data from attackers by providing double encryption scheme. In this paper, Resource Manager is introduced in order to reduce the computational overhead in the encryption process.

Keywords*: IOT, STFC, edge computing, KGC, encryption, decryption.*

1. Introduction

Cloud computing is an emerging area of computer technology that benefits form the processing power and the computing resources of many connected, geographically distanced computers connected via Internet. Cloud computing eliminates the need of having a complete infrastructure of hardware and software to meet users requirements and applications. Cloud computing has its own drawback from the security point of view [18][19]. The current tendency to solve the problems that the Cloud computing has is performing the computations as close as possible to the device. This paradigm is called Edge Computing. However, increasing number of connected devices brings problems, such as low Quality of Service (QoS) due to infrastructure resources and high latency because of the bandwidth limitations. The current tendency to solve the problems that the Cloud computing has is performing the computations as close as possible to the device. This paradigm is called Edge Computing [15] [16]. The edge computing network is generated by the common development of cloud computing and the IoT. Edge computing was developed due to the exponential growth of IoT devices, which connect to the internet for either receiving information from the cloud or delivering data back to the cloud. And many IoT devices generate enormous amounts of data during the course of their operations. Edge computing is a distributed computing paradigm which brings computation and data storage closer to the location where it is needed, to improve response times and save bandwidth. However, the unique features of edge computing, such as content perception, real-time computing, and parallel processing, has also introduced several new challenges in the _field of data security and privacy preserving, which are also the key concerns of the other prevailing computing paradigms, such as cloud computing, mobile cloud computing, and fog computing. Despites its importance, there still lacks a survey on the recent research advance of data security and privacy-preserving in the field of edge computing [10] [11] [12] [13].

To solve these security threats in edge computing, encryption techniques like proxy re-encryption along with searchable encryption techniques are used .Proxy re-encryption (PRE) schemes are cryptosystems which allow third parties (proxies) to alter a cipher text which has been encrypted for one party, so that it may be decrypted by another. Proxy re-encryption furthermore provides a data-sharing function that allows the system to make optimal use of the limited data storage capacity of a network device. A proxy re-encryption (PRE) scheme [23] [24] [25] is a public-key encryption scheme with an additional functionality: Alice and Bob, who have key pairs (pkA, skA) and (pkB, skB), respectively, can generate a re-encryption key (re-key, for short) rkA,B that allows its holder, say Peggy, to act as a proxy; that is, she can transform cipher texts under pkA to cipher texts under pkB without having to know the underlying message. A trivial way to accomplish this would be for Alice to hand her secret key skA to Peggy, who

can then decrypt cipher texts under pkA, encrypt them under pkB and send them to Bob. Alice's secret key acts thus as the re-key and de- and encryption algorithms are used for re-encryption. However, this approach requires Alice to reveal her secret key to Peggy and therefore place complete trust on her. Searchable Encryption (SE) which enable users to secure search on encrypted data stored in the cloud. Encryption (SE) techniques which allow accessing data on encrypted cloud were reviewed and classified. SE is a term of searching on encrypted data located on untrusted server or cloud without the need to decrypt. SE (Searchable Encryption) is a positive way to protect user's sensitive data, while preserving search ability on the server side. SE allows the server to search encrypted data without leaking information in plaintext data. . The two main branches of SE are SSE (Searchable Symmetric Encryption) and PEKS (Public key Encryption with Keyword Search). SSE allows only private key holders to produce cipher texts and to create trapdoors for search, whereas PEKS enables a number of users who know the public key to produce cipher texts but allows only the private key holder to create trapdoors [20] [21] [22].

The main contributions of this paper are as follows:

1) This paper focuses on the security and privacy constraints by introducing Secured Two Fold Encryption Protocol in Edge Computing (STFC) based on two encryption schemes namely proxy re-encryption and searchable encryption.
2) In this paper, Resource Manager is introduced in order to reduce the computational overhead in the encryption process. This takes care of authenticating the user by using attributes of the user.
3) In order to safeguard data from attackers, double encryption scheme is proposed with two private keys.

2. Literature Survey

Manuel Koschuch et al [1] brought out a Searchable Encryption (SE) schemes that enable the cloud provider to search for user supplied strings in the encrypted documents, while neither learning anything about the content of the documents nor about the search terms. In this paper, there is no additional guidance is provided when selecting a specific SE scheme.

SuHyun Kim et al [2] proposed a method for sharing and managing data using the conventional cipher algorithm on lightweight devices in various circumstances. In this method proxy re-encryption is implemented in order to manage data with fewer encryptions, and provides a data sharing function to supplement the insufficient capacity of lightweight device networks. However, a disadvantage of this method is the additional computation in the polynomial equation compared to existing attribute-based encryption methods.

Sneha Kanchan et al [3] introduced the SignReCrypting Proxy Re-signature scheme, which reduces the time taken for encryption at sender side as well as for decryption at receiver side. The number of exponential functions and multiplications used in this algorithm is high and this increases cost.

Wei Wang et al [4] exploited the capability of the edge-cloud architecture and proposed a lightweight-designed scheme called Edge-aided Searchable Public-key Encryption (ESPE). Consequently, ESPE accelerates the ciphertext corresponding procedures on edges and saves over 70% encryption cost of an IoT device.

Rehmat Ullah et al [5] proposed an open source framework integrating Named data networking (NDN) and edge cloud computing (ECC) via N-Tier architecture with 7 layers at Edge Tier and 6 layers at the Cloud Tier. NDN integrated with ECC in order to achieve fast information response time in this paper. In addition, edge computing offers computation, storage, and increased caching capabilities in close proximity of end users/devices, thereby reducing end to end latency and backbone traffic.

Youhuizi Li et al [6] proposed an efficient time-domain multi-authority outsourcing attribute-based encryption (ABE) scheme (TMO) with a dynamic policy updating method for secure data acquisition and sharing in the edge computing. There is need of improving TMO to support a large number of users and devices with more features in the edge environment.

[1] Footnote text.

Abdulatif Alabdulatif et al [7] developed a secure Edge of Things (EoT) framework for smart health surveillance. The framework can aggregate, monitor and perform real-time analysis of bio signal data. In this paper, proposed framework in terms of further reducing homomorphic computational overheads is needed.

Yongchuan Niu et al [8] focussed on three typical AES implementations in edge computing, and proposed a new type of collision attack by making use of leakages from linear layers, which is capable of breaking masking schemes with uniformly distributed random masks. In addition, a novel scalable collision attack of general applicability and high efficiency is proposed and applied to masked linear layers and masked S-boxes.

Hui Cui et al [9] proposed a primitive named proxy-aided cipher text-policy ABE (PACPABE), which outsources the majority of the decryption computations to edge devices. In his paper, they presented a generic

construction of PA-CPABE and then formally prove its security and also presented and implemented an instantiation of PA-CPABE. To evaluate its efficiency.

3. Methodology

3.1 Network Model

The network model illustrates the general architecture of edge computing which consists of a four-layer functional structure: cloud service layer, edge server layer, edge network layer and edge device layer. The cloud service layer provides centralized cloud computing services and management functions for mobile edge devices. All cipher text in the system will be uploaded to the cloud for permanent storage. Secondly, edge servers, which are owned and deployed by the infrastructure provider and equipped with multi-tenant virtualization infrastructure, are responsible for providing virtualized and multiple management services, edge devices include all types of devices connected to the edge network (e.g. mobile terminals, IoT devices) which are not only play role as data consumers, but also data producers to participate in the distributed infrastructure for all four layers.

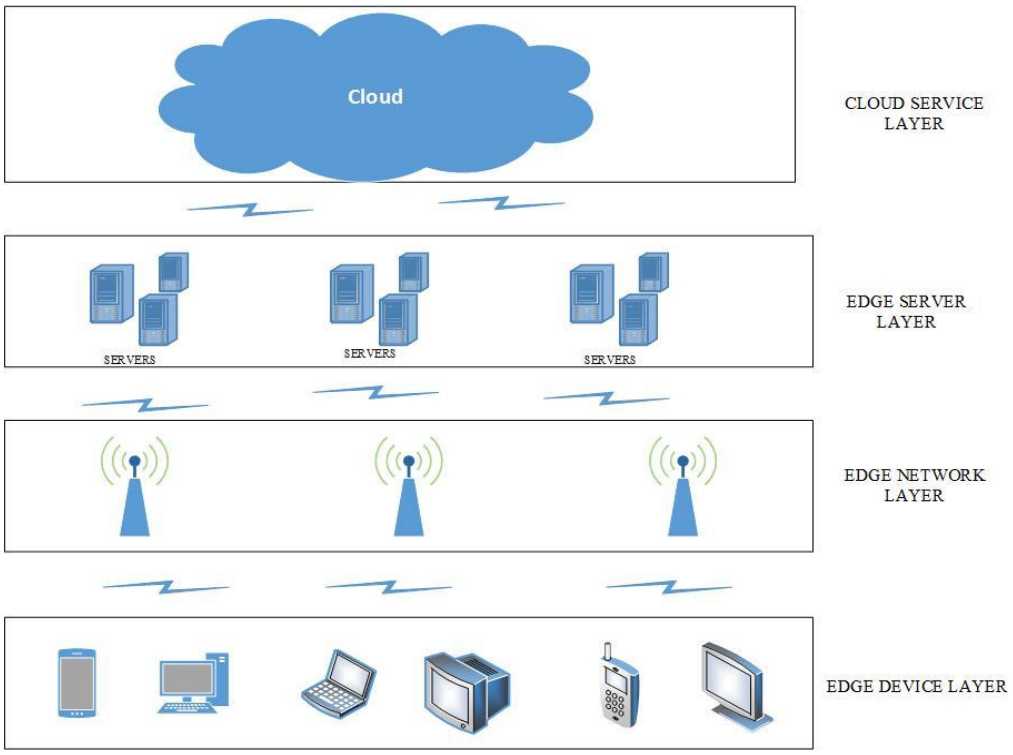

Fig.1 Architecture of Edge Computing

3.2 Threat Model

In this section, the major state-of-the art security threats and attacks faced by edge computing are discussed [17] [26].

3.2.1 DDoS Attacks

DDoS refers to a type of cyber-attack in which attackers aim to disrupt normal services provided by one or more servers based on distributed resources such as a cluster of compromised edge devices. Compared with cloud servers, edge servers are more susceptible to DDoS attacks since they are relatively computationally less powerful to maintain strong defense systems as cloud servers do. In addition, edge servers mainly provide services to edge devices that are well-known to be error-prone in regard to security settings due to their computation-limited hardware and heterogeneous firmware. Having noticed this, attackers Favour first compromising a number of edge devices and turning them into weapons against edge servers. The Mirai botnet is an infamous example where the attacker took control of over 65 000 IoT devices within the first 20 h after its release. DDoS attacks may happen when malicious edge devices communicate with the edges servers. Flooding-based attacks area type of DDoS attacks aiming to shut down the normal service of a server based on a large amount of flooded malformed/malicious network packets.

3.2.2 Side-Channel Attacks

An attacker constantly obtains certain side-channel information from the target edge computing infrastructure and then feeds it into specific algorithms or machine learning models that output the desired sensitive information. The most popular side channels in edge computing include communication signals, electric power consumption, and smartphone or proc file system or embedded sensors. The attacks exploiting communication channels happen when the attacker continuously monitors the transmissions between two edge nodes, those exploiting power consumption occur after the attacker steals power consumption data of the edge devices, and those exploiting smartphone-based channels happen when the attacker secretly accesses the smartphone and steals the information stored in the publicly available or proc file or generated by the embedded sensors.

3.2.3 Malware Injection Attacks

The objective of this attack is to inject malware i.e., malicious codes, into edge devices or edge servers. Malware injection attacks in edge computing is classified into two categories: server-side injections (injection attacks targeting edge servers) and device-side injections.

Fig.2 Malware Injection Attack

4. Design Goals

Our design goal is to propose a reliable and privacy-preserving data scheme in networks, which guarantees the following objectives:

Security: The security requirements such as data security and data privacy should be satisfied.

Efficiency: The proposed scheme should be efficient. That is, the computational costs of the proposed scheme should be acceptable. Also, the communication efficiency could be achieved in the Edge -based architecture.

Easy to Deploy: The proposed scheme should be easy to be deployed. Namely, the Edge-based architecture needs to support different applications, while the IoT devices can easily manage their keys.

In existing system [9], a primitive named proxy-aided ciphertext-policy ABE (PA-CPABE) was proposed, which outsources the majority of the decryption computations to edge devices. It consists of four entities: data users, the trusted key generation center (KGC), untrusted proxies (i.e., edge nodes or edge devices) and data owners. The KGC is in charge of the creation of the common parameter and the master private key. The KGC keeps the latter in secret and make the former public. Once a data user Bob intends to join the network, he registers with the KGC. Firstly, Bob creates a public user-key and a private user-key. Then, Bob transmits the public user-key to the KGC and keeps the private user-key as a secret. The KGC, on the basis of Bob's eligible attributes and public user-key, produces a public transformation key for Bob, which is going to be broadcast to all local edge nodes of Bob. In case that Bob needs to access a cipher text, Bob transmits to the cloud a request, and the cloud will forward the cipher text to the nearby edge device which is capable of performing the computation. If the attribute set possessed by Bob satisfies the access structure associated with the cipher text, the edge device is capable of using the transformation key of Bob to partially decrypt the cipher text. After obtaining the transformed cipher text from the edge device, Bob uses his private user key to fully decrypt it to obtain the underlying plaintext.

4.1 Proposed Design Scheme

In this section, Secured Two Fold Encryption Protocol in Edge Computing (STFC) is proposed based on Attribute based encryption and proxy re-encryption techniques.

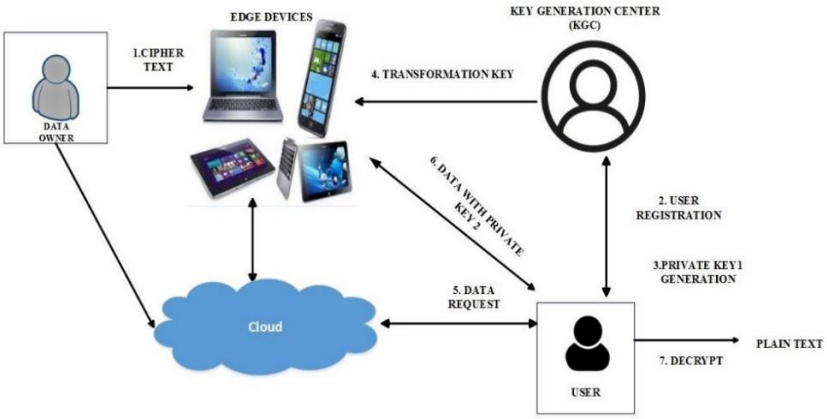

Fig.3 Architecture of the Proposed Method

The proposed scheme includes the following phases: Encryption phase, User Registration, Private Key1 generation, Transformation key generation, Data Request, data with private key2 generation and finally Decryption phase. The figure involves four entities: data users, the trusted Key Generation Center (KGC), untrusted proxies (i.e., edge nodes or edge devices) and data owners.

4.1.1 Encryption phase

In this phase, data owner converts their plain text data into cipher text with public key before uploading data into cloud. The converted cipher text will be sent to nearby edge devices and then it is forwarded by the edge device to the cloud if necessary.

4.1.2 User Registration and Private Key1 generation phase

Key Generation Center (KGC) is responsible for generating Private key1 and transformation key. Once a data user wants to join the network, he wants to first register with the KGC. To decrypt the data, the user needs two private keys. The first one will be generated by the KGC as generate private key1 to the user who involved in registration phase.

4.1.3 Transformation key Generation

The key generation center (KGC) will then generate the transformation key and send it to the edge devices. The edge devices use this transformation key for Re-encrypting the cipher text received from the data owner. The resource manager included in edge servers reduces computational overhead in the proposed method.

4.1.4 Data request phase

In this phase, the user needs to access a cipher text. So, the user request the cloud for accessing a cipher text and the cloud will forward the cipher text to the nearby edge device which is capable of performing the computation.

4.1.5 Data Generation phase

After receiving the transformation key from the Key Generation centre, edge device will check the attributes of the requested user. The attributes also contains the information about the data need to be searched in the cloud. If the attribute set possessed by the user satisfies the access structure associated with the cipher text, the edge device is capable of using the transformation key to re-encrypt the cipher text and send the needed re-encrypted data along with private key2 to the user.

4.1.6 Decryption Phase

The user will use the both private keys (private key1 during data registration and private key2 during data transfer) to decrypt the re-encrypted data into plain text.

4.2 Proposed Algorithm

Input: *Plain text*

Output: *Data transmission to the Data User (DU)*

Begin

Data user (DU): Send an Encrypted Data (DR) to the Edge Device (ED)

Key Generation Centre (KGC): Generate Private key1 (priv_key1) & Transformation key (TK).

User Registration: Request the priv_key1

Data request phase: DU request Cloud

If (DU attributes matched),

ED re-encrypt the cipher text

Transfer the re-encrypted data & Private Key 2 to DU.

end if

End

5. Results and Discussion

5.1 Encryption Time

The following table shows the encryption time of different methods. To comprehensively evaluate the performance of the proposed STFC scheme, we compared its encryption time and private key transformation time with the encryption methods: DACMACS, OOMADO and TMO.

Table 1. Encryption Time of Different Methods

No. of attributes	Encryption time			
	TMO	OOMADO	DACMACS	STFC
2	1000	1600	1200	900
4	1800	2200	1900	1600
8	2000	2800	2200	1800
10	3600	4000	3800	2700
12	4500	7000	5000	3800

The encryption time is found to be reduced for the proposed method in the following graph when compared it with the existing methods. Here, the encryption mechanism takes place in two phases. First, the data owner itself encrypt the plain text and send it to the nearby edge device. Secondly, the encryption takes place in the edge device using transformation key. Since user itself undergoes encryption mechanism with the public key, in the proposed method STFC scheme, encryption time is reduced when compared with existing methods. In the same way encryption time is reduced while re-encrypting the cipher text with transformation key send by the KGC. Once the KGC sends the transformation key to edge device, edge device re-encrypts the data after authentication of the user by comparing the attributes of the user. Here, time consumption for proposed scheme is reduced.

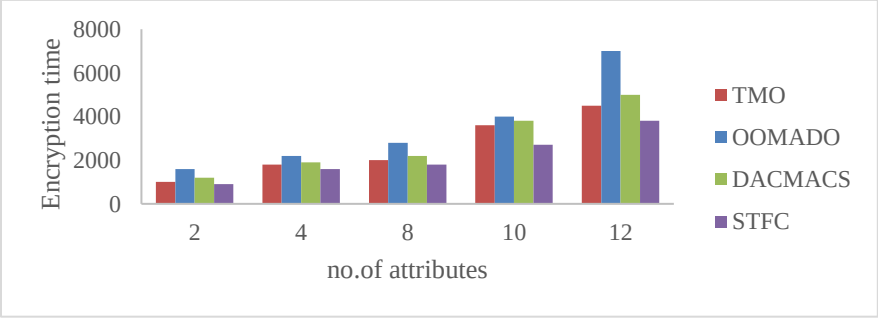

Fig.4 Encryption Time of Different Methods

5.2 Private Key Transformation Time

The following Table 2 shows the private key transformation time for different methods.

Table 2 Private Key Transformation Time for Different Methods

No. of attributes	Private key Transformation time (ms)		
	TMO	OOMADO	STFC
5	410	300	270
10	900	700	650
15	1300	1200	1100
20	1800	1300	1200

To show the advantage of STFC via the private key transformation time, we compare STFC with other two schemes. The private key transformation time is found to be reduced significantly when compared to other two schemes. The edge device need not wait for the transformation key. Once the user registration phase is completed, the Key Generation center generates transformation key and send it the edge device. Here, the time taken to transfer the transformation is significantly reduced when compared with existing schemes in the following graph.

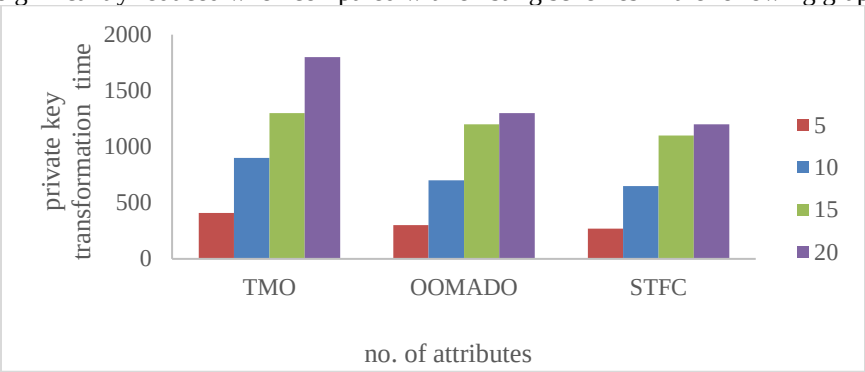

Fig.5 Private Key Transformation Time for Different Methods

5.3 Total Computation Overhead

The following table shows the comparison of total computation overhead with total no. of mobile users for different methods. In the below figure, the total computation overhead has get reduced in the proposed system when compared to the existing methods.

Table 3 Comparison of total computation overhead with total no. of mobile users

Total no. of mobile users	Total Computation overhead		
	HODA	Offloading only	STFC
10	10	12	8
18	20	25	10
26	32	35	23
34	40	90	32
50	43	135	40

The proposed system consists of resource manager in which the edge server handles the computational overhead caused during encryption process. In the proposed scheme, the computational overhead gets significantly reduced to 10 percentage when compared it to the existing schemes.

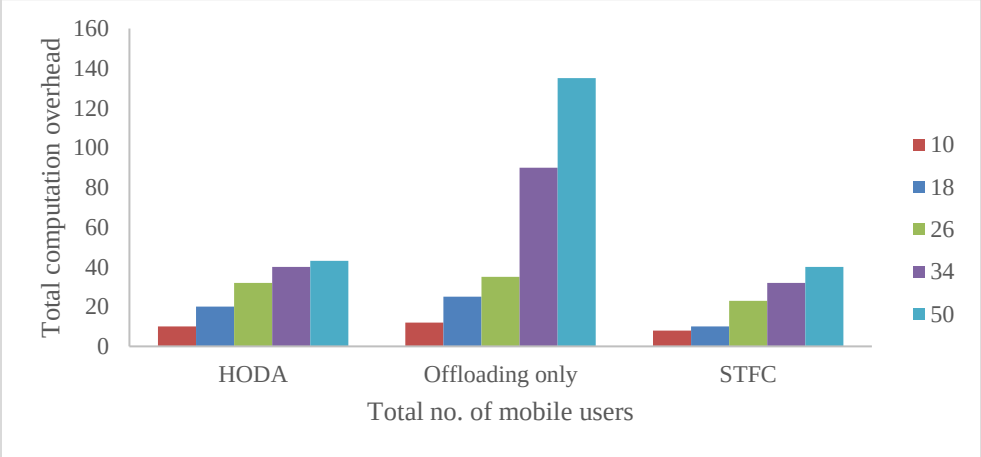

Fig.6 Comparison of total computation overhead with total no. of mobile users

5.4 Performance

Table 4: The variation of the intrusion time with system time after using three methods.

System time/ sec	Intrusion time/sec		
	Bayesian network method	Game method	Proposed method
2	1.1	0.80	0.30
4	1.3	0.94	0.52
6	1.7	1.10	0.6
8	2.9	1.36	1.01
10	3.5	2.5	1.5
12	4.6	3.8	1.3

The Fig.7 represents the variation of intrusion time with the system time for three methods. The intrusion time of proposed work is low when compared to other two methods as shown in Table 4. Due to double encryption of data, the data to be transferred to the data user is found to be secure in edge server. The two encryption keys helps in providing the security to the data. This proves that the proposed system is defensive against attacks and it is safer one and it is shown in Fig.7

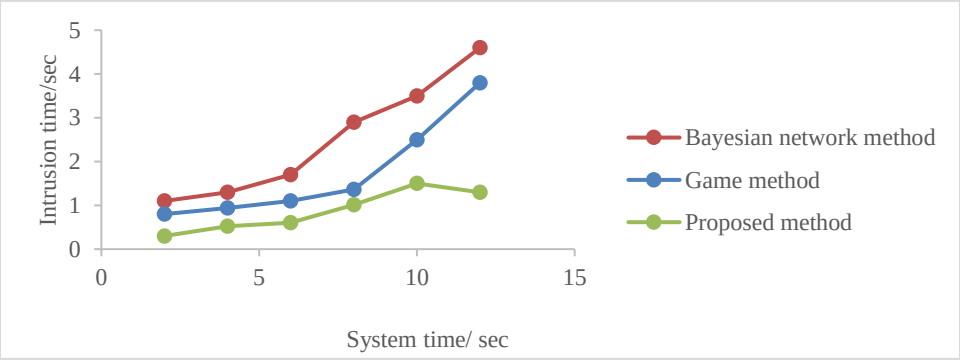

Fig.7 The variation of the intrusion time with system time after using three methods

5.5 Time Cost of Decryption

Fig. 8 shows the time cost to decrypt the data ciphertext under different schemes

Table 5: Time cost of Decryption vs No. of messages

No. of messages	Time cost (sec)		
	BPTM	BPREET-Re	Proposed method
0	1.1	0.8	0.1
20	2.1	1.5	0.2
40	4.8	3	0.5
60	7	4.1	0.8
80	9.2	5.9	0.9
100	11.5	7	1.0

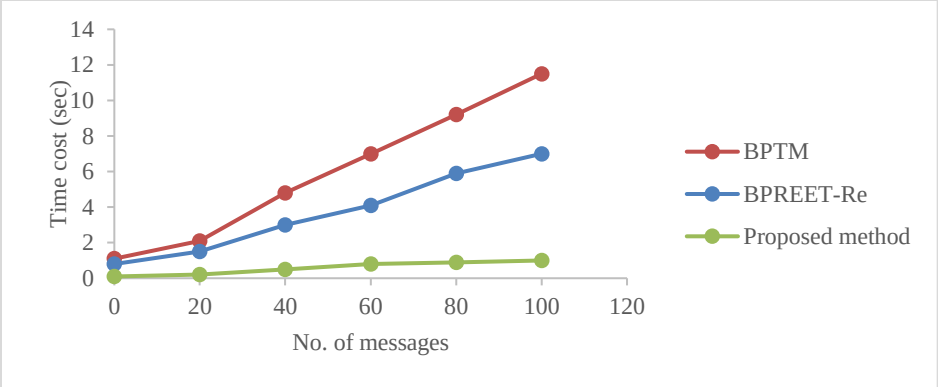

Fig.8 Time cost of Decryption vs No. of messages

Fig.8 shows the time cost to decrypt the data cipher text under different schemes. When the authentication for the data user gets completed, the user gets two gets for decrypting the cipher text. So the time taken for decryption gets reduced when compared to existing schemes.

6. Conclusion

The blooming of IoT and ever-increasing demands of users, edge computing is proposed to leverage the computing and storage resources on the edge to process the massive data. In this paper, Secured Two Fold Encryption Protocol in Edge Computing (STFC) based on two encryption schemes is proposed for solving security and privacy issues. This method achieves 10% reduction in encryption time and key transformation time when compared to the existing methods.

Acknowledgements

I extend my sincere thanks to my research supervisor Dr.S.Prasath working as Assistant Professor, Department of Computer Science, Nandha Arts and Science College, Erode, Tamilnadu, India. He has obtained Masters degree from M.Kumarasamy college of Engineering, Karur under Anna University, Chennai and M.Phil degree and Ph.D., Degree in Computer Science. His area of interests includes, Image Processing, Networking and Data Mining. He has published more than 45 journals papers in referred journals.

References

1. Manuel Koschuch, Michael Hombauer, Sigrid Schefer-Wenzl, Ulrich Habock (2015),"Fogging the Cloud — Implementing and Evaluating Searchable Encryption Schemes in Practice", 1365-1366.
2. SuHyun Kim, ImYeong Lee (2017), "IoT device security based on proxy re-encryption", Springer, pages 1267–1273.
3. [3] Sneha Kanchan, Narendra S. Chaudhari (2018), "SRCPR: SignReCrypting Proxy Re-Signaturein Secure VANET Groups", IEEE, volume:6,59282 – 59295.
4. Wei Wang, Peng Xu, Dongli Liu, Laurence Tianruo Yang, Zheng Yan (2019)," Lightweighted Secure Searching over Public-key Ciphertexts for Edge-Cloud Assisted Industrial IoT Devices", IEEE Transactions on Industrial Informatics,1-9.
5. Rehmat Ullah, Muhammad Atif Ur Rehman,and Byung-Seo Kim (2019), "Design and Implementation of an Open Source Framework and Prototype For Named Data Networking-Based Edge Cloud Computing System", IEEE, 57741 – 57759.
6. Youhuizi Li , Zeyong Dong, Kewei Sha, Congfeng Jiang, Jian Wan, And Yuan Wang (2019), "TMO: Time Domain Outsourcing Attribute-Based Encryption Scheme for Data Acquisition in Edge Computing", IEEE Access, Volume: 7, 40240 – 40257.
7. Abdulatif Alabdulatif , Ibrahim Khalil2, Xun Yi and Mohsen Guizani (2018)," Secure Edge of Things for Smart Healthcare Surveillance Framework", IEEE Access , Volume: 7 , 31010 – 31021.
8. Yongchuan Niu , Jiawei Zhang, An Wang and Caisen Chen (2019)," An Efficient Collision Power Attack on AES Encryption in Edge Computing", IEEE Access ,Volume: 7, 18734 – 18748.
9. Hui Cui, Xun Yi and Surya Nepal (2018)," Achieving Scalable Access Control Over Encrypted Data for Edge Computing Networks", IEEE Access, volume:6,30049 – 30059.
10. Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu and W. Lv (2019), "Edge Computing Security: State of the Art and Challenges", Proceedings of the IEEE, vol. 107, no. 8, pp. 1608-1631, doi: 10.1109/JPROC.2019.2918437.

11. J. Zhang, B. Chen, Y. Zhao, X. Cheng and F. Hu (2018), "Data Security and Privacy-Preserving in Edge Computing Paradigm: Survey and Open Issues", IEEE Access, vol. 6, pp. 18209-18237, doi: 10.1109/ACCESS.2018.2820162.
12. J. Pan and J. McElhannon, "Future Edge Cloud and Edge Computing for Internet of Things Applications," in IEEE Internet of Things Journal, vol. 5, no. 1, pp. 439-449, Feb. 2018, doi: 10.1109/JIOT.2017.2767608.
13. [13] Zhu, R., Liu, L., Song, H., & Ma, M. (2020), "Multi-access edge computing enabled internet of things: advances and novel applications", Neural Comput & Applic, doi:https://doi.org/10.1007/s00521-020-05267-x.
14. N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob and M. Imran (2018), "The Role of Edge Computing in Internet of Things", IEEE Communications Magazine, vol. 56, no. 11, pp. 110-115, doi: 10.1109/MCOM.2018.1700906.
15. Y. Zhou, D. Zhang and N. Xiong (2017), "Post-cloud computing paradigms: a survey and comparison," in Tsinghua Science and Technology, vol. 22, no. 6, pp. 714-732, doi: 10.23919/TST.2017.8195353.
16. Niu, X., Shao, S., Xin, C., Zhou, J., Guo, S., Chen, X., & Qi, F (2019), "Workload Allocation Mechanism for Minimum Service Delay in Edge Computing-Based Power Internet of Things," in IEEE Access, vol. 7, pp. 83771-83784, doi: 10.1109/ACCESS.2019.2920325.
17. Yahuza, M., Idris, M. Y., Wahab, A. W., Ho, A. T., Khan, S., Musa, S. N., & Taha, A. Z. (2020), "Systematic Review on Security and Privacy Requirements in Edge Computing: State of the Art and Future Research Opportunities," in IEEE Access, vol. 8, pp. 76541-76567, doi: 10.1109/ACCESS.2020.2989456.
18. Chen, S., Wen, H., Wu, J., Lei, W., Hou, W., Liu, W., Xu, A., & Jiang, Y. (2019),"Internet of Things Based Smart Grids Supported by Intelligent Edge Computing," in IEEE Access, vol. 7, pp. 74089-74102, , doi: 10.1109/ACCESS.2019.2920488.
19. N. Abbas, Y. Zhang, A. Taherkordi and T. Skeie, "Mobile Edge Computing: A Survey (2018) ," in IEEE Internet of Things Journal, vol. 5, no. 1, pp. 450-465, doi: 10.1109/JIOT.2017.2750180.
20. Hu, C., Song, X., Liu, P., Xin, Y., Xu, Y., Duan, Y., & Hao, R. (2019), "Forward Secure Conjunctive-Keyword Searchable Encryption," in IEEE Access, vol. 7, pp. 35035-35048, doi: 10.1109/ACCESS.2019.2902855.
21. S. Wang, S. Jia and Y. Zhang (2019), "Verifiable and Multi-Keyword Searchable Attribute-Based Encryption Scheme for Cloud Storage," in IEEE Access, vol. 7, pp. 50136-50147, doi: 10.1109/ACCESS.2019.2910828.
22. Siva Kumar, D.V.N., Santhi Thilagam, P. (2019), "Searchable encryption approaches: attacks and challenges", Knowl Inf Syst, vol. 61, pp. 1179–1207 ,https://doi.org/10.1007/s10115-018-1309-4
23. Kim, S., Lee, I. (2018), "IoT device security based on proxy re-encryption", J Ambient Intell Human Comput , vol.9, pp.1267–1273, https://doi.org/10.1007/s12652-017-0602-5
24. Luo, W., Ma, W. Secure and efficient proxy re-encryption scheme based on key-homomorphic constrained PRFs in cloud computing (2019), Cluster Comput, vol. 22, pp.541–551, https://doi.org/10.1007/s10586-018-2862-z
25. Chengyu Hu,Yuqin Xu, Xiangfu Song, Pengtao Liu, Yue Xin, Yuyu Duan, and Rong Hao (2019), "Forward Secure Conjunctive-Keyword Searchable Encryption", IEEE, Volume:7, 2169-3536
26. Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu and W. Lv, "Edge Computing Security: State of the Art and Challenges," in Proceedings of the IEEE, vol. 107, no. 8, pp. 1608-1631, Aug. 2019, doi: 10.1109/JPROC.2019.2918437.
27. S.Prasath,K.Nirmala,"A Performance Comparison of Authentication and Privacy Preserving Techniques for Secured Communication in VANET", International Journal of Innovative Technology and Creative Engineering(ISSN:2045-8711), Vol.9, No.2,Pp.625-635 ,2019.
28. S.Prasath, K.Nirmala, "Adaptive Boosting Classifier Based Attack Detection for Secured Communication in
29. Vanet", International Journal of Advanced Science and Technology(ISSN:2005-4238), Vol.28, No.17,Pp.No.168-177,2019. (Scopus Indexed Journal Sl.No.16260)
30. S.Prasath, K. Nirmala, "Probabilistic Mceliece Public-Key Cryptography Based Identity Authentication for Secured Communication in Vanet", Solid State Technology Vol.63 Iss.6, Pages 10167-10182.