International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 08, 2020
ISSN: 1475-7192

A Secure Cloud-Edge-Shared Storage System for Data Sharing and Searching on Edge Servers

[1]Mrs. D. Naga Swetha, [2]Mr. Anil Tellur

ABSTRACT-- *Ensuring and preserving the data privacy on the cloud and offering security for data sharing between IoT devices and users for pointed of data between the cloud and users on edge servers is made possible with Cloud-Edge Shared Storage System (CESSS). This system effectually diminishes the computing load of IoT devices and users by allocating computation-intensive encryption and decryption algorithms to edge servers. Substantial reduction of the computing and communication overhead for generating keyword search trapdoors is beneficial with reference to maintaining the privacy of IoT devices and user's private keys and attaining further protected or resourceful data sharing and data searching.*

Keywords-- *Cloud-Edge Shared Storage System, Edge servers, Edge Computing, IoT Devices, Data Sharing, Data Searching.*

I. INTRODUCTION

Cloud-Edge Shared Storage System (CESSS) is a framework designed to progress data of the internet of things and allow edge servers to develop and process IoT data in real-time and stores them on a cloud server. It can quickly respond to the requests of IoT devices, provide a massive volume of cloud storage for IoT data, and conveniently share IoT data with users. The susceptibility of edge and cloud servers gives rise to risk of data leakage. Explored secure data search and sharing scheme to improve the existing secure schemes by generation of public-and-private key pair and handling private keys by themselves and by using searchable public key encryption to attain more secure, efficient, and flexible data searching[1]. From security perspective, this scheme confirms the confidentiality of cloud data and secure data sharing and avoids a single point of breakthrough.

Figure 1: Depicting CESSS flow

[1] *Assistant Professor, swetha@gnits.ac.in*

[2] *Assistant Professor, aniltellur@gnits.ac.in,Department of Computer Science and Engineering, G. Narayanamma Institute of Technology and Science (For Women), Hyderabad, Telangana, INDIA.*

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020

CESSS scheme significantly reduces users' computing costs and communication overhead by delegating most of the cryptographic operations to edge servers. CESSS helps to train edge servers between Internet of Things (IoT) devices and cloud servers. The edge servers analyses data collected from IoT devices in real-time and forward processed data to the cloud server to save the cost of IoT devices.

II. RELATED WORK

Edge computing is thriving with the rapid development of cloud computing as a new computing paradigm. Accordingly, data confidentiality in the cloud and edge computing are seizing the attention from academia and industries as well[2].

Cloud Security: Attribute-based encryption was made involved into cloud computing to achieve fine-grained access control over outsourced encrypted data. The problematic role played here is to enable a semi-trusted cloud to compute between ciphertexts while guaranteeing the privacy of the encrypted data has also attracted significant attention. How to perform a secure search over encrypted data. A public-key encryption with keyword search (PEKS) scheme for a single keyword search [9]. In addition to confidentiality, remote data integrity [4] is another concern for secure outsourcing storage.

Edge Security: Constructing a prototype applying secure enclave technologies on edge devices to enforce security isolation. To achieve secure communication in edge environments, Pimentel et al. [3] proposed a secure communication protocol for federated content networks.

Cloud-edge-collaborative Security: Protect the data privacy of outsourced storage in the cloud-and-edge-assisted IoT. This work demonstrated that by deploying searchable encryption (SE) along with another cryptographic algorithm. All edges share the data-search secret key, which easily leads to the problem that any edge server could be compromised and then leveraged to break the security of the entire system. This system utilizes symmetric encryption, public key encryption, digital signature, and PEKS to realize the function and security goals. Symmetric encryption (SE) is a cryptographic primitive that encrypts data or decrypts ciphertexts with the same secret key[5]. Symmetric encryption (SE) algorithm is composed of SE.Setup, SE.Enc and SE.Dec algorithms. Public key encryption (PKE) is a cryptographic primitive that has different encryption key and decryption key. Furthermore, users can publish the encryption key, and the decryption must be kept secret.

III. PROBLEM IDENTIFICATION

The authorized and authenticated users can accessibly share IoT data stored on the cloud server with the help of edge servers. In this case, IoT devices first collect and upload data to nearby edges. Second, edges process IoT data in real-time, return the result and store IoT data on a cloud server. Finally, users can share expected IoT data on the cloud server by submitting corresponding search requests. This scheme is advantageous in reducing the cost of IoT devices by delegating cryptographic operations to edge servers [6]. Still, it has few problems in practice, like, edge servers can know the private keys of IoT devices. If an edge server is compromised, then it can be used to falsify IoT data. Next, edge servers trust each other and share their data-search secret keys. By this, a compromised edge server can be used to generate search trapdoors with arbitrary keywords and retrieve expected

data from the cloud server. Lastly, suppose that a (mobile) IoT device can upload its data via different and uncertain edge servers, that the nearby edge server must fetch the data-search secret keys of all edge servers to retrieve the expected data for an authorized user. In practice, the communication cost is vast if there are many edge servers.

IV. CESSS MODEL

Cloud-Edge Shared Storage System (CESSS) scheme came in to existence by allowing IoT devices and users to generate their public-and-private keys by themselves, and all private keys are finally known only by their generators which can avoid vulnerable activities. CESSS scheme achieves more secure private key management and resists data forgery[7]. It applies searchable public-key encryption instead of searchable symmetric-key encryption and make users generate keyword search trapdoors with their private keys and keywords, and a compromised edge server cannot be used to generate keyword search trapdoors.

CESSS scheme has five entities: IoT devices, users, edges, cloud, and certificate authority (CA). IoT devices can store their data on the cloud through nearby edges. Users can download or retrieve the data shared by IoT devices from the cloud through nearby edges. Edges encrypt data or decrypt ciphertexts for IoT devices and users and communicate with the cloud. The cloud is responsible for storing ciphertexts generated by edges and returning the data that the edges request. The CA issues digital certificates of IoT devices and Users.

V. CESSS WORKFLOW

Users Setup phase: Registration of IoT devices and users done at CA and certificates storing is looked after by CA.

Data Uploading phase: An IoT device sends its data and the equivalent keywords to an adjacent edge server and encrypting the data along with storing ciphertexts on the cloud.

Data Sharing phase: An authorized and authenticated user submits a sharing request to a nearby edge server and the edge requests and downloads the encrypted data of the requested IoT devices from the cloud, decrypts them and verifies their integrity. Finally, the edge returns the decrypted data to the user.

Data Search phase: An authorized user generates a search request to the cloud through a nearby edge. The cloud searches for the matched ciphertexts and sends them to the edge. The edge decrypts them and verified their integrity. Finally, the edge returns the decrypted data to the user[11].

CESSS Security Goals: CESSS model achieves the following security goals:

Confidentiality: Sustaining with the confidentiality of the IoT device data stored on the cloud and protecting data from outside attackers.

Securely Sharing data: This need only authorized users to download the shared data and verify the data integrity through edges.

Securely Retrieving data: This goal requires that only the authorized users are allowed to retrieve the shared data, and no important distinctive information about the shared data is trickled to the cloud.

Avoiding a single point of breakthrough: If an outside attacker attacks an edge, it cannot destabilize the security of other edges.

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020

VI. EVALUATING CESSS SYSTEM

Setup Phase: In the Setup phase, IoT devices and users makes a public-and-private key pair and then registers the public key to the certificate authority (CA), maintaining secrecy of private key.

Figure2: Setup Phase action flow

Public Key Encryption Scheme (PKE): Considering an input security parameter for suppose 1's and providing an output of public as well as private key pairs, followed by encryption and decryption procedures as usual.

Public Key Encryption with Keyword Search (PEKS): Considering an input security parameter for suppose 1's and providing an output of public as well as private key pairs, followed by encryption and decryption procedures as usual.

Input as public key and random keyword makes a searchable cipher text as approximate output.

Keyword and private key as input and output as keyword trapdoor and searching is followed by sequence of cipher texts.

Data Uploading Phase: Here, signing of an IoT device with data and its private key. It sends the data, the corresponding signature, the extracted keywords from the data along with its certificate and authorized users info to a nearby edge server through a secure channel. To secure shared data, the edge server encrypts the secret key with the public key of the authorized users. To support secure data retrieval, PEKS provides additional support to generate keyword searchable ciphertexts with the authorized users' public keys.

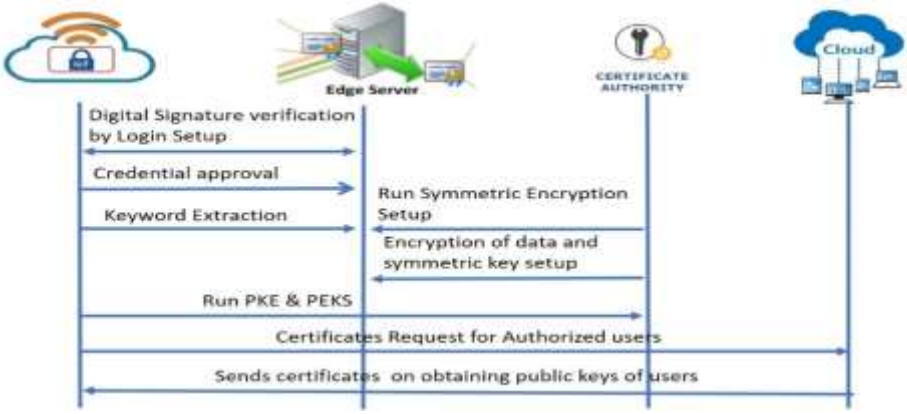

Figure 3: Data Uploading Phase action flow

Data Sharing Phase: In the Data Sharing phase, an authorized user submits a data sharing request to the nearby edge server. The edge server requests and obtains the corresponding ciphertexts from the cloud and sends the contained PKE ciphertext to the authorized user. Next, the authorized user decrypts the secret key and uses a secure channel to send this key to the edge server[8]. After receiving the secret key, the edge decrypts data and uses a secure channel to return the data to the authorized user if the data signature is valid.

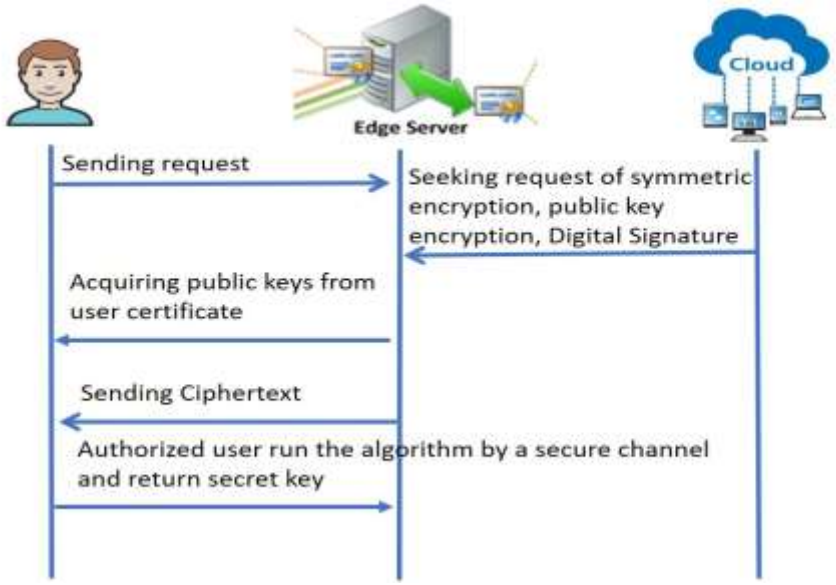

Figure 4: Data sharing Phase action flow

Data Search Phase: Here, an authorized user performs keyword search trapdoor generation with private key and desired keyword. Next, submits this trapdoor and certificate to the cloud through a nearby edge server. Now, the cloud searches for the matched ciphertexts and sends them to the edge server[10]. The edge server decrypts the requested data as it does in the Data Sharing phase and returns the data to the user through a secure channel if it is a valid data.

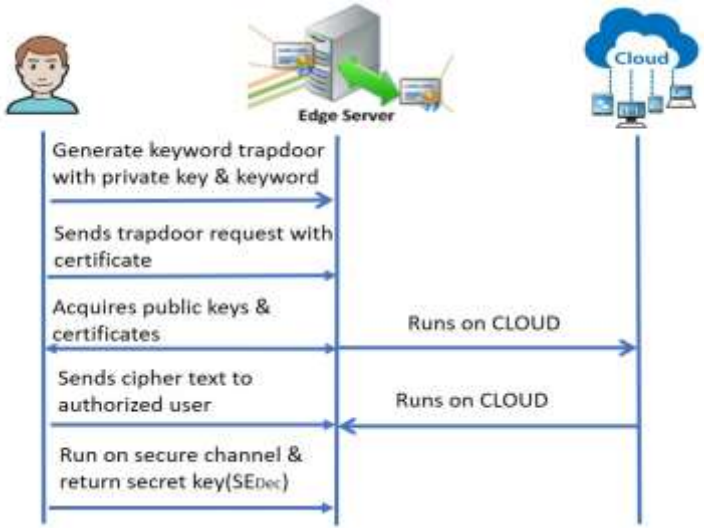

Figure 5: Data Search Phase action flow

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020

VII. EXPERIMENTAL RESULTS

In the Data Uploading phase, CESSS requires the IoT device to generate the data signature and upload the data, the extracted keywords, the authorized users, the generated signature, and its certificate to the nearby edge server. It enables stronger security upon its requirements like additional computing and communication overhead to upload data. In the Data Sharing phase, it requires the edge server to send the shared PKE ciphertext to the authorized user. Next, the authorized user decrypts the received ciphertext with his private key and returns a data-sharing secret key to the edge server. In the Data Search phase, the authorized user only needs to generate one keyword search trapdoor. Therefore, it reduces the communication and computing overhead for generating keyword search trapdoors.

Table 1: Comparisons of communication costs.

Scheme	**Phase**	**Communication Trips**				
		IoT Device	Edge (near to the IoT device)	Cloud	Edge (near to the user)	User
CESSS Scheme	Data Uploading	1	4	3	0	0
	Data Sharing	0	0	2	8	4
	Data Search	0	0	4	8	4

Table 1 shows a comparison of the communication cost of CESSS scheme.

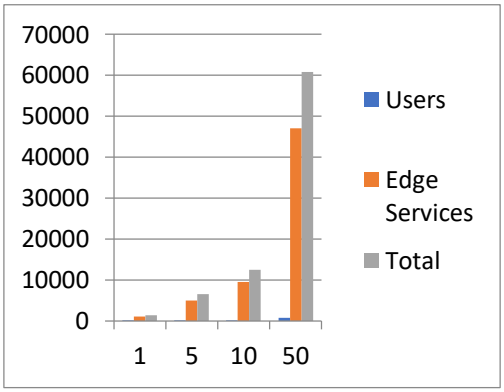

Figure 7: Time cost of CECS scheme in **the** Data Uploading phase.

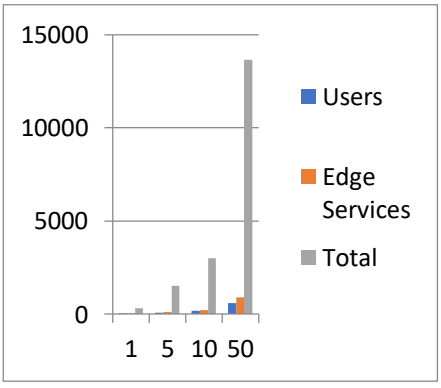

Figure 8: Time cost of our CECS scheme in the Data Sharing phase.

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020

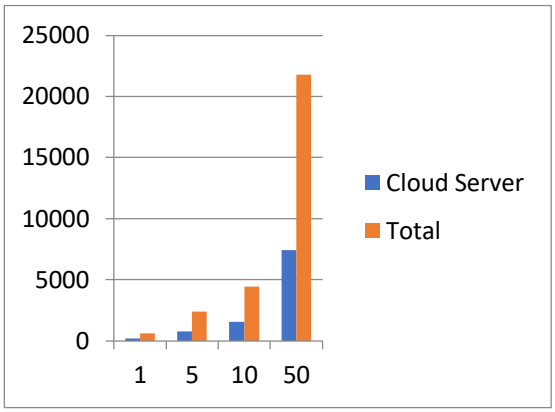

Figure 9: Time cost of our CECS scheme in the Data Search phase.

VIII. CONCLUSION

In terms of performance, the experimental results depict that CESSS scheme effectively reduces the computing burden of IoT devices and users by delegating computation-intensive encryption and decryption algorithms to edge servers. CESSS scheme significantly reduces users' computing costs and communication overhead by delegating most of the cryptographic operations to edge servers. CESSS helps to train edge servers between Internet of Things (IoT) devices and cloud servers where they analyse data collected from IoT devices in real-time and forward processed data to the cloud server to save the cost of IoT devices.

REFERENCES

1. M. B. Mollah, M. A. K. Azad, A. Vasilakos, "Secure data sharing and searching at the edge of cloud-assisted internet of things", IEEE Cloud Comput., vol. 4, no. 1, pp. 34-42, 2017.
2. S. Kamara, C. Papamanthou, T. Roeder, "Dynamic searchable symmetric encryption", Proc. ACM Conf. Comput. Commun. Secur., pp. 965-976, 2012.
3. H. M. Pimentel, S. Kopp, M. A. Simplicio, R. M. Silveira, G. Bressan, "OCP: A protocol for secure communication in federated content networks", Comput. Commun., vol. 68, pp. 47-60, 2015.
4. M. Sookhak et al., "Remote data auditing in cloud computing environments: A survey taxonomy and open issues", ACM Comput. Surv., vol.47, no. 4, pp. 65, 2015.
5. YE TAO, PENG XU, and HAI JIN1, "Secure Data Sharing and Search for Cloud-Edge-Collaborative Storage", IEEE Cloud Comput., vol 4, pp.1-10, 2016.
6. N. Kaaniche, M. Laurent, "SHoPS: Set Homomorphic Proof of Data Possession Scheme in Cloud Storage Applications", Proc. IEEE World Congr. Services, pp. 143-150, 2015.
7. . Ghareh Chamani, D. Papadopoulos, C. Papamanthou, R. Jalili, "New constructions for forward and backward private symmetric searchable encryption", Proc. ACM SIGSAC Conf. Comput. Commun. Secur., pp. 1038-1055, 2018.
8. C. B. Tan, M. H. A. Hijazi, Y. Lim, A. Gani, "A survey on Proof of Retrievability for cloud data integrity and availability: Cloud storage state-of- the-art issues solutions and future trends", J. Netw. Comput. Appl., vol. 110, pp. 75-86, 2017.

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020

9. D. Naga Swetha, “Conjunctive Keyword Search (CKS) technique using Series of N-gram filters for Security and Privacy in Cloud” ,Jour of Adv Research in Dynamical & Control Systems, Vol. 10, 07-Special Issue, 2018.
10. A. D. Caro, The Java Pairing Based Cryptography Library, [online] Available: http://libeccio.di.unisa.it/projects/jpbc/.
11. P. Xu, Q.Wu,W.Wang,W. Susilo, J. Domingo-Ferrer, H. Jin, "Generating searchable public-key ciphertexts with hidden structures for fast keyword search", IEEE Trans. Inf. Forensics Security, vol. 10, no. 9, pp. 1993-2006,2015.

Received: 27 Feb 2019 | Revised: 20 Mar 2019 | Accepted: 30 Apr 2020