Jun 7th, 12:00 AM

Insider Threat Mitigation in Attribute-based Encryption

Runhua Xu
University of Pittsburgh, runhua.edu@pitt.edu

James B.D. Joshi
University of Pittsburgh, jjoshi@pitt.edu

Prashant Krishnamurthy
University of Pittsburgh, prashant@sis.pitt.edu

David Tipper
University of Pittsburgh, tipper@tele.pitt.edu

Insider Threat Mitigation in Attribute based Encryption

Runhua Xu, James B.D. Joshi, Prashant Krishnamurthy, David Tipper
School of Information Science, University of Pittsburgh
135 North Bellefield Avenue
Pittsburgh, PA 15260
runhua.xu@pitt.edu,jjoshi@pitt.edu,prashant@sis.pitt.edu,tipper@tele.pitt.edu

ABSTRACT

Recent advances in computing have enabled cloud storage service, among others, that collect and provide efficient long term storage of huge amounts data that may include users' privacy sensitive information. Concerns about the security and privacy of the sensitive data stored in the cloud is one key obstacle to the success of these cloud based applications and services. To tackle these issues, Attribute based Encryption (ABE) approaches, especially the Ciphertext-Policy Attribute based Encryption (CP-ABE), have been shown to be very promising. ABE helps provide access control solutions to protect the privacy-sensitive information stored in the cloud storage centers. However, use of an ABE approach in such cases suffers from two key insider threats: insider threat due to colluding users; and that due to a potentially malicious or compromised authority center. Even though the users' collusion has been addressed in the literature, to our best knowledge, the authority center as an insider has not been addressed, and existing schemes assume that the authority is fully trusted or *honest-but-curious*. In this paper, we propose a new technical solution to mitigate the threat from the authority as an insider in an ABE system. We present analysis to show that the proposed work is efficient from the perspective of algorithms and can mitigate the insider threat in the authority party effectively.

CCS CONCEPTS

• **Security and privacy** → **Cryptography**; **Public key (asymmetric) techniques**; **Access control**; *Authorization*; *Usability in security and privacy*; Management and querying of encrypted data;

KEYWORDS

Insider threat, authority, data security, attribute based encryption, ciphertext-based attribute based encryption

1 INTRODUCTION

Recent advances in *Cloud Computing* have enabled the applications that can generate or collect huge amounts of users' personal/sensitive data. The cloud storage service is a very promising approach to help aggregate and/or maintain these sensitive data. According to a report from the *Gartner Inc.*, in 2016, more than 50 percent of global 1000 companies will have their customer-sensitive data stored in the public cloud. Security and privacy issues of the sensitive data are the main concerns that stand as obstacles to the success of such applications. Hence, solutions to ensure protection of data stored in the cloud from all potential threats are critically needed.

One key initial approach to ensure the security and privacy of the sensitive data is employing cryptographic mechanisms, such as symmetric encryption algorithms to help guarantee the confidentiality of the stored sensitive data. However, such mechanisms reduce the utility of the data and also introduce new issues like key management. To tackle these issues, Sahai and Waters [11] propose an *Attribute based Encryption* (ABE) that encrypts the data based on a specified access structure over users' attributes and provides fine-grained access control on the encrypted data [6]. Bethencourt et al. propose *Ciphertext-Policy Attribute based Encryption* (CP-ABE) scheme in [2] in order to make the ABE approaches applicable in the data use/sharing scenarios in the cloud computing environment. Here, the data owner encrypts the data based on a specified access policy and only the users who have the attributes that satisfy the access policy can decrypt the ciphertext.

The ABE schemes support both confidentiality feature and access control function for the data stored in the cloud. When implementing ABE schemes within an infrastructure, *Authority* is a critical component that is a fully trusted third party that helps in setting up the ABE related parameter such as the public keys, and in generating the users' private keys based on users' authorized attributes. Even though the security of the ABE schemes has been provided in literature such as [7, 10], there is a lack of specific focus from the perspective of the insider threat to check the security of the ABE schemes.

We believe there are two possible types of insider attack within an ABE scheme: attack through colluding users and attack from the *Authority* when the assumption of its trustworthiness is not valid. In a collusion attack, a group of users collaborate with each other in decrypting a ciphertext that is not authorized to them by utilizing their respective attribute related key components. While a collusion attack has been well studied in the literature, to the best of our knowledge, the protection against *untrusted Authority* has not been studied. An *Authority* can be untrusted because of malicious intent of people managing it, or when it is compromised by another entity. For instance, if the authority is deployed in the cloud environment, the employee of the cloud service provider may access the secret credentials of the authority. Even though the authority is deployed in the private server that is isolated physically, the system administrator may also

NCS'17, Huntsville, Alabama USA

be the potential insiders stealing the secret credentials from the authority.

In this paper, we focus on the issues of insider threat in the ABE ecosystem, especially the insider threat from the *Authority* of ABE with the assumption of its trustworthiness removed. We analyze the potential insider attacks in the ABE systems and propose a new multi-authority CP-ABE (MA-CP-ABE) scheme to help mitigate the threat. Then we propose two approaches that are built on our proposed multi-authority CP-ABE scheme, namely I_N-*tolerance* and I_{N-1}-*tolerance*.

The rest of the paper is organized as follows. In Section 2, we analyze the potential insider threat in the ABE system. Then we present the mitigation approaches in Section 3. We propose a multi-authority CP-ABE scheme in Section 3.2, and then present two mitigation solutions in Section 3.3. The analysis and discussion are presented in Section 4. Finally, we introduce the related work and conclude our work in Section 5 and Section 6, respectively.

2 INSIDER THREAT IN THE ABE

Here, we overview the ABE system, and then present the potential insider threat in the ABE system.

2.1 ABE Roles/Components

In a typical ABE-based application, there are three elements or roles:

- *Data Owner* who employs the ABE scheme to protect the data stored in the cloud.
- *Data User* who downloads the encrypted data from the cloud and uses the ABE scheme to decrypt it for access.
- *Authority* who is responsible for generating the public keys and users' private keys based on users' authorized attributes.

Note that we let *Authority* be a general role to represent the authority component of ABE. There are still specific roles in the authority center, such as system administrator, attribute authenticator, and other employees. If the authority server is deployed in the private cloud environment, the cloud administrator and other cloud provider employees with access are potential entities associated with the authority center. To simplify the discussion, we use the *Authority* to represent all of the roles in the authority center.

2.2 Potential Insider Threats

Based on the elements/roles discussed in Section 2.1 about the ABE system, we can see the following potential insider threats.

- A main insider threat is that of collusion among the users to compromise data security. In essence, malicious users in the ABE system may collaborate with each other to exchange/collect their key components that are related to their attributes to decrypt the ciphertext that is not authorized to them.
- Another key insider threat comes from the *Authority* who may compromise the authorities' secret credentials or inadvertently misuse them. These *Authorities* can be a malicious insider to modify or steal confidential or sensitive information for a personal gain. Also, they can be a careless insider who inadvertently make the authority center vulnerable to compromise.

To the best of our knowledge, the collusion threats have been well-addressed in the literature for CP-ABE schemes; however, the insider threat from the authority has not been addressed in the literature, which is the focus of our paper.

3 INSIDER THREAT MITIGATION

3.1 Preliminaries

Before the introduction to our proposed multi-authority CP-ABE, we first present the preliminaries of the CP-ABE schemes.

Definition 3.1. Linear Secret Sharing Schemes [1]. If a secret-sharing scheme Π, is linear, it should satisfy the following two conditions:

- For each party, the generated shares should be a vector (over $\mathcal{Z}_p$).
- There should be a share-generating matrix, $M_{l\times n}$,for the scheme. For each row in the matrix, let ρ be a function such that $\rho(i)$ maps to the i-th party. We generate $n-1$ random number (over $\mathcal{Z}_p$) and combine with the secret s to get the column vector v. Then $S = Mv$ should be the sharing vector for the secret s and the share S_i is for party $\rho(i)$.

Suppose we have an access structure $\mathcal{AS}$ and Π is the corresponding LSSS. We define the authorized set as $S \in \mathcal{AS}$ and sharing set as $I = \{i : \rho(i) \in S\}$. Let $\{\omega_i \in \mathcal{Z}_p\}_{i\in I}$ be a set of constants that satisfy $\sum_{i\in I} \omega_i\lambda_i = s$, where λ_i are valid shares of the secret s by Π. According to [1], constants ω_i can be generated in polynomial time in the size of M.

3.2 Multi-Authority CP-ABE

3.2.1 CP-ABE Model. The CP-ABE scheme usually consists of four randomized algorithms: *Setup, KeyGen, Encrypt, Decrypt* that are described as follows [2]:

- Setup. The *Setup* algorithm is executed by the *Authority* to setup the parameters such as public keys and master private keys.
- KeyGen. The *KeyGen* algorithm is executed by the *Authority* to generate the users' private keys based on their authorized attributes.
- Encrypt. The *Encrypt* algorithm is executed by the *Data Owner* to encrypt the data based on a specific access policy.
- Decrypt. The *Decrypt* algorithm is executed by the *Data User* to decrypt the data using the applied private keys from the *Authority*.

3.2.2 Our Construction. Suppose we have n authorities that are accessible to handle the requests from users in the

ABE system. These authorities are independent with each other. Let $\mathcal{A}_i$ be the i-th authority.

Setup$(U, g, \mathcal{G}) \rightarrow (PK_i, MSK_i)$:

The *Setup* algorithm takes the global parameters and the attributes set as the input to generate the public key and master secret key. For each authority $\mathcal{A}_i$, given the universe of attributes U and a generator g of $\mathcal{G}$ with order p, the $\mathcal{A}_i$ generates its parameters independently. The authority $\mathcal{A}_i$ first randomly chooses elements $\alpha_i, a_i \in \mathcal{Z}_p$ as the random exponents, and selects $x_{i,j} \in \mathcal{Z}_p$ for each attributes in U. Then it computes the public key as follows:

$$PK = \{g, e(g,g)^{\alpha_i}, g^{a_i}, att_{i,j} = (g^{x_{i,j}})_{\forall j \in U}\}$$

And keeps the master key as follows:

$$MSK = \{\alpha_i\}$$

KeyGen$(MSK_i, S_i) \rightarrow K_i$:

The *KeyGen* algorithm takes the master secret key and users' attribute set as the input to generate users' private key. For the given attribute set S_i, the authority $\mathcal{A}_i$ first generates random elements $t_i \in \mathcal{Z}_p$. Then computes the private key as follows:

$$K_i = \{k_1 = g^{\alpha_i} \cdot g^{a_i t_i}, k_2 = g^{t_i}, k_3 = (att_{i,j}^{t_i})_{\forall j \in S_i}\}$$

Encrypt$(M, (A, \rho), PK_i) \rightarrow CT$:

The *Encrypt* algorithm takes the message, public keys from all authorities, and the specified access structure to generate the final ciphertext. The access structure includes an $n \times l$ access matrix A with function ρ mapping each row to an attribute. It first chooses a random secret $s \in \mathcal{Z}_p$ and random elements $\{y_k\}_{1 \le k \le l-1} \in \mathcal{Z}_p$ to construct a vector $v = (s, \{y_k\}_{1 \le k \le l-1})$. For each row of A, it chooses a random element $r_x \in \mathcal{Z}_p$. Then it randomly generates a sequence $Q = \{q_1, q_2, ..., q_i, ..., q_l\}_{1 \le q_i \le n}$. Finally, it computes the ciphertext as:

$$C = M \prod (e(g,g)^{s\alpha_{q_i}})_{\forall q_i \in Q}$$

$$C^{'} = g^{s}$$

$$C_x = g^{a_{q_i} \vec{A}_x \vec{v}^T} \cdot att_{q_i,x}^{-r_x}$$

$$D_x = g^{r_x}$$

And output the ciphertext as:

$$CT = \{C, C^{'}, \{C_x, D_x\}_{x \in (A,\rho)}, (A, \rho), Q\}$$

Decrypt$(CT, K_i, PK_i) \rightarrow M$:

The *Decrypt* algorithm takes the ciphertext, the public keys from all authorities, and the private keys applied from all authorities for attribute set S. Suppose that S satisfies the (A, ρ) and let $I_{(A,\rho)} \subset \{1, 2, ..., l\}$ be defined as $I_{(A,\rho)} = \{x : \rho(x) \in S\}$. Let $\omega_x \in \mathcal{Z}_p$ be a set of constants where $\sum_{x \in \mathcal{I}} \omega_x \vec{A}_x = (1, 0, ..., 0)$. Then it extracts the sequence Q from the ciphertext CT and selects the elements from K_i and PK_i based on the sequence. Finally, the message M is recovered as follows:

$$M = C \cdot \frac{\prod (e(C_x, k_{i,2}) \cdot e(D_x, k_{i,3}))^{\omega_x}}{\prod e(C^{'}, k_{i,1})}$$

3.3 Insider Threat Mitigation Solutions

Here we first present two specific insider threat issues associated with the authority.

- In the environment without insiders' collusion, how to prevent an insider from a single authority directly compromising the ABE system?
- In the environment with insiders' collusion, how to prevent insider threat resulting from different authorities collaborating with each other compromising the ABE system?

For the first insider threat, our proposed multi-authority CP-ABE scheme in Section 3.2 can directly prevent the insider's attack from a single authority. In the single-authority scenario, it is not possible to prevent the insider from stealing the confidential credentials. In the multi-authority scenario, the data is encrypted using the components from different public keys and the private keys are also generated by different authorities. Even though an insider threat agent can get the confidential credentials from his/her authority, he cannot decrypt the users' ciphertext.

For the second insider threat, we present two approaches to mitigate the collusion of insiders who are from different authorities. We named the two approaches as I_N *tolerance* and I_{N-1} *tolerance* as shown in Definition 3.2 and Definition 3.3, respectively.

Definition 3.2. I_N *tolerance.* Suppose there are N authorities in the multi-authority ABE environment. The ABE system can resist the insiders' collusion attack from all N authorities.

Definition 3.3. I_{N-1} *tolerance.* Suppose there are N authorities in the multi-authority ABE environment. The ABE system can resist the insiders' collusion attack from at most $N-1$ authorities.

3.3.1 I_N *Tolerance.* This approach can let the ABE system be tolerant from insiders from the authority set with size N in the multi-authority environment. Our proposed MA-CP-ABE scheme supports multi-authority that indicates any party can simply act as an ABE authority. Consequently, to prevent the insiders' collusion completely, the data owner can play as an ABE authority itself, namely, self-authority. Then the data owner ensures that at least one component related to the attributes (e.g., $e(g,g)^{\alpha_i}, att_{i,j}$) should come from the self-authority during the encryption phase. Even though the insiders from the N authorities collaborate to break the ciphertext, they cannot acquire enough key components (e.g., k_1, k_3) to break the ciphertext. This is because the data owner has at least one key component, and can not leak his secret credentials to these insiders.

However, the I_N *tolerance* solution has its limitation. One responsibility of the authority is to provide key service, generating the users' private keys K_i based on the attributes. Consequently, the self-authority should be available when the data user needs the key services.

3.3.2 I_{N-1} *Tolerance.* This approach can ensure the ABE system that resist insiders from the $N-1$ authority parties in

Algorithm 1 The sequence Q generating algorithm.

Input: the number of attributes in the access structure l; the number of authorities N; the identity set of authorities $S_{\mathcal{A}}$.
Output: the generated sequence Q.

```
if l ≥ N then
    Q_A ← select all identities from S_A.
    Q_rest ← randomly select l − N identities from S_A.
    Q ← Q_A ∪ Q_rest
    Shuffle the Q.
else
    Q ← randomly select l identities from S_A.
    Shuffle the Q
end if
return Q
```

the multi-authority environment. That indicates the solution can resist at most $N-1$ insiders among the N authorities based on probability. Unlike I_N *tolerance*, this solution does not require self-authority. As the sequence Q represents the identities of the authorities where the encryption algorithm will extract the components for each attribute, the idea is to randomly select the sequence Q from different authorities. Thus, the key issue is how to generate the sequence Q, as showing in Algorithm 1. As the length of the sequence is different from the number of authorities, there are two cases to be considered.

- $l \geq N$, which indicates the number of attributes in the access policy is greater than or equal to the number of authorities.
- $l < N$, which indicates the number of attributes in the access policy is less than the number of authorities.

Here, we discuss in detail about Algorithm 1. In the case of $l \geq N$, we first select all identities from the N authorities to ensure that we have enrolled all authorities. Then we randomly select the $l-N$ identities to fill the rest positions. Finally, we shuffle the sequence to keep its randomness. In the case of $l < N$, it is much simpler as we can just randomly select l authorities first and then shuffle the sequence.

4 DISCUSSION AND ANALYSIS

4.1 Security Analysis

4.1.1 Security of MA-CP-ABE. The security proof for multi-authority CP-ABE systems by the following simulation game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Consider the following game that is similar to simulation game in [4, 12]:

- **Setup** The adversary sends a list of attribute sets to each authority. The challenger generates public parameters using *Setup* algorithm and sends them to the adversary.
- **Secret Key Queries** The adversary can make as many secret key queries as it wants to the authorities by providing the sets of attributes $S_1, ..., S_n$ Then the challenger responds the query request by providing the adversary the corresponding secret keys.
- **Challenge** The adversary should specify two equal length messages M_0 and M_1 and an access structure (A, ρ) such that none of sets of attributes $S_1, ..., S_n$ from *Secret Key Queries* phase satisfy the access structure. The challenger flips a random coin $b \in \{0, 1\}$. Then the challenger encrypts M_b under the access structure (A, ρ) and sends the ciphertext to the adversary.
- **More Secret Key Queries** Similar to the *Secret Key Queries* phase, the adversary could make more secret key queries with several sets of attributes under the same constriction that the attribute sets can not satisfy the access structure in the *Secret Key Queries* phase.
- **Guess** The adversary outputs a guess $b^{'}$ that message $M_{b'}$ has been encrypted.

The adversary $\mathcal{A}$ is claimed to be successful if it can correctly identify the ciphertext, i.e., if $b = b^{'}$. The advantage of an adversary $\mathcal{A}$ in this game is defined as $\Pr[b^{'} = b] - \frac{1}{2}$.

THEOREM 4.1. *A multi-authority ciphertext-policy attribute based encryption scheme is secure if all polynomial time adversaries have at most a negligible advantage in the above simulation game.*

Here we present a brief proof to the *Theorem* 4.1.

PROOF. The multi-authority CP-ABE scheme is built on the single-authority CP-ABE that is proposed by Waters [12]. And the security assumption that our proposed work relies on is decisional q-parallel Bilinear Diffie-Hellman Exponent (BDHE) assumption.

The methodology about the proof procedure is a kind of reduction proof. The adversary $\mathcal{A}$ tries to break our scheme, while challenger $\mathcal{C}$ tries to solve the mathematical hard problem by taking the advantage of the adversary $\mathcal{A}$. For instance, suppose that the adversary $\mathcal{A}$ has non-negligible advantage $\epsilon = Adv_{\mathcal{A}}$ to break the simulation game. Then the challenger $\mathcal{C}$ can break the q-parallel BDHE problem by taking the adversary $\mathcal{A}$'s non-negligible advantage ϵ. However, the q-parallel BDHE assumption is proved, i.e., no polynomial time algorithm has a non-negligible advantage in solving BDHE challenge. Therefore, $\mathcal{A}$ does not have non-negligible advantage to break our system.

Note that the basic architecture of our proposed scheme is the same as that of [12]. We conclude that the differences in the scheme construction are:

(1) The first difference is the construction of component C in the encryption phase. We multiply more public key component $e(g,g)^{s\alpha_i}$.
(2) The second difference is the random selection of component $att_{i,j}$ in the encryption phase.

Based on the security proof presented in [12] and since our proposed scheme builds on that with differences are not only

as mentioned here, we believe the proof of security of MP-CP-ABE can be shown by simply extending earlier proof. Thus, we do not present the specific proof of our proposed scheme here. □

4.1.2 Insider Tolerance Analysis. There are two types of attack in authority insiders: collusion based and non-collusion based.

In the non-collusion case, even though the individual insider could acquire confidential credentials from the corresponding authority, they are not able to break the users' ciphertext because of the multi-authority environment.

In the collusion case, suppose the probability that the authority $\mathcal{A}_i$ has an insider is p_i. The theoretical probability of the insiders could be calculated as $p_{insider} = \prod_{i \in Q} p_i$. For the two approaches mentioned in Section 3.3, we analyze the probability of an insider attack as follows:

- I_N tolerance:
  In this approach, the self-authority can be viewed as the $(n+1)$th authority in the ABE system, while $p_{n+1} = 0$. Because the data owner itself could not be the insider to break its ciphertext. Consequently, the probability $p_{insider} = \prod_{i \in Q} p_i \cdot p_{n+1} = 0$. This indicates that the approach can fully prevent the insiders from the authorities.
- I_{N-1} tolerance:
  In this approach, despite the probability p_i, the overall probability is also related to the selection of authorities. There are two cases: if $l < N$, $p_{insider} = \frac{C_{l-1}^{N}}{C_{l}^{N}} \prod_{i \in Q} p_i = \frac{l}{N-l+1} \prod_{i \in Q} p_i$; if $l \geq N$, we should use all authorities, thus $p_{insider} = \prod_{i \in Q} p_i$. For the worst case, $p_{insider} = 1$, it requires $p_i \to 1$. However, if there is one honest authority, $p_x = 0$, the final probability $p_{insider} = 0$. Thus our approach can resist at most $N-1$ insiders out of N authorities.

4.2 Complexity Analysis

In this section, we analyze the complexity of our proposed MA-CP-ABE scheme. To the best of our knowledge, there is only one other proposed multi-authority CP-ABE scheme [8]. We compare the encryption and decryption efficiency of our proposed scheme with that of [8] theoretically.

In the ABE schemes, the main complexity is related to the computing on the exponent and the bilinear map. Thus, we analyze these computation times for these two in the two schemes, which are shown in Table 1. From the table, we can see that our proposed scheme is more efficient than that of the [8] both in encryption and decryption costs, respectively.

4.3 Correctness

Here we give the correctness proof of our proposed multi-authority CP-ABE scheme. We first calculate a temporary component T as follows:

$$
\begin{aligned}
T &= \frac{\prod_{i \in Q} e(C^{'}, k_{i,1})}{\prod_{i \in Q, x \in I} (e(C_x, k_{i,2}) e(D_x, k_{i,3}))^{\omega_x}} \\
&= \frac{\prod_{i \in Q} e(g^s, g^{\alpha_i} \cdot g^{a_i t_i})}{\prod_{i \in Q, x \in I} (e(g^{a_i \vec{A}_x \vec{v}^T} \cdot att_{i,x}^{-r_x}, g^{t_i}) e(g^{r_x}, att_{i,j}^{t_i})))^{\omega_x}} \\
&= \frac{e(g,g)^{\sum_{i \in Q} s(\alpha_i + a_i t_i)}}{e(g,g)^{\sum_{i \in Q} (a_i t_i \sum_{x \in I} \vec{A}_x \vec{v}^T \omega_x)}} \\
&= \frac{e(g,g)^{\sum_{i \in Q} s(\alpha_i + a_i t_i)}}{e(g,g)^{\sum_{i \in Q} a_i t_i s}} \\
&= e(g,g)^{\sum_{i \in Q} s \alpha_i}
\end{aligned}
$$

Then the message M could be recovered as follows:

$$
\frac{C}{T} = \frac{M \prod (e(g,g)^{s \alpha_{q_i}})_{q_i \in Q}}{e(g,g)^{\sum_{i \in Q} s \alpha_i}} = \frac{M e(g,g)^{\sum_{q_i \in Q} s \alpha_i}}{e(g,g)^{\sum_{i \in Q} s \alpha_i}} = M
$$

The specific inference presented above shows that our construction of *MA-CP-ABE* scheme could correctly recover the encrypted data. That indicates our proposed scheme achieves the correctness requirement.

5 RELATED WORK

Attribute based Encryption (ABE) scheme is first proposed by Sahai and Waters [11], which provides both confidentiality feature and access control function on the encrypted data by specifying an access policy over the users' attributes. The initial version of Ciphertext-Policy Attribute based Encryption (CP-ABE) is proposed by Bethencourt et al. [2]. In CP-ABE scheme, the access structure is associated with ciphertext and users' private key is associated with their attributes. Thus it is applicable in the cloud storage scenarios.

Multi-authority issues have been addressed in the literature [3–5, 9]. The initial work of multi-authority in ABE system is presented by Chase [4], where he proposes a hierarchical authority architecture that includes a central authority and several attribute authorities. Then Božović et al. [3] try to reduce the importance of the central authority and propose a *honest-but-curious* central authority. The scheme from Gorasia et al. [5] is also based on this architecture but more efficient decryption algorithms.

Two multi-authority ABE schemes, proposed in [8, 9], respectively, incorporate distributed authorities that indicates no requirement for any global coordination (central authority); the scheme proposed in [8] is a CP-ABE scheme, while the scheme proposed in [9] is a KP-ABE scheme. A KP-ABE scheme is not useful in the cloud storage scenarios, and the scheme proposed in [8] cannot be applied in our proposed insider threat mitigation solutions. Even though the users' collusion that can be viewed as a kind of insider threat in ABE system has been addressed in these schemes, the insider threat from the authority has not been considered. These schemes always assume that the authority as fully trusted

Table 1: Comparison of efficiency

schemes	Our scheme	[8]
Encryption	$(4l+1)\mathcal{C}_{exp}$	$(4\|i\|+1)\mathcal{C}_{exp} + \|l\|\mathcal{C}_{map}$
Decryption	$3\|S\|\mathcal{C}_{map} + \|S\|\mathcal{C}_{exp}$	$3\|S\|\mathcal{C}_{map} + 3\|S\|\mathcal{C}_{exp}$

[1] Let $|\mathcal{C}_{exp}|$, $|\mathcal{C}_{map}$ be the calculation of exponent and bilinear map over $\mathcal{G}$, respectively.
[2] l is the attribute number in the access structure, and $|S|$ is the minimum set of users' attributes.

or *honest-but-curious*, which may not be the case as we have discussed in this paper.

6 CONCLUSION

Recently, cloud storage service is being widely used over the Internet as it provides convenience for users in terms of data storage and management. However, users' concern on the security and privacy issues of the sensitive data stored in the cloud is the main obstacle in the successful uses of these cloud based applications. Even though the Ciphertext-Policy Attribute based Encryption provides both confidentiality and access control features for the cloud storage scenarios, the insider threat in the ABE system, especially with regards to the authority component, has not been addressed in the literature. In this paper, we focus on analyzing the insider threat in the authority component of the ABE system and have proposed technical solutions to mitigate these insider threat by proposing new multi-authority CP-ABE scheme, and two insider threat mitigation solutions based on the MP-CP-ABE schemes. Also, the security proof of the proposed multi-authority CP-ABE scheme has been presented. Based on the analysis, our proposed approach is efficient as can be seen from the complexity analysis and can mitigate the insider threat effectively.

ACKNOWLEDGMENT

Joshi, Krishnamurthy, and Tipper are supported by NSA Cybersecurity grant BAA-003-1.

REFERENCES

[1] Amos Beimel. 1996. *Secure schemes for secret sharing and key distribution*. Ph.D. Dissertation. Technion-Israel Institute of technology, Faculty of computer science.
[2] John Bethencourt, Amit Sahai, and Brent Waters. 2007. Ciphertext-policy attribute-based encryption. In *IEEE Symposium on Security and Privacy, 2007. SP'07*. IEEE, 321–334.
[3] Vladimir Božović, Daniel Socek, Rainer Steinwandt, and Viktória I Villányi. 2012. Multi-authority attribute-based encryption with honest-but-curious central authority. *International Journal of Computer Mathematics* 89, 3 (2012), 268–283.
[4] Melissa Chase. 2007. Multi-authority attribute based encryption. In *Theory of Cryptography Conference*. Springer, 515–534.
[5] Nikita Gorasia, RR Srikanth, Nishant Doshi, and Jay Rupareliya. 2016. Improving Security in Multi Authority Attribute Based Encryption with Fast Decryption. *Procedia Computer Science* 79 (2016), 632–639.
[6] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. 2006. Attribute-based encryption for fine-grained access control of encrypted data. In *Proceedings of the 13th ACM conference on Computer and communications security*. ACM, 89–98.
[7] Junbeom Hur. 2013. Improving security and efficiency in attribute-based data sharing. *Knowledge and Data Engineering, IEEE Transactions on* 25, 10 (2013), 2271–2282.
[8] Allison Lewko and Brent Waters. 2011. Decentralizing attribute-based encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 568–588.
[9] Riccardo Longo, Chiara Marcolla, and Massimiliano Sala. 2015. Key-policy multi-authority attribute-based encryption. In *International Conference on Algebraic Informatics*. Springer, 152–164.
[10] Yannis Rouselakis and Brent Waters. 2013. Practical constructions and new proof methods for large universe attribute-based encryption. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 463–474.
[11] Amit Sahai and Brent Waters. 2005. Fuzzy identity-based encryption. In *Advances in Cryptology–EUROCRYPT 2005*. Springer, 457–473.
[12] Brent Waters. 2011. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In *Public Key Cryptography–PKC 2011*. Springer, 53–70.