ISSN- 2394-5125 VOL 9, ISSUE 04, 2022

# Effective Management with Flexibility to Access Authorized Data from Cloud based IoT

P Adi lakshmi,[1]E Maheswari[2], G Vineela[3], R Mounika[4], A Manogna[5]

[1]Asst. Professor, Department of Computer Science and engineering

[2,3,4,5]Student, Department of Computer Science and engineering

[1,2,3,4,5]QIS College of Engineering and Technology, Ongole-523001

**Abstract**-Data confidentiality and access control in the cloud may be achieved via attribute-based encryption. Attribute privacy is exposed when obvious characteristics are added to the ciphertext to help users locate accessible data in large datasets. The anonymous key-policy attribute-based encryption (AKP-ABE) is extended in this research to provide fine-grained data retrieval while maintaining attribute privacy to create an efficient attribute-based access control with authorised search scheme (EACAS). Data users may build search rules based on access policies and generate the trapdoor using the secret key supplied by data owners in EACAS, which allows for retrieval of data that is relevant to them. As a result, a virtual attribute with no semantic value is used in data encryption and trapdoor creation to enable the cloud to do an attribute-based search on the outsourced encrypted data. By setting the search rules, data users are able to search for interesting data based on protected characteristics while data owners may accomplish fine-grained access control on outsourced data. Finally, we show that EACAS is more efficient in terms of compute and storage overheads than currently available methods.

**Keywords—** Access control, authorized search, cloud storage, data sharing, key-policy attribute-basedencryption.

## I. INTRODUCTION

Internet of Things (IoT) management services such as Amazon AWS IoT [1] or Google Cloud IoT Core [2] outsource the IoT data to cloud services. IoT devices in cloud-based management systems usually belong to distinct trust domains with complicated, asymmetrical trust relationships, such as the internet of things. Because of this, it is difficult to control access to outsourced IoT data from a single security domain. As a potential approach, attribute-based encryption (ABE) allows safe and fine-grained access control over encrypted data in accordance with the rules connected to it [3]. An encryptor may establish an access policy for the ciphertext using a collection of descriptive characteristics in ciphertext-policy ABE (CP-ABE) [4]. Using a decryption key, an attacker can only decipher the plaintext if the access privileges granted by his secret key match those in the ciphertext.

There are a number of concerns that need to be resolved before CP-ABE may be used in cloud-based IoT administration. To begin, the ciphertext develops in size in a manner that is proportional to the number of characteristics [5, 6, 7]. In IoT systems, this might be a major issue because of the many qualities required by IoT applications and services [8]. IoT systems can't be protected by CPABE schemes that only allow constant-size ciphertexts, even if they exist [9, 10, 11, 12].

A second advantage of CP-ABE is that it places high computing demands on a decryptor (rather than an encryptor), which may be battery-operated mobile devices like laptops. An untrusted cloud server may decode portion of ciphertexts on behalf of users, according to recent research in this area. Because of this, ciphertext volume could not be solved. It may be possible to combine current technologies such as outsourceable decryption [5] and constant-size cypher text ABE schemes for each function in order to deal with the aforementioned concerns A key blinding mechanism in the outsourced decryption algorithm [5] prevents it from solving the challenge. A user's private key may be obscured using a (secret) blinding factor, say z, such that the cloud can partially decode the data using the obscured key and return the plaintext obscured by z, specifically. As a result, z is all that is needed to uncover it. But when used with constant-size encryption text [11], this approach masks not just the plaintext but also additional parts that are required for decryption, but which the user does not know about. As a result, the user will never be able to accurately retrieve the plaintext. To accomplish both small ciphertext and outsourceable decryption functionality, Li et al. [16] devised a technique. Nevertheless, their technique severely restricts the ability to manage access, since only users whose characteristics match those in the access policy may decode a cypher text. As a last point, secret keys are vulnerable to being misused by unauthorised users who share their private keys. Secret keys may be unlawfully shared between authorised and unauthorised users in this situation. It would then be possible for anybody to access IoT data on the cloud. Traceable ABE was proposed as a solution to the leakage issue in previous investigations. Sadly, the majority of traceable ABE schemes simply look for the original owners of the keys. That example, if dishonest people exchange secret keys, the shared key holders still have access to encrypted cloud data. Due to realistic key recovery techniques such as side channel analysis, traceability alone is not enough to prevent key leakage. When it comes to the shared (or leaked) key issue, key revocation is not a viable solution, since the key holders will still be able access and retrieve data until it is revoked. In cloud-based IoT management systems, establishing a secure and efficient access control mechanism that addresses the aforementioned issues is critical. Using our new CPABE structure, we provide an effective and secure cloud-based IoT data management system in this study. The suggested system has the potential to track traitors who unlawfully distribute their secret keys, as well as efficient storage and bandwidth management. A user-specific transformation key enables the cloud server to handle a considerable portion of the computational cost associated with decryption. In order to prevent unwanted access by a shared (or leaked) key holder, the cloud authenticates the identity of the key holder, partly decrypts the ciphertext using the key holder's transformation key, and finally provides the partially decrypted result. As a reminder, the transformation key is intimately linked to the original owner of the attribute key, who may decode the partly encrypted cypher text and retrieve plaintext only if he is the original owner. The plaintext can't be accessed by anybody else using the shared (or leaked) keys. Against summarise, the proposed system is impervious to forensically intractable key abuse assaults.. If the properties of the shared (or leaked) key are in line with the access policy associated to the encrypted IoT data, then this property holds. A shared (or leaked) key prevents users from accessing the IoT data. Only the original key holder can decrypt the message correctly. Our CP-ABE scheme, which allows for outsourced decryption and key traceability as well as constant-size ciphertext, is the foundation for this resiliency method.

## II. RELATEDWORKS

ABE schemes with outsourceable decryption [5] and constant-size ciphertext [11] may be combined to address the aforementioned concerns. A key blinding mechanism in the outsourced decryption algorithm [5] prevents it from solving the challenge. Users may blind their own secret key by applying a (secret) blinding factor known as z, which allows the cloud to do partial decryption using this blinded key, and then return plaintext masked by z. As a result, z is all that is needed to uncover it. Constant-size ciphertext, on the other hand, is disguised not only by z but also by additional components that are required for decryption but unknown to the user when using this approach [11]. As a result, the user will never be able to accurately retrieve the plaintext. To accomplish both small ciphertext and outsourceable decryption functionality, Li et al. [16] devised a technique.

Nevertheless, their technique severely restricts the ability to manage access, since only users whose characteristics match those in the access policy may decode ciphertexts.Data from IoT sensors can be outsourced to the clouds via cloud servers and stored, exchanged, and processed via centralised or decentralised servers in the cloud, which makes these IoT systems vulnerable to both internal and external attacks. Cloud computing has contributed to the success of the Internet of Things by providing abundant storage and computation resources. Some cryptographic algorithms have been used to safeguard IoT data from possible harmful users and adversaries in order to maintain its confidentiality and integrity. Even if the data is encrypted, it is difficult to do any arithmetical operations. In theory, lattice-based fully-homomorphic encryption may offer a solution, but since it is computationally intensive, it cannot be deployed to the Internet of Things. When a semi-trusted server is involved, full-homomorphic encryption is possible. A distributed system for sharing IoT data, on the other hand, is difficult to implement. A fully-homomorphic encryption technique for cloud-based IoT applications has been developed to address this issue. A semi-trusted server may be used to assist in the calculation of homomorphic multiplications without getting any helpful information from the encrypted input.The term "e-Health" refers to the usage of the Internet and other linked networks in a healthcare system. It was our goal in this study to look at studies from 2017–2020 to see how the integration of Internet of Things (IoT) devices and cloud computing has changed the way intelligent technologies are used in health care. Health information produced from electronic sources and gained knowledge, as characterised by the term "e-health," may be used to diagnose, treat, and prevent disease. The Internet has the ability to safeguard consumers from damage and allow them to fully engage in informed health-related decision-making as a storehouse for health information and e-Health analysis. High e-Health integration levels reduce the danger of encountering false material on the Internet. IoT-cloud-based eHealth systems are evaluated from a variety of viewpoints, with a focus on the prospects, advantages, and obstacles of their deployment. Combining the Internet of Things and cloud computing with eHealth systems that are based on smart goals and applications is an exciting new trend. The Internet of Things has several privacy and security concerns (IoT). The

Internet of Things (IoT) has a number of issues, including a lack of effective and strong security mechanisms, user ignorance, and the well-known active device monitoring. By investigating the history of Internet of Things (IoT) systems and security measures, we are able to better understand: (a) different security and privacy issues; (b) approaches used to secure IoT-based environments and systems; (c) existing security solutions; and (d) the best privacy models necessary and suitable for different layers of IoT-driven application. IoT layered model: general and extended privacy and security components and layers identification were presented in this study. The suggested cloud/edge backed IoT system is put into action and assessed.

Amazon Web Service (AWS) Virtual Machines constitute the bottom layer of the IoT nodes. With the Greengrass Edge Environment on AWS, a Raspberry Pi 4 hardware kit was used to implement the intermediate layer (edge). AWS's cloud-enabled IoT ecosystem was utilised to build the top layer of our solution (the cloud). Users' information was protected by the security protocols and crucial management sessions that were placed between each of these levels. In order to facilitate data movement across the levels of the proposed cloud/edge enabled IoT paradigm, we introduced security certificates. With the best security methodologies, the suggested system model may be utilised in conjunction with the proposed system model to countermeasure cybersecurity risks at each tier; cloud, edge, and IoT. As a result of Cloud Computing (CC) and Internet of Things (IoT) integration, treatment procedures have been radically altered in the ubiquitous computing era. As the volume of data created by IoT devices grows, so does the demand for a storage and processing infrastructure like the CC. Despite the fact that users and IoT devices continue to share computing and networking resources remotely, security challenges in CoT remain increasingly significant.

Furthermore, maintaining data privacy in such a setting is a top priority. As a result, the CoT is constantly expanding its focus on security and privacy. In this article, we examined some of the issues around data security and privacy and how they may be addressed. The CoT architecture and current applications have been examined in order to attain this goal. In addition, this study addresses a variety of security and privacy concerns and difficulties, as well as unresolved obstacles.

## III. PROPOSED SYSTEM ARCHITECTURE

Using our new CPABE structure, we offer a cloud-based IoT data management system that is both efficient and safe. The suggested system has the potential to track traitors who unlawfully distribute their secret keys, as well as efficient storage and bandwidth management. A user-specific transformation key enables the cloud server to handle a considerable portion of the computational cost associated with decryption.

In order to prevent unwanted access by a shared (or leaked) key holder, the cloud authenticates the identity of the key holder, partly decrypts the ciphertext using the key holder's transformation key, and finally provides the partially decrypted result. You should be aware that the attribute key is intimately linked to the key holder, who can only decipher the plaintext if he is also the original owner of the attribute key. The plaintext can't be accessed by anybody else using the shared (or leaked) keys.To begin, the owner of the data must first sign up for an account on the cloud server and get authorization. Data owner will encrypt and upload file to cloud server after receiving consent from cloud data owner, and data owner will then request content key and master secret for file he uploaded and discovers Only when the keys have been produced can the file be sent to the cloud server for deduplication. After uploading a file, the data owner must grant the user access to search and download the file, as well as authorization to share the material. The cloud server is in charge of running and maintaining a cloud for the purpose of storing data. Data owners encrypt their data files and put them in the cloud for cloud End users to access and collaborate on. A content key and a master secret key are required for users to access shared data files. And the cloud will provide access to the data, as well as monitor all transactions and attacks associated to them.. The end user requests a content key and a secret key from Key Authority. The content key and master secret key produced with the associated data owner information of a certain file may be seen by KeyAuthority on all files. To see files stored in the cloud, the user must first create an account and log in.

The cloud has granted the user permission to validate the user's registration. To download the file, the user must request the MSK master secret key and the content key. As long as the data owner allows it, users may download and search their files.

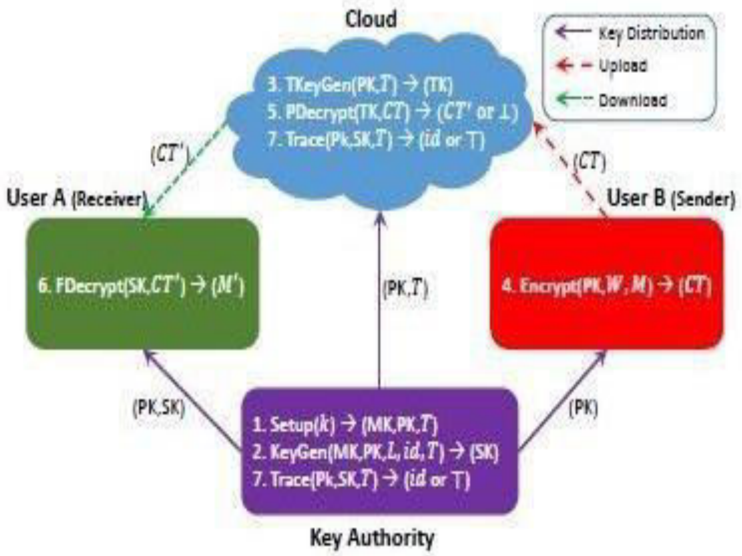
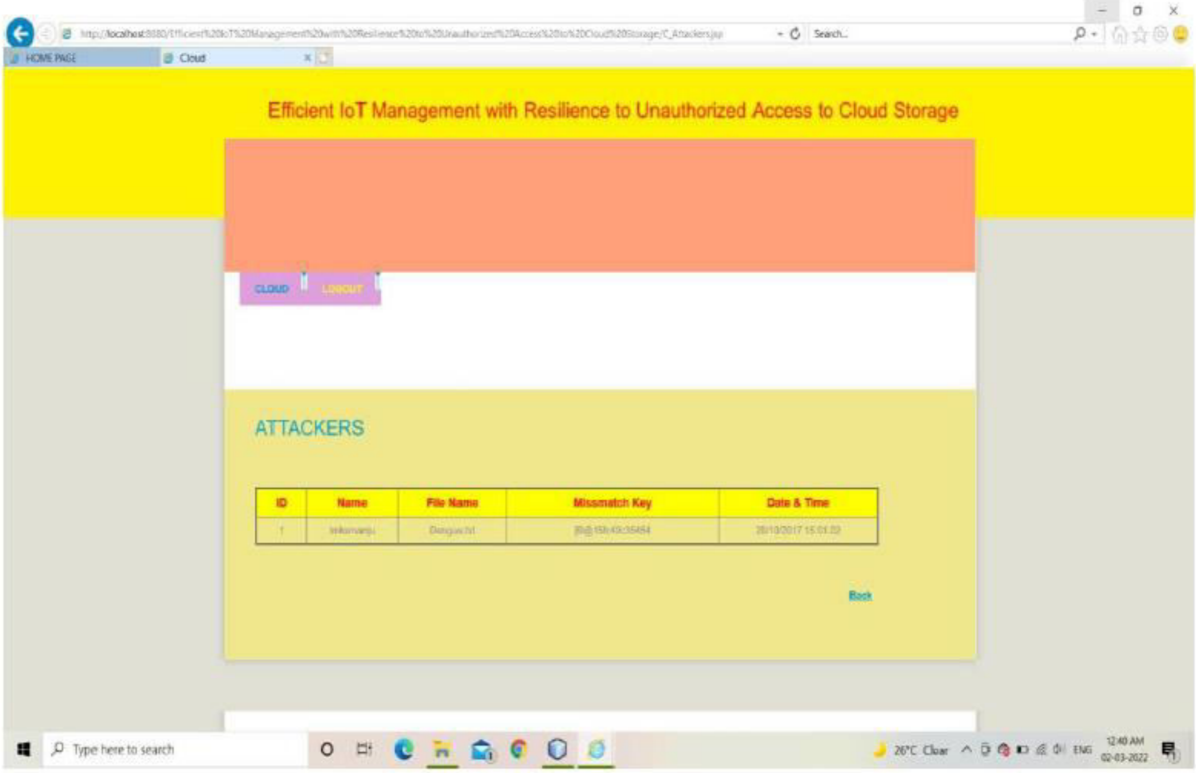

Cloud
Key Distribution
Upload
Download
3. TKeyGen(PK,T) → (TK)
5. PDecrypt(TK,CT) → (CT' or ⊥)
7. Trace(Pk,SK,T) → (id or T)
(CT')
(CT)
User A (Receiver)
User B (Sender)
6. FDecrypt(SK,CT') → (M')
(PK,T)
4. Encrypt(PK,W,M) → (CT)
(PK,SK)
(PK)
1. Setup(k) → (MK,PK,T)
2. KeyGen(MK,PK,L,id,T) → (SK)
7. Trace(Pk,SK,T) → (id or T)
Key Authority

Fig:1 System Architecture

## IV. RESULTS AND DISCUSSION

The results obtained are shown from Fig. 2 to Fig.5.


http://localhost:8080/Efficient%20IoT%20Management%20with%20Resilience%20to%20Unauthorized%20Access%20to%20Cloud%20Storage/C_Attackers.jsp
Search...
HOME PAGE
Cloud
CLOUD
LOGOUT
Type here to search
26°C Clear
ENG
12:40 AM
02-03-2022

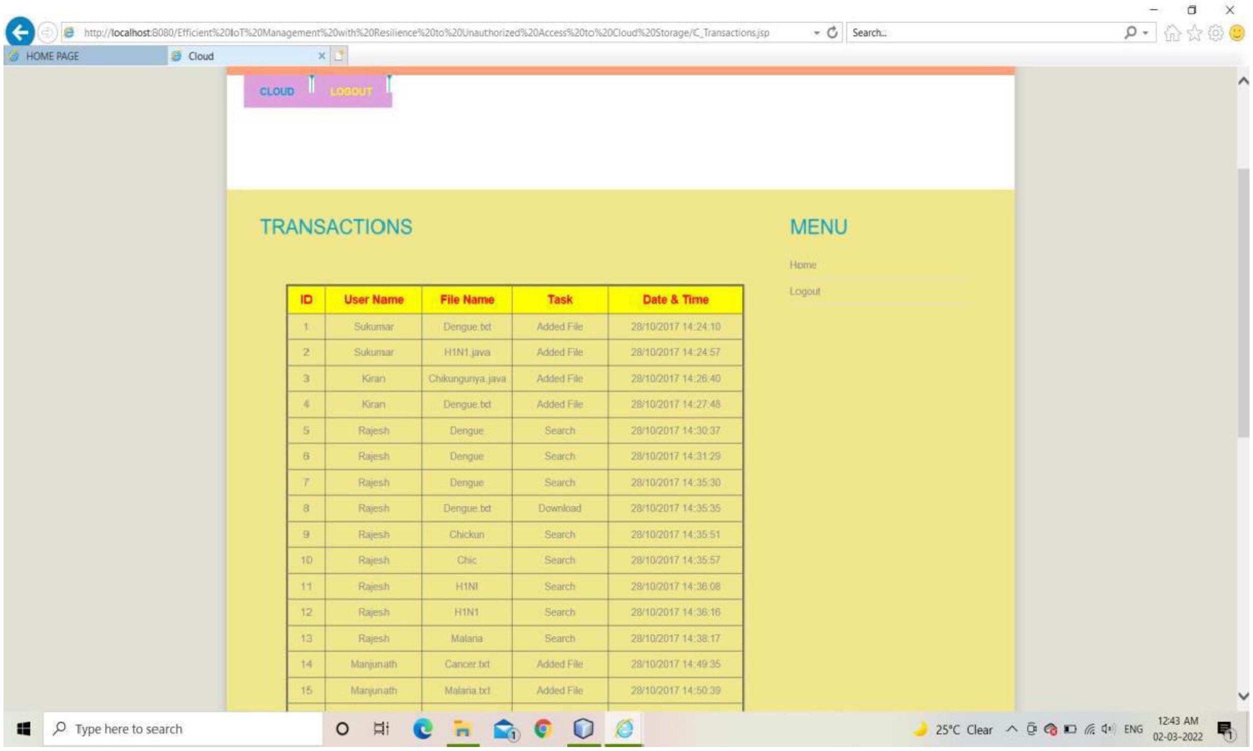
Efficient IoT Management with Resilience to Unauthorized Access to Cloud Storage

ATTACKERS

| ID | Name | File Name | Missmatch Key | Date & Time |
|---|---|---|---|---|
| 1 | tmkmanju | Dengue.txt | [B@15h49:35454 | 28/10/2017 15:01:22 |

Back

Fig.2 Attackers


http://localhost:8080/Efficient%20IoT%20Management%20with%20Resilience%20to%20Unauthorized%20Access%20to%20Cloud%20Storage/C_Transactions.jsp
Search...
HOME PAGE
Cloud
CLOUD
LOGOUT
MENU
Home
Logout
Type here to search
25°C Clear
ENG
12:43 AM
02-03-2022

TRANSACTIONS

| ID | User Name | File Name | Task | Date & Time |
|---|---|---|---|---|
| 1 | Sukumar | Dengue.txt | Added File | 28/10/2017 14:24:10 |
| 2 | Sukumar | H1N1.java | Added File | 28/10/2017 14:24:57 |
| 3 | Kiran | Chikungunya.java | Added File | 28/10/2017 14:26:40 |
| 4 | Kiran | Dengue.txt | Added File | 28/10/2017 14:27:48 |
| 5 | Rajesh | Dengue | Search | 28/10/2017 14:30:37 |
| 6 | Rajesh | Dengue | Search | 28/10/2017 14:31:29 |
| 7 | Rajesh | Dengue | Search | 28/10/2017 14:35:30 |
| 8 | Rajesh | Dengue.txt | Download | 28/10/2017 14:35:35 |
| 9 | Rajesh | Chickun | Search | 28/10/2017 14:35:51 |
| 10 | Rajesh | Chic | Search | 28/10/2017 14:35:57 |
| 11 | Rajesh | H1NI | Search | 28/10/2017 14:36:08 |
| 12 | Rajesh | H1N1 | Search | 28/10/2017 14:36:16 |
| 13 | Rajesh | Malaria | Search | 28/10/2017 14:38:17 |
| 14 | Manjunath | Cancer.txt | Added File | 28/10/2017 14:49:35 |
| 15 | Manjunath | Malaria.txt | Added File | 28/10/2017 14:50:39 |

Fig.3 Transactions

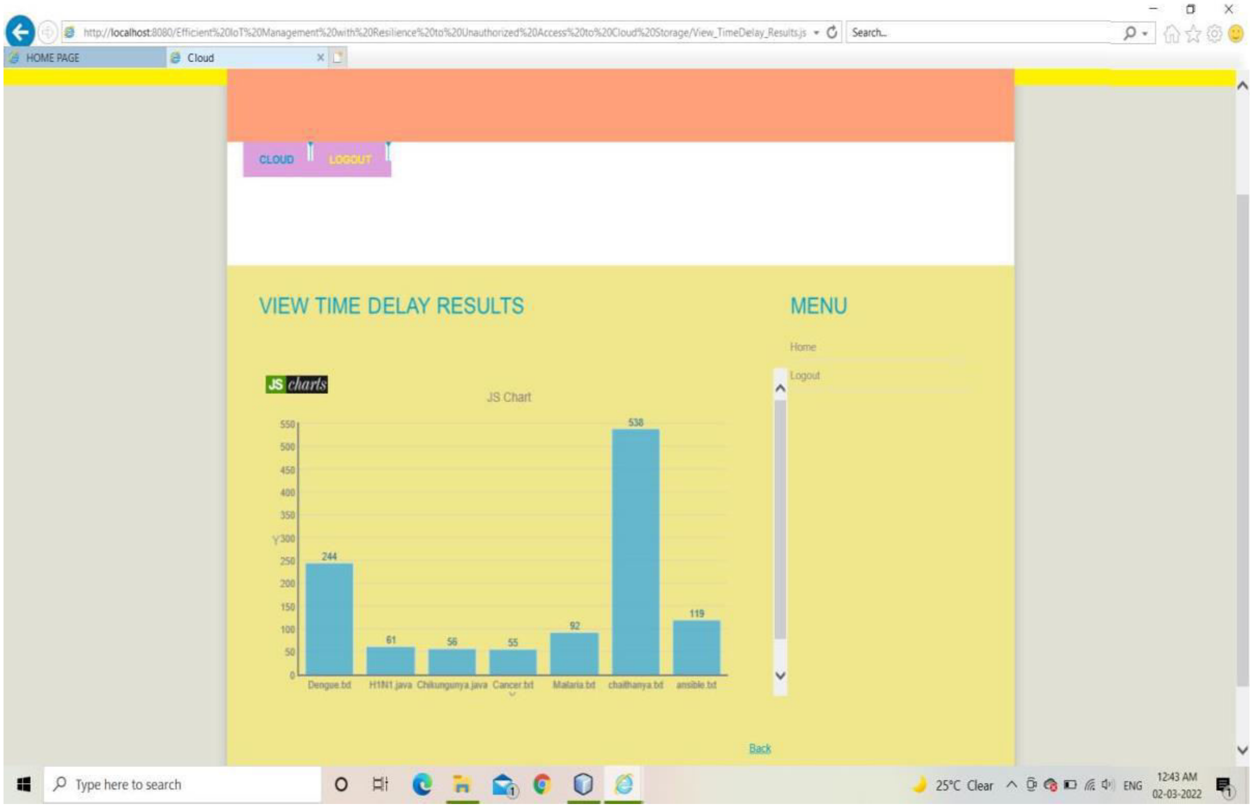


Fig.4 View Time Delay results

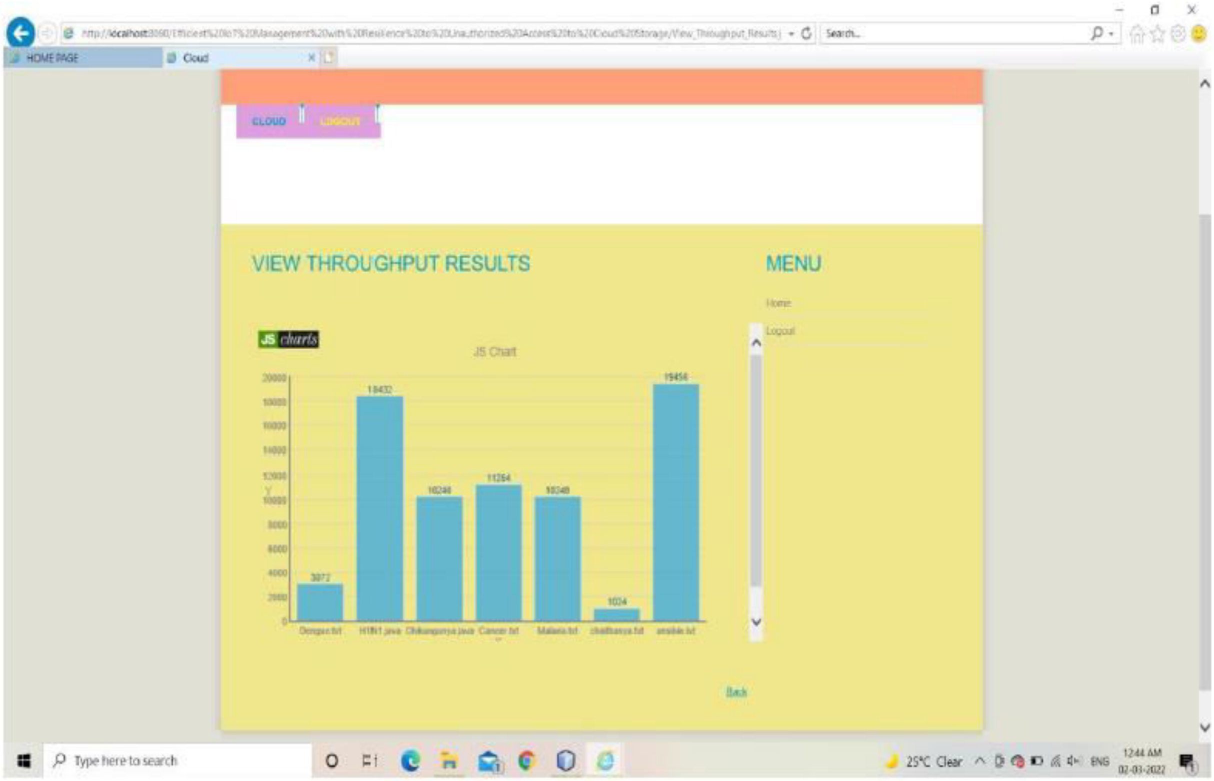


Fig.5 View Throughput results

## V. FUTURE SCOPE AND CONCLUSION

Cloud-based IoT management systems may benefit from a revolutionary CP-ABE method. IoT devices create a consistent size ciphertext regardless of the amount of characteristics, making the proposed technique efficient in terms of transmission costs. Battery-powered user devices may transfer a considerable portion of decryption efforts to the cloud under the suggested system. Key traceability ensures that only the original owner of a given key may decode an encrypted file, preventing unlawfully shared key holders from accessing the material that has been outsourced. In IoT systems, forensically intractable key abuse assaults are common, and the suggested approach can withstand them.

## REFERENCES

[1] “AWS IoT.” AWS. https://aws.amazon.com/ko/iot. (accessed Dec. 17, 2019)

[2] “Cloud IoT Core.” Google Cloud. https://cloud.google.com/iotcore. (accessedDec. 17, 2019)

[3] A. Sahai, B. Waters, “Fuzzy identity-based encryption,” Advances in Cryptology-EUROCRYPT, 2005, pp. 457–473.

[4] J. Bethencourt, A. Sahai, B. Waters, “Ciphertext-policy attributebasedencryption,” In IEEE symposium on security and privacy (SP’07), 2007, pp. 321–334.

[5] M. Green, S. Hohenberger, B. Waters, “Outsourcing the decryption of ABEciphertexts,” In USENIX Security Symposium, Vol. 2011, No. 3, 2001.

[6] J. Lai, R. H. Deng, C. Guan, J. Weng, “Attribute-based encryption with verifiableoutsourced decryption,” IEEE Transactions on Information Forensics and Security,8(8), 2013, pp. 1343–1354.

[7] S. Lin, R. Zhang, H. Ma, M. Wang, “Revisiting attribute-based encryption withverifiable outsourced decryption,” IEEE Transactions on Information Forensics andSecurity, 10(10), 2015, pp. 2119–2130.

[8] S. Soursos, I. P. Zarko, P. Zwickl, I. Gojmerac, G. Bianchi, G. ˇ Carrozzo,“Towards the cross-domain interoperability of IoT platforms,” In EuropeanConference on Networks and Communications (EuCNC), 2016, pp. 398–402.

[9] C. Chen, Z. Zhang, D. Feng, “Efficient ciphertext policy attributebased encryptionwith constant-size ciphertext and constant computation-cost,” In Provable Security,2011, pp. 84–101.

[10] C. Hahn, H. Kwon, J. Hur, “Efficient attribute-based secure data sharing withhidden policies and traceability in mobile health networks,” Mobile InformationSystems, 2016.

[11] Z. Zhou, D. Huang, Z. Wang, “Efficient privacy-preserving ciphertext-policyattribute based-encryption and broadcast encryption,” IEEE Transactions onComputers, 64(1), 2015, pp. 126–138.

[12] Z. Zhou, D. Huang, “On efficient ciphertext-policy attribute based encryptionand broadcast encryption,” In Proceedings of the 17th ACM conference on Computerand communications security, 2010, pp. 753–755.

[13] Shang, W., Bannis, A., Liang, T., Wang, Z., Yu, Y., Afanasyev, A., and Zhang,L., “Named data networking of things,” In IEEE First International Conference onInternet-of-Things Design and Implementation (IoTDI), 2016, pp. 117–128.

[14] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M., “Internet of Things(IoT): A vision, architectural elements, and future directions,” Future generationcomputer systems, 29(7), 2013, pp. 1645–1660.

[15] Ghose, A., Biswas, P., Bhaumik, C., Sharma, M., Pal, A., and Jha, A., “Roadcondition monitoring and alert application: Using invehicle smartphone as internet -connected sensor,” In IEEE International Conference on Pervasive Computing andCommunications Workshops (PERCOM Workshops), 2012, pp. 489–491.

[16] Li, J., Sha, F., Zhang, Y., Huang, X., and Shen, J., “Verifiable outsourceddecryption of attribute-based encryption with constant ciphertext length,” Securityand Communication Networks, 2017.

[17] Y. Jiang, W. Susilo, Y. Mu, F. Guo, “Ciphertext-policy attributebased encryptionagainst key-delegation abuse in fog computing,” Future Generation ComputerSystems,78, 2017, pp. 720–729.

[18] Y. B. Saied, A. Olivereau, D. Zeghlache, M. Laurent, “Lightweight collaborativekey establishment scheme for the Internet of Things,” Computer Networks, 64, 2014,pp. 273–295.

[19] M. J. Hinek, S. Jiang, R. Safavi-Naini, S. F. Shahandashti, “Attribute-basedencryption with key cloning protection,” International Journal of AppliedCryptography, 2(3), 2012, pp. 250–270.

[20] S. Yu, K. Ren, W. Lou, J. Li, “Defending against key abuse attacks in KP –ABEenabled broadcast systems,” In International Conference on Security and Privacy inCommunication Systems, 2009, pp. 311–329.

[21] J. Li, Q. Huang, X. Chen, S. S. Chow, D. S. Wong, D. Xie, “Multi-authorityciphertext-policy attribute-based encryption with accountability,” In Proceedings ofthe 6th ACM Symposium on Information, Computer and Communications Security,2011, pp. 386–390.

[22] Z. Liu, Z. Cao, D. S. Wong, “Blackbox traceable CP-ABE: how to catch peopleleaking their keys by selling decryption devices on ebay,” In Proceedings of the ACMSIGSAC conference on Computer & communications security, 2013, pp. 475–486.

[23] J. Ning, Z. Cao, X. Dong, L. Wei, X. Lin, “Large universe ciphertext –policyattribute-based encryption with white-box traceability,” In European Symposium onResearch in Computer Security, 2014, pp. 55–72.

[24] G. Yu, Z. Cao, G. Zeng, W. Han, “Accountable ciphertext-policy attribute-basedencryption scheme supporting public verifiability and nonrepudiation,” InInternational Conference on Provable Security, 2016, pp. 3–18.

[25] J. Ning, Z. Cao, X. Dong, J. Gong, J. Chen, “Traceable CP -ABE with shortciphertexts: how to catch people selling decryption devices on eBay efficiently,” InEuropean Symposium on Research in Computer Security, 2016, pp. 551–569.