information

MDPI

Article

RSA-CP-IDABE: A Secure Framework for Multi-User and Multi-Owner Cloud Environment

Sonali Chandel [1,*], Geng Yang [1,2] and Sumit Chakravarty [3]

[1] School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; yangg@njupt.edu.cn
[2] Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing 210003, China
[3] Department of Electrical and Computer Engineering, Kennesaw State University, Marietta, GA 30060, USA; schakra2@kennesaw.edu
* Correspondence: f2015010102@njupt.edu.cn

Received: 23 June 2020; Accepted: 20 July 2020; Published: 29 July 2020

Abstract: Cloud has become one of the most widely used technologies to store data due to its availability, flexibility, and low cost. At the same time, the security, integrity, and privacy of data that needs to be stored on the cloud is the primary threat for cloud deployment. However, the increase in cloud utilization often results in the creation of a multi-user cloud environment, which requires its owners to manage and monitor the data more effectively. The security of information faces an additional threat, which is related to the increasing number of users and owners who deal with the data stored on the cloud. Many researchers have developed several frameworks and algorithms to address the security issues of the cloud environment. In the present work, a novel algorithm is proposed with the integration of Ciphertext Policy-Identity Attribute-based Encryption (CP-IDABE) and the Rivest–Shamir–Adelman (RSA) algorithm for securing the cloud. Both the owners and users are provided with the public and distinct secret keys that are generated by the Automated Certificate Authority (ACA). The attribute policy differentiates between the user and owner for accessing the cloud data. The proposed RSA-CP-IDABE algorithm also prevents the Man in the Middle (MITM) attack effectively. The performance of the proposed algorithm is evaluated for its time used for encryption, decryption, and execution for varying sizes of data. The obtained results are compared with the existing framework to show its effectiveness. The proposed algorithm can be enhanced with the revocation of privileges in the future.

Keywords: cloud data; ciphertext; CP-IDABE; multi-owner; MITM attack; privacy; RSA; security

1. Introduction

Cloud computing has revolutionized data management in various organizations, globally. The related industries have realized the usage of cloud as a shared environment scheme, which helps to improve the efficiency of data storage [1]. The Cloud storage service provides a comparably low-cost, scalable, and position-independent platform for clients. As a result, it becomes a rapidly profit-earning growth service. It can also integrate multiple internal and/or external cloud services mutually to give high interoperability with open architectures and interfaces [2]. However, there are numerous security issues and challenges in cloud computing because it encompasses many technologies, such as networks, databases, operating systems, virtualization, resource scheduling, transaction management, concurrent control, and memory management [3,4]. In a cloud environment, the responsibility for employing and preserving efficient security mechanisms are in the hands of the providers. To reduce the panic of customers regarding the cloud, these providers try to assure the customers by claiming that the user data and applications stored in their space will be accurately secured [5].

Security is considered one of the most significant obstacles for cloud computing [6], making it a massive challenge for the organizations that provide various cloud services to the users. The security of data in a cloud is mainly about preserving its integrity, confidentiality, and privacy. As a cloud accumulates more and more data, the threats and risks from hackers and cybercriminals also increase proportionally. The hackers can break into all kinds of public, private, or hybrid cloud computing environments [7]. Several security schemes have been proposed in the past for efficient and secure data sharing on untrusted servers. In all of these approaches, the encrypted data files are stored on untrusted storage space, and the data owners [8] distribute the corresponding decryption keys only to the authorized users. The most common form of attack that is observed in the cloud environment is Man in the Middle (MITM) attack. In MITM attacks, the attacker attempts to intercept the messages that are generated during the exchange of a public key and echo them. Then they substitute those keys with a new key of their own to deceit the steps involved in the processing of the user's request [9,10]. During the attack process, it still appears as if the two parties that are the user and cloud are continuously communicating with each other. The message sender does not get any clue that the receiver is a hacker trying to access or modify the message before retransmitting it to the receiver.

This paper is motivated by the work of Chase [11] and Anand et al. [12]. Chase [11] had reflected on the concept and notion of using the Attribute-Based Encryption (ABE) scheme for both users and owners in the cloud environment. Anand et al. [12] have successfully implemented and integrated the Enhanced Elliptical Curve Cryptography (ECC), along with the Diffie-Hellman algorithm (EECDH), to secure the data in a multi-owner cloud environment. But it does not consider the user's side. In the present work, the issues of multi-owner and multi-user sharing are addressed in the cloud environment with the help of the authentication and signature-verification mechanism, so that the data sharing among owners and users can be more secure and private. A dual encryption model is proposed for both multi-user and multi-owner cloud environment. The data in the cloud can be modified and accessed only by its owner and the intended user. However, this happens only after authentication of the data through the Cipher Plain text-based Identity Attribute-based Encryption (CP-IDABE). Additionally, the Rivest–Shamir–Adelman (RSA) encryption is also used to secure the data in the cloud. The proposed approach is intended to prevent the Man in the Middle (MITM) attack in the cloud environment as well.

The significant contribution of the proposed RSA-CP-IDABE algorithm for securing the data integrity and privacy in the cloud are as follows:

(a) An Automated Cloud Authority (ACA) is established to issue the certificates and keys for both the user and multi owners only after the registration in the cloud.
(b) Each of the users and secondary owners is provided with the distinct secret keys to access the data on the cloud-based on their attributes.
(c) Since the data is double encrypted, only authorized people can access the data or make any modifications.
(d) With the usage of different secret keys on both the user and the owner's side, the confidentiality and integrity of the data are ensured through the proposed scheme.
(e) Prevents the MITM attack effectively in the cloud environment.

The rest of the paper is structured as follows. Section 2 describes the related works on cloud security under a multi-owner environment. Section 3 discusses the preliminaries of the proposed algorithm. The system architecture of the secure cloud environment is provided in Section 4. Section 5 provides the system algorithms. Section 6 lists the results obtained from the proposed algorithm and compares it with the earlier algorithms. Section 7 concludes the paper and states future works.

2. Related Work

Anand et al. [12] had proposed an ECC-based, Diffie-Hellman, key-exchange protocol, and digital signature to protect the multi-owner cloud environment. The proposed algorithm prevents the MITM

attack and secures the data integrity among the multi-owners. However, it does not consider the user side in particular with varying attributes. Huang et al. [13] proposed a novel scheme for the cloud to secure the sharing of data among the users and established the conditional dissemination for the multiple owners. The Identity-based Broadcast Encryption (IBBE) technique is employed to share the data among the users that are obtained from its owners. Additionally, the owners, based on the preferences, provide the fine-grained access policy. The proposed approach is found to provide adequate security to the data in the multi-owner clouds. Miao et al. [14] presented a privacy-preserving scheme that is developed with attribute-based keyword search techniques in the multi-owner cloud environment. The proposed scheme improved the tracing of malicious users. The scheme is useful in providing adequate security and prevents the keyword-guessing attack in offline mode. The performance of the proposed scheme is evaluated on real-world datasets.

Sangeetha et al. [15] addressed the issues that are present in the Personal Health Record (PHR) frameworks due to multi-owners. Their work proposed two different frameworks with a Secure-Key Policy Attribute-based Encryption (S-KP-ABE) in the personal domain and Privacy-Preserving-Decentralized Collusion Resistant- Attribute-based Encryption (PP-DCR-ABE) in the public domain. By employing the tokenization technique, the proposed algorithm provided security over the collusion attacks. The experimental outcome validated the improved performance of the proposed frameworks in a multi-owner-based PHR cloud environment.

Miao et al. [16] proposed a novel conjunctive keyword search framework for securing the data in the multi-owner cloud environment. The multi-signature approach was employed to secure the cloud over the keyword guessing attack in the proposed model. The proposed model is verified against the real-time data and is found to be effective. Rong et al. [17] proposed a K-means Clustering-based Privacy-preserving Scheme for securing the distributed cloud in the multi-owner setting. The proposed work provides the set of building blocks of privacy-preserving and employs the protocol of outsourced K-means clustering. From the theoretical analysis, it was observed that the proposed scheme could provide the confidentiality of the cloud data with reduced computational overhead. The proposed scheme experimentally validates its performance against the existing approaches.

Aruna et al. [18] discussed the security and protection that are required for a multi-owner cloud environment. The work examined the different techniques in securing the multi-owner cloud and suggested that the research on the multi-owner cloud should be extended to provide enhanced security, storage, and processing of information in the cloud. Guo et al. [19] proposed an accurate, secure, and efficient multi-owner cloud environment through a scheme of multi-keyword ranked search techniques. A new weight formulation scheme was developed for the keywords for the quality-based ranking of the document. The greedy depth-first search algorithm was employed to improve the constructed global balanced binary tree index.

Peng et al. [20] proposed another tree-based ranking scheme for multi-keyword searches to secure the cloud in a multi-owner environment. Additionally, a privacy-preserving protocol was proposed to enhance security through the process of bilinear mapping. From the security analysis, it is observed that the proposed scheme can secure the cloud, and the security is validated from its performance analysis. Li et al. [21] proposed multi-owner, key-aggregate, searchable encryption through a trapdoor technique to share the data in a multi-owner cloud environment. The scheme supports effective data sharing for both multiple owners and users by reducing unnecessary trapdoors that are hard for generating by mobile devices during the querying step. The security and performance analysis showed the effectiveness of the proposed scheme.

3. Preliminaries

For securing the cloud in the multi-owner and multi-user environment, a novel Ciphertext Policy Attribute-based Encryption (CP-IDABE) with the identity of a user is integrated with the RSA encryption techniques. The preliminaries for the proposed method are given in this section, and their corresponding notations are given in Table 1.

Table 1. Notations used in the CP-IDABE cryptic mechanism.

S.no	Notation	Explanation
1	D_O	Digital signature of the owner
2	D_U	Digital signature of the user
3	VM_O	Verified message for the owner
4	VM_U	Verified message for the user
5	ID_i	Identity of the owner
6	ID_j	Identity of the user
7	A_O	Attribute of the owner
8	A_U	Attribute of the user
9	PUK	Public key for CP-IDABE
10	MAK	Master key for CP-IDABE
11	PUK_R	Public key for RSA
12	MAK_R	Master key for RSA
13	OSK	Owner's secret key
14	USK	User's secret key
15	Et	Encryption text after CP-IDABE
16	A	Attribute policy set
17	Et′	Encrypted text after RSA
18	M	Message for CP-IDABE
19	M′	Message for RSA
20	K_R	The secret key for RSA

3.1. CP-IDABE

The CP-IDABE combines both the attributes and the ID of the user/owner for the cryptic mechanism over the cipher data under their respective access policy. The users utilize the unsymmetrical security key to access the data in the cloud environment. Additionally, in the proposed scheme, the secondary owners of the data were provided with a distinct secret key. The user/owner generates their username along with a password that serves to be the identity for the proposed scheme. The attribute set that encloses the privileges of access defines whether the person is either a user or an owner. Furthermore, a set of answers to a set of questions [22] is used to validate the user and owner when accessing the cloud through the ACA.

- Attributes: In the proposed model, the attributes of the user/owner can be anything coming from the set of five random questions provided randomly by ACA. The five random questions provided in the present schemes are (i) primary job (ii) last three digits of credit card (iii) native place (iv) favorite sports (v) favorite team.
- Policy: The access policy is very significant in the proposed model, as it is established over the multi-owner and multi-user cloud environment through ACA. In addition to the authorized access to the data, the access policy also provides privileges like editing and removing the data for multiple owners. However, users must be restricted only to access the data, and they do not have the privileges to edit it. In general, the access policy for the owners are defined as $(A_O \bigwedge ID_i \bigwedge O_i)$, and the users are defined as $(A_U \bigwedge ID_j \bigwedge U_j)$. When the above conditions are satisfied, the access will be approved, or else it will be denied.

The CP-IDABE consists of the following steps in performing the cryptic mechanism to the cloud data:

1. Setup (1^P): The public key, PUK, and the master key, MAK, are generated for the user and the owner based on security parameter P through ACA. Similarly, the public key and the master key are generated for the RSA algorithm as PUK_R and MAK_R, respectively.

Multi-owner:

2. KeyGen (MAK, A_O, ID_i): given the owner attributes A_O and MAK, with the identity (ID_i), this algorithm yields the private key of owner OSK_i.

3. Enc (PUK, M, ID_i, A): given PUK, user identity ID_i with access policy set A, the ciphertext Et is generated with the message M.
4. Dec (OSK_i, Et): given the secret key OSK_i, a ciphertext Et is decrypted through the owner attributes A_O and user identity ID_i to get the message M.

Multi-user:

- KeyGen (MAK, A_U, ID_j): given the owner attributes A_U and MAK, with the identity (ID_j), this algorithm yields the private key of owner USK_i.
- Enc (PUK, M, ID_j, A): given PUK, user identity ID_j with access policy set A, the ciphertext Et is generated with the message M.
- Dec (USK_i, Et): given the secret key USK_i, a ciphertext Et is decrypted through the A_U and ID_j to get the message M.

3.2. RSA-Cryptology

To improve the security and to have effective access control among multi-owners and multi-users, the cloud data is encrypted with the CP-IDABE is encrypted one more time with the RSA algorithm. The RSA algorithm generally takes two prime numbers L and M, randomly to generate the secret key [23,24].

- Enc (M', PUK_R): given key PUK_R, a message M' yield final ciphertext Et'.
- Dec (K_{Ri}, Et'): given key K_{Ri}, this algorithm yields the message M' from Et'.

After selecting two prime numbers L and M, the following steps are followed for the key generation:

Step 1: estimate N = LxM
Step 2: estimate $\varphi(N) = (L-1)(M-1)$
Step 3: choose integer e
Step 4: GCD $(\varphi(N), e) = 1; 1 < e < \varphi(N)$
Step 5: calculate d

$d^e \bmod \varphi(N) = 1$
public key PUK_R = {e, n}
private key K_R = {d, n}.

3.3. Digital Signature

- SignGen (ID_i, ID_j, A): with the user/owner identity along with their access policy A, yields digital signature D_U and D_O for the user and owner, respectively, with the verifying message VM.

4. System Model

4.1. Description of the System Model

The proposed RSA-CP-IDABE for securing the multi-owner and multi-user cloud environment is given in Figure 1. The proposed framework has four essential components which are discussed below:

- Automated Cloud Authority (ACA): This component is employed to register the users and the owners of the data. Initially, the primary owner of the data registers and obtains the secret key for uploading and accessing the cloud data. The principal owner approves other owners through ACA only. The owners can approve any users through the ACA. The ACA access the attribute set of both the user and owner and generate the keys that are used by them to access the data. The ACA controls access over the data through verification of the keys for both users and owners distinctly.

- Cloud: It is a vital component in the proposed framework that stores the encrypted data. The encryption over the data is initially performed with the CP-IDABE using the attribute policy set. Then, the RSA algorithm is applied over encrypted data and stored in the cloud. The user processes the request for data from the cloud and receives it for accessing it.
- Multi-owner: It is the group of people who possess the privilege to access the data and modify or update it regularly. In the proposed framework, the data has one primary owner who monitors and controls other owners through ACA. The multi-owners provide access to multiple users and track their access over the data. The primary owner can revoke the secondary owner at any stage. Similarly, the owner can revoke the user over suspicious activity through ACA.
- Multi-user: The user in the cloud environment accesses the data through a secret key. The user is authorized by anyone of the multi-owners of the data in the cloud. The user has the privilege to access the data, but they are not allowed to modify or update the cloud data. The data owner through ACA can revoke the user at the time because of any suspicious activities.

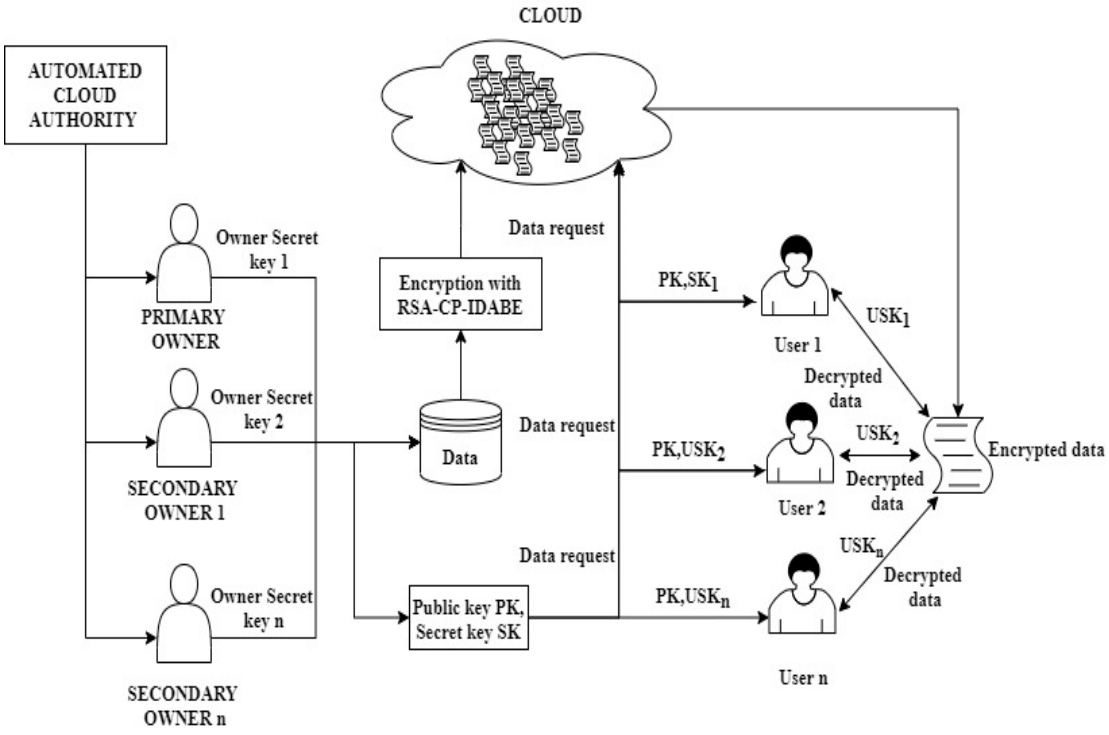

Figure 1. A proposed security framework for the cloud environment.

4.2. Security Analysis

Once the primary owner uploads the data into the cloud, additional secondary owners can be added through the approval of the primary owner. Both the primary and the secondary owners are provided with the public key and the secret key from the ACA to access the data and edit it. Let us consider the owners, as in Figure 2: The primary owner, i.e., owner 1, uploads the encrypted data into the cloud. The secondary owner can access the data on providing the secret key that encloses their identity and the attributes of the owner. The primary owner has to offer both the RSA key and the CP-IDABE key to access the data to the user and secondary owners.

Similarly, when the user requests for the data, the digital signature of the user is verified, and access to the data is granted to them. The user uses the secret key to access the obtained encrypted cloud data. If the data owners revoke the user, the users cannot get the data stored in the cloud, as shown in Figure 2.

The MITM attack often occurs when the attacker tries to establish a clear connection among the owners or users. The messages are relayed between the two owners. As seen in Figure 2, consider a

scenario where a connection is established between the owner-2 and the user-2 through the MITM attack. When the attacker attempts to access the cloud data from the owner's or the user's end, they must know the answer to the random security questions. They also must provide the secret key to access the data. Hence, the attacker may not breach the security framework, and the data is secured from a MITM attack.

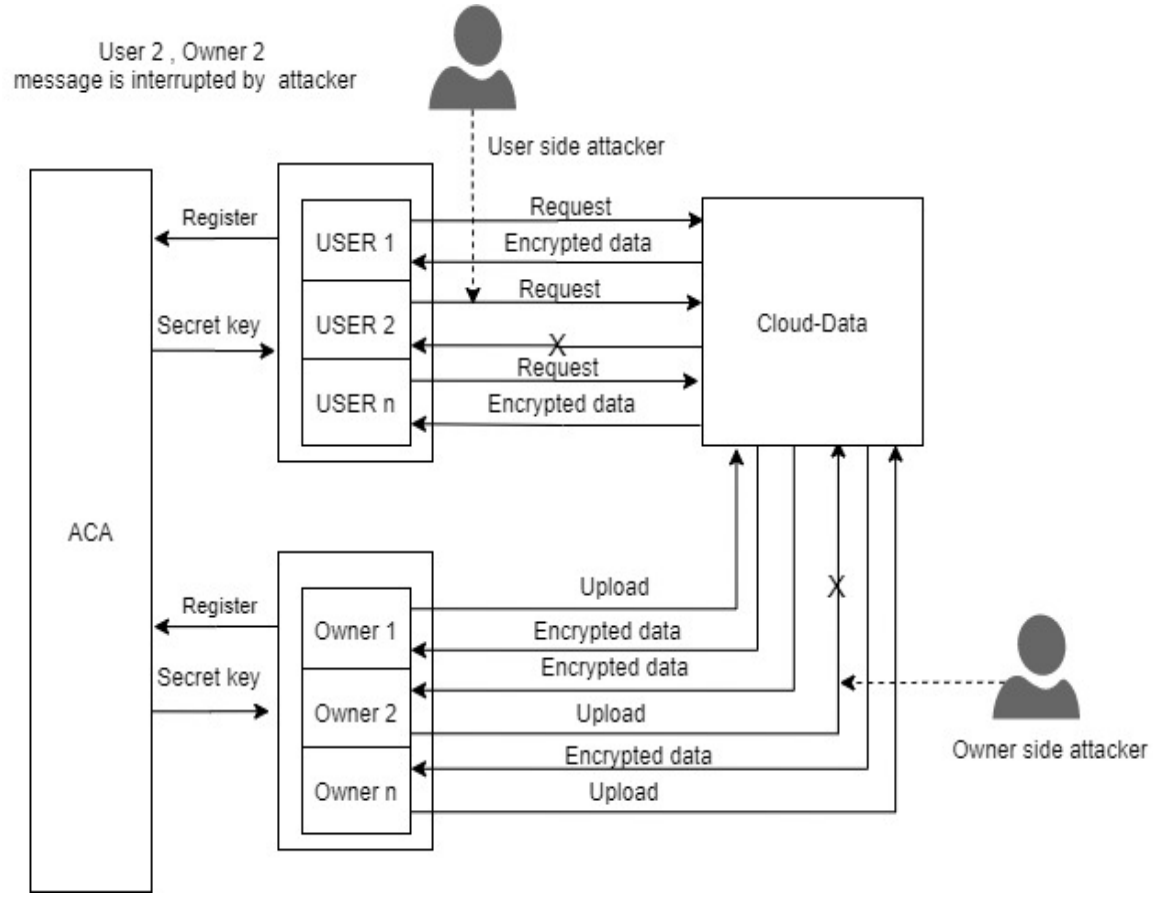

Figure 2. The security model of proposed RSA-CP-IDABE.

Additionally, the security key provided is robust against any breaches. The central aspect of the proposed framework is that the RSA algorithm uses the unsymmetrical key cryptography, and as a result, the attacker has to know both the decryption keys individually to access the data. Even when the attacker breaches the RSA key, they need to obtain the CP-IDABE, which was formulated with access policy that contains the user/owner attribute along with its random identity. Therefore, the proposed model is more secure than the EECDH [12], which is the symmetrical key encryption model. Due to the complex security-key model in the proposed scheme, the data in the cloud can be secured adequately against many attacks.

5. Construction of the Algorithm

The proposed RSA-CP-IDABE algorithm consists of three different algorithms to ensure the security of the data in the cloud. The first algorithm is the digital signature algorithm that generates the digital signature for both the user and the owner at the time of registration in the cloud. The second algorithm is the CP-IDABE that encrypts the data initially. The final algorithm is the RSA algorithm that encrypts the previously encrypted information once again before storing it on the cloud. Both the CP-IDABE and RSA are used together for the key generation process in the proposed framework.

5.1. Digital Signature

5.1.1. For Owners

- DO, VMO ← SignGen ($ID_{i,}$ A_O): the ACA gets the identity of the owner ID_i along with the attributes of the owner A_O to generate the digital signature DO with the verifying message VMO.
- Verify ← DO, VMO: the owner can be verified with the generated digital signature DO and verifying message VMO.

5.1.2. For Users

- DU, VMU ← SignGen (ID_j, A_U): the ACA gets the identity of the user ID_j along with the attributes of the user A_U to generate the digital signature DU with the verifying message VMU.
- Verify ← DU, VMU: the user can be verified with the generated digital signature DU and verifying message VMU.

5.2. Key Generation with CP-IDABE & RSA

PUK, MAK, PUK_R, MAK_R ← Setup (1^P): The ACA with the security parameter P generates the public key, PUK, and the master key, MAK, for the cloud owner and user for CP-IDABE. Similarly, for RSA, the public and master keys are PUK_R and MAK_R.

5.2.1. For Owners

- OSK_i ← KeyGen (MAK, A_O ID_i): this algorithm uses the master key, MAK, generated by the ACA, along with the identity ID_i and attributes of the owner, A_O, respectively, to generate the secret key for the owner OSK_i.
- Et ← Enc (PUK, M, ID_i, A): based on the attribute policy set A, the identity of the owner ID_i and the public key, PUK, the data is encrypted with the message M to obtain the ciphertext, Et.
- Et′ ← Enc (Et, M′, PUK_R): when the ciphertext Et is obtained, by applying the public key, PUK_R, with the message M′, the ciphertext is again encrypted as Et′.
- M′ ← Dec (K_{Ri}, Et′): using the secret key K_{Ri}, the encrypted data Et′ will yield the decrypted message M′.
- M ← Dec (OSK_i, Et): using the secret key OSK_i, a ciphertext Et is decrypted through the message M.

5.2.2. For Users

- USK_j ← KeyGen (MAK, A_U, $ID_{j)}$: this algorithm uses the master key, MAK, generated by the ACA along with the identity and attributes of the user ID_j and A_U, respectively, to generate the secret key for the owner USK_j.
- Et ← Enc (PUK, M, ID_j, A): based on the attribute policy set A, the identity of the user, ID_j, and the public key, PUK, the data is encrypted with the message M to obtain the ciphertext, Et.
- Et′ ← Enc (Et, M′, PUK_R): when the ciphertext Et is attained, by applying the public key, PUK_R, with the message M′, the ciphertext is again encrypted as Et′.
- M′ ← Dec (K_{Rj}, Et′): using the secret key K_{Rj}, the encrypted data Et′ will yield the decrypted message M′.
- M ← Dec (USK_j, Et): using the secret key USK_j, a ciphertext Et is decrypted through the message M.

6. Results & Discussion

The proposed RSA-CP-IDABE framework to secure the cloud data in the multi-owner and multi-user environment is implemented through Java. The private cloud is established through the Eucalyptus that runs on the i5 Intel core processer with 2.50 GHz using the 16 GB RAM. The performance of the proposed framework is analyzed for its performance through the time taken for encryption

and decryption and the overall execution time. The obtained results are compared with the EECDH model [12], since it was implemented to secure only the multi-owner cloud, and the proposed RSA-CP-IDABE is developed to secure the cloud with both multi-owner and multi-user.

6.1. Encryption Time

The encryption time is the time taken for encrypting the data through the cryptic mechanism in the cloud security framework. The encryption time generally depends on the size of data that is to be encrypted. In the proposed RSA-CP-IDABE scheme, the encryption time for 8-KB data is 40 ms, and for 1024-KB data, it is 124 ms. However, for the existing Enhanced Elliptical Curve Diffie Hellman (EECDH) algorithm [12], the time taken for encrypting 8-KB data is 51 ms, and 1024-KB is 136 ms, respectively. The comparison between the proposed RSA-CP-IDABE and existing EECDH [12] is given in Table 2 and Figure 3.

Table 2. Encryption time vs. file size for RSA-CP-IDABE & EECDH.

File Size (KB)	EECDH [12]	RSA-CP-IDABE
8	51	40
16	52	44
32	55	52
64	66	62
128	78	74
256	110	86
512	122	100
1024	136	124

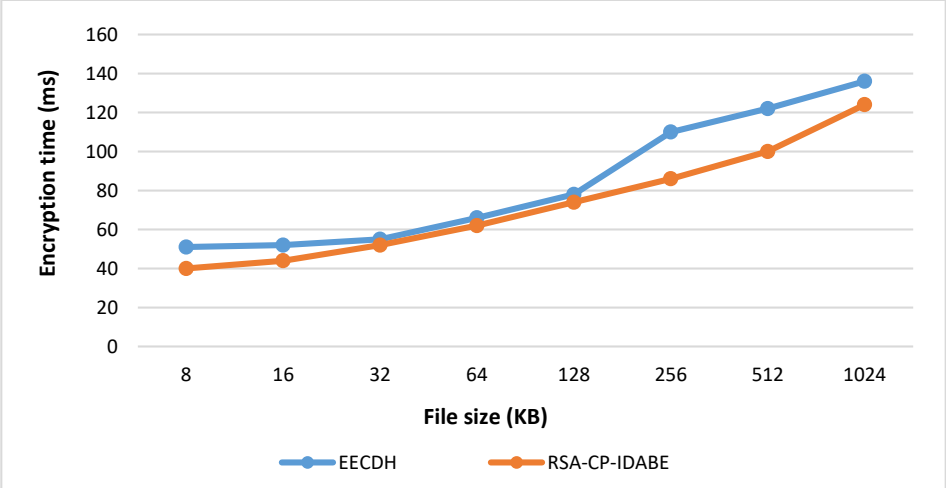

Figure 3. Comparison between the proposed RSA-CP-IDABE and EECDH for encryption time.

6.2. Decryption Time

The decryption time is the time taken by the owner or the user to decrypt the data using the proposed algorithm. For the proposed RSA-CP-IDABE, the decryption time for 8-KB and 1024-KB data is about 45 ms and 71 ms, respectively. The comparison between the proposed RSA-CP-IDABE and the existing EECDH over decoding different file sizes is given in Table 3 and Figure 4. The existing EECDH [12] takes 54 ms and 80 ms for decrypting the 8-KB and 1024-KB data, respectively. The comparison between the proposed RSA-CP-IDABE and the existing EECDH over decoding different file sizes are given in Table 3 and Figure 4.

Table 3. Decryption time vs. file size for RSA-CP-IDABE & EECDH.

File Size (KB)	EECDH [12]	RSA-CP-IDABE
8	54	45
16	59	48
32	62	56
64	67	61
128	70	65
256	73	68
512	78	71
1024	80	71

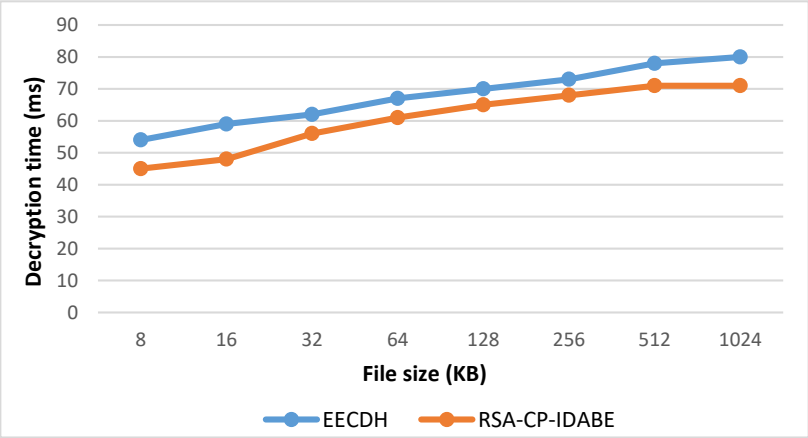

Figure 4. Comparison between the proposed RSA-CP-IDABE and EECDH for decryption time.

6.3. Execution Time

The execution time is the total time taken to secure the data in the cloud environment. The execution time includes the key generation time, encryption time, uploading time, downloading time, decryption time, and verification time. The execution time over the 8-KB and 1024-KB data is about 725 ms and 17,523 ms, respectively, for the existing EECDH [12]. In comparison, the proposed RSA-CP-IDABE has an execution time of 675 ms and 15,792 ms for 8-KB and 1024-KB data, respectively. Table 4 and Figure 5 shows the comparison of performances between the RSA-CP-IDABE and EECDH [12].

Table 4. Execution time vs. file size for RSA-CP-IDABE & EECDH.

File Size (KB)	EECDH [12]	RSA-CP-IDABE
8	725	675
16	1065	930
32	1185	985
64	3857	3000
128	4652	3724
256	7474	6592
512	9863	8520
1024	17,523	15,792

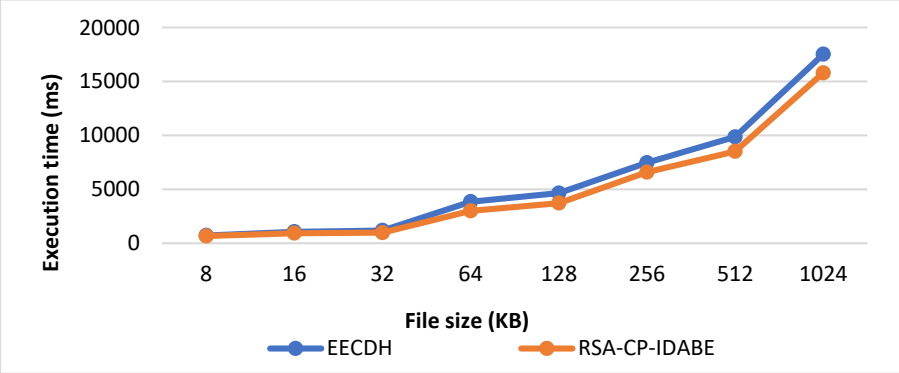

Figure 5. Comparison between the proposed RSA-CP-IDABE and EECDH for execution time.

It is observed in Table 5 that both the encryption and decryption time of the proposed approach is higher than the Improved CP-ABE (I-CP-ABE) [25] when the number of attributes is less. However, when there is an increase in the attributes, the difference between the times decreases, and it was observed that the proposed RSA-CP-IDABE is better than the existing I-CP-ABE, as shown in Figures 6 and 7.

Table 5. Encryption, decryption time vs. the number of attributes.

Process	No. of Attributes	10	20	30	40	50
Encryption	I-CP-ABE [25]	150	250	400	500	600
	RSA-CP-IDABE	175	240	365	445	510
Decryption	I-CP-ABE [25]	80	90	110	150	190
	RSA-CP-IDABE	92	101	111	146	175

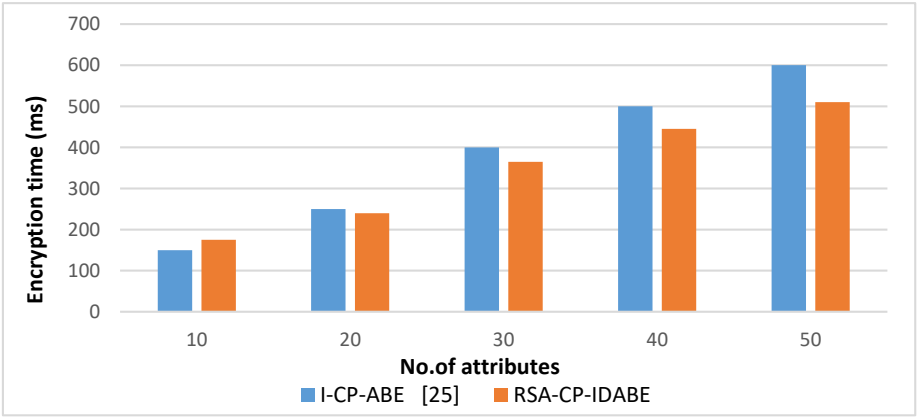

Figure 6. Comparison between the proposed RSA-CP-IDABE and I-CP-ABE for encryption time.

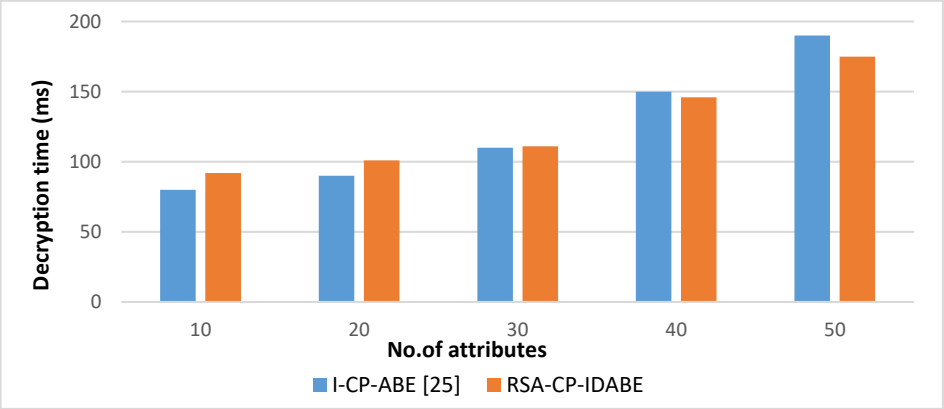

Figure 7. Comparison between the proposed RSA-CP-IDABE and I-CP-ABE for decryption time.

7. Conclusions and Future Work

For securing the data in the multi-user and multi-owner cloud environment, a novel RSA-CP-IDABE algorithm was proposed. Both the user and the owner have to register through the ACA in the cloud. After the registration, they are provided with a public key and a distinct secret key based on the attributes. The RSA-based secret key is provided to both the owner and the user. The primary owner monitors the activities of secondary owners over the data. The multiple owners monitor user activities. When the owners upload the data, both the CP-IDABE and RSA algorithm are executed by the system to encrypt the data. The user accesses the data using the dual decryption keys. The proposed algorithm also prevents the MITM attack effectively through the double encryption over the cloud data.

The proposed RSA-CP-IDABE is evaluated for its performance over the varying sizes of data. The encryption time for 1024-KB data is about 124 ms, and its decryption time is 71 ms. The total execution time for 1024-KB data is about 15,792 ms. From the comparison, it is observed that the proposed RSA-CP-IDABE is more useful and effective in securing data in the cloud than the existing EECDH and I-CP-ABE algorithm.

The drawback of the proposed security scheme is that it performed better over the existing I-CP-ABE model only when the number of attributes increases. The concept of revocation is vital for cloud users, as it establishes control over user activities. In the proposed model, revocation is not considered. The future scope may include the revocation of users and secondary owners using their attributes to ensure the improved integrity and privacy of data.

Author Contributions: Conceptualization, S.C. (Sonali Chandel); data curation, S.C. (Sumit Chakravarty); formal analysis, S.C. (Sumit Chakravarty); funding acquisition, G.Y.; project administration, G.Y.; software, S.C. (Sumit Chakravarty); supervision, G.Y.; validation, G.Y.; writing—original draft, S.C. (Sonali Chandel); writing—review and editing, S.C. (Sonali Chandel). All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 61972209.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shajina, A.R.; Varalakshmi, P. A novel dual authentication protocol (DAP) for multi-owners in cloud computing. *Clust. Comput.* **2017**, *20*, 507–523. [CrossRef]
2. Zhu, Y.; Hu, H.; Ahn, G.-J.; Yu, M. Cooperative provable data possession for integrity verification in multi-cloud storage. *IEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 2231–2242. [CrossRef]
3. Doelitzscher, F.; Sulistio, A.; Reich, C.; Kuijs, H.; Wolf, D. Private cloud for collaboration and e-Learning services: From IaaS to SaaS. *Computing* **2011**, *91*, 23–42. [CrossRef]

4. Information Resources Management Association. *Standards and Standardization: Concepts, Methodologies, Tools and Applications*; IGI Global: Hershey, PA, USA, 2015.
5. Bernsmed, K.; Jaatun, M.G.; Meland, P.H.; Undheim, A. Thunder in the Clouds: Security challenges and solutions for federated Clouds. In Proceedings of the 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, Taipei, Taiwan, 3–6 December 2012.
6. Rimal, B.P.; Choi, E.; Lumb, I. A Taxonomy and Survey of Cloud Computing Systems. In Proceedings of the Fifth International Joint Conference on INC, IMS and IDC, Seoul, Korea, 25–27 August 2009.
7. Ahmat, K. Emerging Cloud Computing Security Threats. *arXiv* **2015**, arXiv:1512.01701.
8. Liu, X.; Zhang, Y.; Wang, B.; Yan, J. Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *24*, 1182–1191. [CrossRef]
9. Chowdary, P.R.; Challa, Y.; Jitendra, M.S.N.V. Identification of MITM Attack by Utilizing Artificial Intelligence Mechanism in Cloud Environments. *J. Physics Conf. Ser.* **2019**, *1228*, 012044. [CrossRef]
10. Amara, N.; Zhiqui, H.; Ali, A. Cloud Computing Security Threats and Attacks with their Mitigation Techniques. In Proceedings of the 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Nanjing, China, 12–14 October 2017; pp. 244–251.
11. Chase, M. Multi-Authority Attribute Based Encryption. In *Theory of Cryptography Conference*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 515–534.
12. Anand, S.; Perumal, V. EECDH to prevent MITM attack in cloud computing. *Digit. Commun. Netw.* **2019**, *5*, 276–287. [CrossRef]
13. Huang, Q.; Yang, Y.; Yue, W.; He, Y. Secure Data Group Sharing and Conditional Dissemination with Multi-Owner in Cloud Computing. *IEEE Trans. Cloud Comput.* **2019**, 1. [CrossRef]
14. Miao, Y.; Liu, X.; Choo, K.-K.R.; Deng, R.H.; Li, J.; Li, H.; Ma, J. Privacy-Preserving Attribute-Based Keyword Search in Shared Multi-owner Setting. *IEEE Trans. Dependable Secur. Comput.* **2019**, 1. [CrossRef]
15. Sangeetha, D.; Vaidehi, V. A secure cloud based Personal Health Record framework for a multi owner environment. *Ann. Telecommun.* **2016**, *72*, 95–104. [CrossRef]
16. Miao, Y.; Ma, J.; Liu, X.; Jiang, Q.; Zhang, J.; Shen, L.; Liu, Z. VCKSM: Verifiable conjunctive keyword search over mobile e-health cloud in shared multi-owner settings. *Pervasive Mob. Comput.* **2017**, *40*, 205–219. [CrossRef]
17. Rong, H.; Wang, H.; Liu, J.; Hao, J.; Xian, M. Privacy-Preserving-Means Clustering under Multi owner Setting in Distributed Cloud Environments. *Secur. Commun. Netw.* **2017**, *99*, 1–19. [CrossRef]
18. Aruna, K.B.; LallithaShri, A.; Aravindh; Jayasurya; Jayakumar. Protection for Multi Owner Data Sharing Scheme. *Bonfring Int. J. Adv. Image Process.* **2017**, *7*, 1–5. [CrossRef]
19. Guo, Z.; Zhang, H.; Sun, C.; Wen, Q.; Li, W. Secure multi-keyword ranked search over encrypted cloud data for multiple data owners. *J. Syst. Softw.* **2018**, *137*, 380–395. [CrossRef]
20. Peng, T.; Lin, Y.; Yao, X.; Zhang, W. An Efficient Ranked Multi-Keyword Search for Multiple Data Owners Over Encrypted Cloud Data. *IEEE Access* **2018**, *6*, 21924–21933. [CrossRef]
21. Li, T.; Liu, Z.; Jia, C.; Fu, Z.; Li, J. Key-aggregate searchable encryption under multi-owner setting for group data sharing in the cloud. *Int. J. Web Grid Serv.* **2018**, *14*, 21–43. [CrossRef]
22. Chandel, S.; Yang, G.; Chakravarty, S. AES–CP–IDABE: A Privacy Protection Framework against a DoS Attack in the Cloud Environment with the Access Control Mechanism. *Information* **2020**, *11*, 372. [CrossRef]
23. Kalpana, P.; Singaraju, S. Data security in cloud computing using RSA algorithm. *IJRCCT* **2012**, *1*, 2278–5841.
24. Padmaja, N.; Koduru, P. Providing data security in cloud computing using public key cryptography. *Int. J. Eng. Sci. Res.* **2013**, *4*, 1059–1063.
25. Xue, S.; Ren, C. Security Protection of System Sharing Data with Improved CP-ABE Encryption Algorithm under Cloud Computing Environment. *Autom. Control Comput. Sci.* **2019**, *53*, 342–350.