Privacy Preserving Attribute Based Encryption for Multiple Cloud Collaborative Environment

Nikos Komninos
Department of Computer Science
City University London,UK

Aisha Kanwal Junejo
Department of Computer Science
City University London,UK

***Abstract*—In a Multiple Cloud Collaborative Environment (MCCE), cloud users and cloud providers interact with each other via a brokering service to request and provision cloud services. The brokering service considers several pieces of data to broker the best deal between users and providers which can subsequently risks the privacy and security of MCCE. In this paper, we propose a Privacy Preserving Attribute-Based Encryption(PP-ABE) scheme which protects MCCE from a compromised broker.**

The proposed encryption scheme preserves the privacy by employing data access policy over sets of attributes. The identifying attributes are anonymoized using pseudonyms. The data access policy is further anonymized so as it remain unknown to unauthorized parties. The PP-ABE achieves unlinkability between different data items which flows through the collaborative cloud environment and preserves the privacy of cloud users and cloud providers.

I. Introduction

Multiple Cloud Collaborative Environment (MCCE) is emerging as the new facade of cloud computing. A multiple cloud collaborative environment is conceptually similar to an e-commerce platform where several cloud providers outsource services to different users. International Organization for Standardization defines a cloud service broker (CSB) [1] as a "cloud service partner that negotiates relationships between cloud service customers and cloud service providers." Such a collaborative environment is quiet different from traditional cloud setup where all the resources belong to a single cloud provider whereas in a multiple cloud environment resources belong to different cloud providers. Each resource being completely distributed, heterogeneous, and totally virtualized. The MCCE is facilitated by an intermediary (brokering service) for efficient matching of cloud services (offered by a multitude of service providers) to satisfy user requirements. A brokering service is responsible for selecting optimal cloud resources to deploy a service, optimally distribute the different components of a service among different clouds, and to move a given service component from one cloud to another to satisfy the optimization criteria.

The basic components of MCCE under our consideration are cloud service users, brokering service, and cloud service providers. The steps of a typical transaction between different components are underlined below. Cloud Service Providers register their services with the brokering Service. Cloud Users specify their requirements and preferences (under business policy compliance) for selection of a reliable and trustworthy cloud service provider. The service user negotiates Service Level Agreement (SLA) with the service provider and make a SLA contract. The brokering service (resource matching module) selects and composites highly trusted resources and allocates to users from the trusted resource pool. The brokering service sorts high performance resources by analysing the history information of the resources for dynamically providing the highly trusted resources. Users provide ratings for cloud service(s) at the end of transaction. The brokering service collects locally-generated user's ratings and aggregates with the directly monitored state information i.e. Quality of Service (QoS) attributes to compute the global evaluation score. The global evaluation score is the overall degree of trust for a cloud service (i.e. aggregation of user feedback and directly monitored quality of service information).

It is imperative to underline the role of cloud brokering service in service provisioning cycle; and it is more important to do so for a multiple cloud collaborative environment. The cloud brokering service can be vulnerable for MCCE and it is therefore vital to adopt some security measures to control its behavior. The brokering service discussed in [2] takes into consideration cloud user's requirements, preferences, business policy and QoS attributes of cloud providers to perform the above mentioned tasks. The storage and processing of such detailed information records(consisting of both identifying and non-identifying attributes) by a third party brokering service raises the security and privacy concerns. The brokering service can collude with the competitor cloud provider(s) or use the information for marketing purposes etc. The privacy requirements of a MCCE are discussed below.

A. Requirements of a Privacy Preserved Multiple Cloud Collaborative Environment

- **Unlinkability:** There must not be any link(or connectivity) between cloud user's details and its requirements, preferences, and feedback such that it could be used for malicious purposes.
- **Anoymization:** All the identifying details must be anonymized to preserve the privacy of collaborating entities.
- **Control and Audit:** The brokering service must not black-list any user on the submission of a negative feedback. The brokering service must not perform maliciously and be bound by loyalty to a single cloud service provider. It must not report wrong monitoring data (i.e. modifies the QoS attributes) to support a particular cloud service provider. The brokering service must not collude with the competitors cloud services and harm the overall reputation of a trustworthy cloud service provider.

B. Our Contribution

Privacy preservation is therefore the absolutely essential requirement of every brokering service; it is indeed the motivation of our research to propose a privacy preserving encryption scheme for both the cloud users and cloud service providers such that they are protected from any type of malicious behavior of the brokering service.The security and privacy can be guaranteed by considering the attribute based encryption scheme. In this paper, we propose a privacy preserving attribute-based encryption scheme which uses cipher-text policy to anonymize attributes in order to achieve unlinkability between different data items which flows through the collaborative cloud environment. *Attribute Anonymyzation* is achieved by considering the hidden data access policy.

C. Organization

The remainder of this paper is organized as follows. In section II, we present the most important studies related to cloud brokering services and privacy preserving encryption schemes. In section III, we discuss data access policy and the basic terms we use in the rest of the paper. In section IV, we present the privacy preserving scheme, while in section V we provide a security analysis of our proposed scheme. Finally, in section VI we conclude the paper.

II. Related Work

We review recent studies related to cloud brokering services(CBS) to underline their role, contribution, and requirements in cloud computing environments. We further review some privacy preserving attribute base encryption schemes.

In [3] authors discuss the current practices and upcoming challenges of cloud brokering. The legal and practical implications of cloud brokering are highlighted. It is maintained that there can be different motivations(i.e. economical, special security, compliance regulation or specific requirements) behind the use of a brokering service but in any case the role of brokering service in data processing chain and respective liabilities must be clearly defined otherwise the security and privacy issues would become complex. It is argued that the brokering service must be independent and must not be bound by loyalty to a single company. The legal frameworks applicable to personal data must explicitly address the problems of data location, cross-border transfer, portability, access and accountability in wake of cloud brokering service appearing as a new phenomenon in cloud computing.

An open federated cloud computing model called Reservoir is presented in [4]. The work is motivated by the fact the cloud computing has the potential of becoming a future service technology and that vision could be materialized by addressing the deficiencies in current cloud models. The identified deficiencies include 1) inherently limited scalability of single-provider clouds, 2) lack of interoperability among cloud providers, and 3) no built-in support for business service management. The Reservoir Model proposes an architecture where different cloud service providers can collaborate with each other to create a seemingly infinite pool of IT resources.

The Reservoir architecture [4] consists of multiple components like service provider, infrastructure provider, virtual execution environment(VEE), reservoir site, reservoir cloud and various other sub-components. A service application is executed and deployed on a number of virtual hosts to ensure an on-demand provision of resources and services. The paper does not discuss the roles(i.e. data owner, data controller, and data processor) and liabilities of these various components in service provisioning cycle. The lack of focus on these aspects affects the end user and risks it privacy and security.

A cloud brokering service based on the resource allocation perspective is proposed in [2]. The T-Broker service is a middle-ware for cloud trust management and service matching. T-Broker monitors the cloud services and computes the overall trust degree of the cloud service provider by aggregating the experience-based trust and feedback trust. The experience-based trust is computed by monitoring cloud services on five kinds of QoS attributes(i.e. node spec profile, average resource usage information,average response time, average task success ratio, and the number of malicious access). The feedback trust is computed from the ratings of the users. The computed trust value is used to allocate the most trustworthy resources to users as per their requirements. Again, the proposed scheme does not discuss the security and privacy issues, the trustworthiness of the brokering service and the ways to control any malicious behavior on part of the brokering service.

The concept of a decentralized meta-broker for inter-cloud setting is introduced in [5]. Meta-broker enables the collaboration and inter-operation among several disperse (and highly likely heterogeneous) clouds. The idea is to develop coordination among different clouds for establishing a reactive cross-exchange and service automation process while offering transparency to users. Every cloud has a separate meta-broker which acts on behalf of its clients to map user requests to cloud data-centers. These meta-brokers establish inter-cloud collaboration by exchanging information such as list of datacenter hosts, VMs, jobs (cloudlets) along with the characteristics of the datacenter (e.g. hosts CPU, Memory, number of processing elements (PEs), architecture etc.). This work recognizes the exchange of information between meta-broker can lead to security, privacy, and trust issues; and proposes the use of encryption techniques such as a shared key authentication process for fundamentally gaining secure access among clouds.

A privacy preserving constant size cipher-text attribute-based encryption scheme is proposed in [6]. The two unique features of this scheme are privacy preservation and hidden data access policies. The recipient's privacy is preserved by leveraging a hidden data policy construction. In this scheme,the cipher-text size remains constant with any given number of attributes. Our proposed scheme follows similar approach with [6] for user privacy preservation and attribute anonymization.

III. Preliminaries of PP-ABE

In this section, we discuss the notion of attributes, data access policy, and anonymized data access policy. First, we define these terms in an abstract way and then correlate them to a collaborative cloud computing environment. Table I lists the symbols we have used through out the paper.

A. Attributes, Policy, and Anonymity

Let $U = \{A_i\}$ $i \in$ [1, k] be the Universe of attributes and k is the number of attributes in the universe. Each attribute A_i has 3 values: $\{A_i^+, A_i^-, A_i^*\}$. Intuitively, A_i^+ denotes the user has A_i ; A_i^- denotes the user does not have A_i or it's not a proper attribute of the user. As for A_i^*, it denotes a "wild-card" value, which means policy does not care about the value of attribute A_i. The attribute with a wild-card value does not contain any identifying value and therefore does not

TABLE I. SYMBOLS AND THEIR DESCRIPTION

Symbol	Description
U	Universe of Attribute
A_i	ith Attribute of user
k	Number of attributes in the universe
L	Attribute list of the user
P	Data Access Policy
$\overline{P}$	Anonymized Data Access Policy
λ	Security Parameter
e	Bilinear Mapping
K_P	Public Key
$K_{P_{Ai}}$	Public Key of the ith Attribute
K_M	Master Key
K_S	Private Key of the User
$K_{S_{Ai}}$	Private Key of the ith Attribute
K_{SYM}	Symmetric Key
C_T	Cipher-Text

need to be privacy preserved. On the other hand, the attributes with identifying values are anonymized using "do-not care" values to preserve the privacy. When a user joins the system, it is tagged with an attribute list defined as follows: A user's attribute list is defined as L = { L [i] $i \in$ [1, k] }, where L [i] $\in$ {A_i^+, A_i^- }.

AND-Gate Access Policy: Let P = { P [i]}$i \in$ [1, k] be an AND- Gate access policy, where [P] $\in$ {A_i^+, A_i^-, A_i^* }. The access policy contains both identifying and non-identifying attributes. We use the notation $L \vdash P$ to denote that the attribute list L of a user satisfies P, as:

$$L \vdash P \Leftrightarrow P \subset L \bigcup \{A_i^*\} i \in [1, k]$$

A_i^+ or A_i^- requires the exact same attribute in the user's attribute list. Effectively, each user with A_i^+ or A_i^+ fulfills A_i^* automatically.

Anonymized AND-Gate Access Policy: Accordingly, we also define an Anonymized AND-gate policy that removes all identifying attribute values, i.e. { A_i^+, A_i^+}, except wild card values, i.e. A_i^*. Formally, an anonymized AND-gate policy is defined as follows: Let

$$\overline{P} = P \bigcap \{A_i^*\} i \in [1, k]$$

be an anonymized AND-gate access policy. We note that the "do-not care" attribute values are included in the anonymized access policy. If we hide the "wild card" attributes, the decryptor will need to guess 2_k possible access policies if there are k attributes in the policy, i.e., for its value can be either A_i^* or the specific value (A_i^+ or A_i^-) assigned to the decryptor. This would make the scheme infeasible in terms of performance. The anonymity policy is defined as the state of being not identifiable within a set of subjects, i.e., the anonymity set. As the access policy is one-to-many mapped to users, we can extend this definition of policy anonymity set of blinded policy as the anonymity set of blinded policy $\overline{P}$ is the set of access policies which are identically blinded to P. Here, we briefly analyze the anonymity level of the blinded access policy. If there are no "wild-cards" in the original access policy (hidden), the blinded policy $\overline{P}$ will be empty. In this case, the size of anonymity set is 2_k, as there are j "wild-cards" in the original access policy(hidden), the size of the anonymity set is 2_{k-j}.

Bilinear Maps: A pairing is a bilinear map e: $G_0 \times G_0 \rightarrow G_1$, where G_0 and G_1 are two multiplicative cyclic groups with large prime order p where the discrete logarithm problem on both G_0 and G_1 is considered hard to solve. Pairing has following properties:

- Bilinearity:
$$e(P^a, Q^b) \ = e(P, Q)^a b, \forall P, Q \in G_0, \forall a, b \in Z_P^*.$$
- Nondegeneracy:
$$e(g, g) \neq 1$$
- Computability: A mapping is said to be computable if an algorithm exists which can efficiently computes $e(P$,$Q)$ for any P, $Q \in G_0$.

Complexity Assumption: The security of our proposed scheme is based on a complexity assumption called the Bilinear Diffie-Hellman Exponent assumption (BDHE).

Let G_0 be a bilinear group of prime order p. The K-BDHE problem in G_0 is stated as follows: given the following vector of 2K+1 elements (Note that $g^{\alpha^{K+1}}$ is not in the list):

$$(h, g, g^{\alpha}, g^{\alpha^2}, ..., g^{\alpha^K}, g^{\alpha^{K+2}}, ...g^{\alpha^{2K}}) \in G_0^2 k + 1$$

as the input and the goal of the computational K-BDHE problem is to output $e(g$,$h)^{\alpha^{K+1}}$.

Decisional K-BDHE: The decisional K-BDHE assumption is said to be held in G_0 if there is no probabilistic polynomial time adversary who is able to distinguish

$$\left\langle h, g, Y_{g,\alpha,K}, e(g,h)^{\alpha^{(K+1)}} \right\rangle$$

and

$$\left\langle h, g, Y_{g,\alpha,K}, e(g,h)^R \right\rangle$$

with non-negligible advantage, where $\alpha, R \in Z_P$ and $g, h \in G_0$ are chosen independently and uniformly at random.

B. Privacy Preservation for MCCE

As has already been mentioned, the different entities in multiple cloud collaborative system interact with each other to request cloud service(s), provide cloud service(s), and broker the services between the cloud user(s) and cloud provider(s). An interaction between any of the two entities takes place on the basis of pre-agreed terms and conditions (i.e. Service Level Agreement(SLA) and policies). The policies and SLA(s) in turn consist of various data tuples and each tuple consist of several data attributes. Some of the data attributes contain the identifying information (i.e. IDs, personal details etc.) which must not be revealed to other entities in the system. Table II shows some attributes of cloud users such as **ID**, **Name**,**Requirement**, **Feedback**, **Preferences**, **SLA**, and **Cloud Provider ID**. Apart from ID and Name all other attributes are denoted by a unique identifier for that particular attribute. RQ_{ID1} denotes the requirements identifier for cloud user CU_1. Similarly, FD_1 , PR_{ID1}, SLA_{ID1}, and CP_1 denotes the feedback, preferences, SLA, and cloud provider identifiers respectively. The attributes ID and Name represent user's personal details whereas other attributes represent the linked information. We believe that the identifying attributes must be anonymized in order to preserve the privacy of the user(s). The anonymization is achieved by a hidden data access policy that specifies which attributes are to be disclosed and which not. The identifying attributes are anonymized by pseudonyms which are obtained from the registration authority.

Table III lists data access policies (P_1 and P_2) and anonymized data access policies ($\overline{P1}$, $\overline{P2}$) for cloud users C_{U1} and C_{U2} respectively. The anonymized data access policy

TABLE II. CLOUD USER ATTRIBUTES

Attributes	L(1)	L(2)	L(3)	L(4)	L(5)	L(6)	L(7)
Description	CU_{ID}	Name	Requirement	Feedback	Preference	SLA	CP_{ID}
CloudUser1	CU_1	Alice	RQ_{ID1}	FD_1	PR_{ID1}	SLA_{ID1}	CP_1
CloudUser2	CU_2	John	RQ_{ID2}	FD_2	PR_{ID2}	SLA_{ID2}	CP_2

TABLE III. DATA ACCESS POLICY AND HIDDEN DATA ACCESS POLICY

Attributes	L(1)	L(2)	L(3)	L(4)	L(5)	L(6)	L(7)
Description	CU_{ID}	Name	Requirement	Feedback	Preference	SLA	CP_{ID}
PolicyP1	A_1^+	A_2^+	A_3^+	A_4^-	A_5^+	A_6^+	A_7^+
Policy$\overline{P1}$	$A^{--\$--}$	$A^{--\$--}$	A^*	A^*	A^*	A^*	$A^{--\$--}$
PolicyP2	A_1^+	A_2^+	A_3^+	A_4^-	A_5^+	A_6^+	A_7^+
Policy$\overline{P2}$	$A^{--\$--}$	$A^{--\$--}$	A^*	A^*	A^*	A^*	$A^{--\$--}$

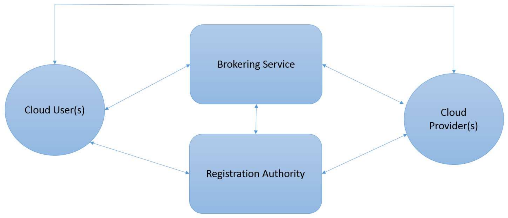

Fig. 1. Privacy Preserved Multiple Cloud Collaborative Environment

specifies the user details by "do not care" and "wild card" attributes. $A^{--\$--}$ denotes “do not care” attribute which means value is hidden (or privacy preserved) and A^* denotes a "wild card" value which means the policy does not care about the value of the attribute.

Whenever the cloud user sends some information to the brokering service or to cloud provider its identifying attributes are replaced with the "do not care" attribute values. The receiving party will verify the sender details by contacting the registration authority which confirms the registration of such a user. Upon verification from registration authority, the brokering service will respond to the query. Anonymizing the identifying details with pseudonyms preserves the user’s privacy. Similar attribute tables and anonymized data access policy tables can be created for cloud providers and brokering service.

IV. DESCRIPTION OF PP-ABE

In this section, we introduce our scheme which satisfies the above mentioned requirements and offers a privacy preserving mechanism for the users (i.e. cloud users and cloud providers) of a multiple cloud collaborative environment. Figure **??** shows the basic components of the scheme which includes cloud user(s), cloud service provider(s), registration authority (RA), and brokering service. Cloud users, cloud providers and brokering service all get registered with the registration authority. The registration authority asks for a number of attributes and upon verification creates secret keys for them. There are a number of attributes in the system. The registration authority also generates various parameters which are used by different algorithms. Our scheme consists of four fundamental algorithms:

1) Setup(1^λ, k)
2) Key Generation
3) Encryption
4) Decryption

A. *Setup(1^λ, k)*

The RA initiates the setup process. There are total k attributes in the system. The setup algorithm takes input of the security parameter 1^λ and the number of attributes(k) in the system. λ denotes the size of the prime number which is the order of the cyclic group in bi-linear mapping. The size of prime number is usually λ bits long. The setup algorithm returns public key K_P and master key K_M. The public key K_P is used for encryption while the master key K_P is used for private key generation.

B. *Key Generation —KeyGen(K_P, K_M, L)*

The KeyGen algorithm takes the public key K_P, the master key K_M and the user’s attribute list L as input. It outputs the private key K_S of the user.

C. *Encryption —Encrypt(K_P, M, P)*

The encrypt algorithm takes the public key K_P, the message M and the specified access policy P (over the universe of attributes) as input. The algorithm outputs cipher-text C_T such that the only users whose private keys satisfy the access policy can decrypt the message. The cipher-text also associates the anonymized access policy P.

The encrypt algorithm is based on hybrid encryption scheme which can be constructed using public key encryption and symmetric key encryption schemes. The encryptor first encrypts the message using a symmetric key K_{SYM} and then re-encrypt it using the public key K_P and the public keys of the attributes specified in the access policy.

D. *Decryption —Decrypt(K_P, K_S, C_T)*

Before performing decryption, the receiver has little information about the access policy that is enforced over the cipher-text. Only if $L \vdash P$ then user can successfully recover the valid plain text and access policy. Otherwise, user can only get a random string which can be easily detected. Moreover, the access policy remain unknown to the unsuccessful decryptor(s). It takes the public key K_P, the private key K_S of the user and the cipher-text C_T, which only includes the anonymized access policy as input. It returns a valid plain-text M if the set of attributes of the private key satisfies the access structure of the cipher-text.

Note: Each public key is mapped to an attribute value, including A_i. To encrypt a message, the encryptor specifies

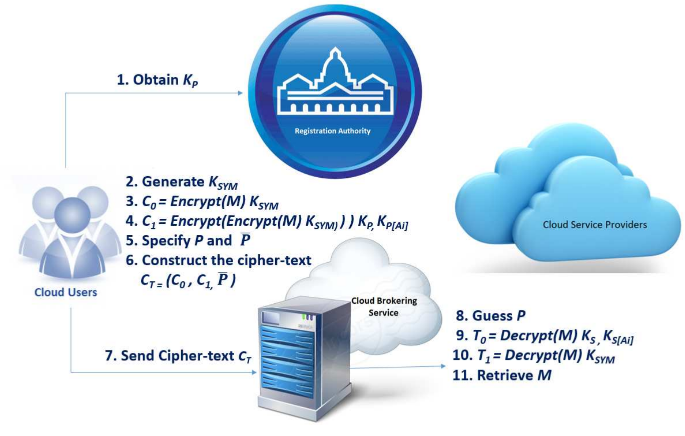
1. Obtain K_P
Registration Authority
2. Generate K_{SYM}
3. C_0 = Encrypt(M) K_{SYM}
4. C_1 = Encrypt(Encrypt(M) $K_{SYM)}$)) $K_{P,}$ $K_{P[Ai]}$
5. Specify P and $\overline{P}$
6. Construct the cipher-text
$C_{T=}$ (C_0 , $C_{1,}$ $\overline{P}$)
Cloud Users
Cloud Service Providers
Cloud Brokering Service
7. Send Cipher-text C_T
8. Guess P
9. T_0 = Decrypt(M) K_S , $K_{S[Ai]}$
10. T_1 = Decrypt(M) K_{SYM}
11. Retrieve M

Fig. 2. Privacy Preserved Multiple Cloud Collaborative Environment

an access policy P by assigning an attribute value ($A_i \in \{1, 0, *\}$) for each of the n attributes in the universe and encrypts the message using public keys of the attribute values in the P. Each decryptor is generated as a set of private key components corresponding to attribute list L. All the private key components of the same user are tied together by a common random factor to prevent collusion attacks.

E. Example

We elaborate the execution of our proposed scheme by an example in which the cloud user wants to send a message to the brokering service. The steps of the encryption-decryption process are shown in figure 2. The same steps will be followed whenever a message is to be sent between any two parties in the multiple cloud collaborative environment (MCCE).

Encryption

First of all the cloud user obtains registration authority's public key K_P. It then generates a fresh one-time symmetric key K_{SYM} and encrypts the message $Encrypt(M)_{K_{SYM}}$ using the symmetric key just generated. The result of the encryption is first part of cipher-text i.e. C_0 which is then re-encrypted $Encrypt(\ Encrypt(M)_{K_{SYM}})_{K_{P[Ai]}}$ using public key K_P and the public keys of policy attributes. This makes the second part C_1 of the cipher-text. The attributes are already defined in pre-agreed data access policy between the cloud User and the brokering service. The cloud user then specify the data access policy P and anonymizes data access policy $\overline{P}$ by removing the identifying attributes with the "do-not-care" values. The cipher-text $C_T = (C_0, C_1, \overline{P})$ is then sent to the brokering service.

Decryption

To decrypt this hybrid cipher-text, the brokering service do the following. It uses the data policy attributes to make a guess of the access policy P. The brokering service then uses the parts of private key to decrypt the first part of cipher-text and retrieve symmetric key K_{SYM} contained in. It then uses this symmetric key K_{SYM} to decrypt the message contained in the second part of the cipher-text and retrieves the plain text message M.

V. Security Analysis

In this section, we discuss the security analysis of PP-ABE and further elaborate its robustness against some attacks.

A. Adaptive Chosen Cipher-text Attacks in PP-ABE

The PP-ABE is a hybrid encryption scheme which requires both the public key encryption (PKE) and symmetric key encryption (SKE) schemes to be secure against adaptive chosen cipher-text attacks because it then inherits those properties as well. Our proposed scheme is secure for both the encryption approaches. The generation of public/ private key pairs using bilinear-mapping makes it robust against key compromise and indistinguishability attacks. The one-time generation of the randomized symmetric key makes it hard to guess the K_{Sym} symmetric key. Moreover, the K-BDHE complexity assumption makes the PP-ABE more robust and secure against common security vulnerabilities. The security of the PP-ABE scheme is based on cipher-text indistinguishably i.e. indistinguishably under chosen plain-text attack(IND-CPA) which considers that the adversary should not learn any information from seeing a cipher-text and it should not be able to do better than a random guess. The goal of adversary in anonymous attribute-based encryption system can be either one of the following: 1) extracting information of plain-text from the cipher-text or 2) distinguishing underlying access-policy in the cipher-text.

The two goals of adversary can be integrated in the indistinguishability against cipher-text-policy and chosen cipher-text attacks (CP-IND-CCA). In this paper, we consider indistinguishability against selective cipher-text policy and chosen message attack (sCP-IND-CPA) [7] [8] [9]. The definition is similar to CP-IND-CCA, although in sCP-IND-CPA, the adversary has to submit its challenge attributes before the setup phase. Here, the adversary is allowed to control some users and access their attribute private keys that do not match the cipher-text-policy. Note that, the decryption oracle is not available to the adversary. The formal definition is given based on the following sCP-IND-CPA game between an adversary and a brokering service in MCCE.

Initial: The adversary commits to the challenge cipher-text policies P_0, P_1 before setup algorithm.

Setup: The challenger chooses a sufficiently large security parameter 1^λ, and runs setup to get public key K_P and master secret key K_S. It retains K_S and give K_P to the adversary.

Phase 1: The adversary can perform a polynomially bounded number of queries to key generation oracle on attributes L, the only restriction on attribute list L is that the it either satisfies both the policies or it does not satisfy any policy,

$L \vdash P_0$ = $L \vdash P_1$ = 0 or
$L \vdash P_0$ = $L \vdash P_1$ = 1.

Challenge: The adversary outputs two messages M_0 and M_1, and it wishes to be challenged with respect to P_0 and P_1. It requires that M_0 = M_1 if any attribute private key on L satisfying L $\vdash$ P_0 = L $\vdash$ P_0 = 1 has been queried. The challenger randomly chooses a bit $b = \{0, 1\}$, computes C_T = Encrypt(K_P, M, P) and sends C_T to the adversary.

Phase 2: The adversary continues to issue queries to the key generation oracle, with the same restriction as before.

Guess: The adversary outputs a guess $\acute{b}$ for b. The adversary wins the game if $b = \acute{b}$. The advantage of adversary in game sCP-IND-CPA is considered as the probability that adversary wins the game minus 1/2. This model can be considered to be analogous to the selective-ID model [10] utilized in IBE protocols. In their security model, the adversary should commit to the challenge identity ID before the setup phase.

$$\Pr[b = \acute{b}] - \frac{1}{2}$$

The PP-ABE encryption scheme is fully secure if all polynomial time adversaries have at most a negligible advantage in sCP-IND-CPA. Our claim will remain true because of the Bilinear Diffie-Hellman Exponent (K-BDHE) assumption. The decisional K-BDHE ensures that no probabilistic polynomial time adversary would be able to guess the exponents for the private keys of the attributes. The data access policy is not revealed to any other user and the anonymity is not compromised.

B. Security Attacks in MCCE:

It is essential to protect the MCCE against security threats and vulnerabilities such that the colluding behavior of the brokering service can be controlled. The privacy requirements of the MCCE demand unlinkability of data, anonymity of users IDs, and control and audit of the collaborating entities. The privacy requirements of MCCE can be guaranteed if the PP-ABE is secure against the following security vulnerabilities:

1) Breaking the Anonymity: As we have already been discussed in MCCE the privacy of cloud users and providers must be preserved, such that no adversary can break the anonymity of the identifying attributes and be able to link pieces of information. The privacy is protected by considering the hidden data access policy which is constructed using the "do-not-care" ($A^{--\$--}$) and "wild-card" ($A^{*}$) terms. The identifying attributes (IDs) are represented by "do-not-care" ($A^{--\$--}$) terms. The RA defines the pseudonyms for each of the identifying attribute i.e., the $A^{--\$--}$ term. The pseudonyms are generated by bi-linear mapping of cyclic groups with very large prime order i.e. security parameter λ in PP-ABE. It is computationally infeasible to guess the corresponding group element. Moreover, no adversary can determine if some target element is either a special combination of given parameters or a random element because of the K-BDHE complexity assumption of PP-ABE.

2) Data Modification: The other possible attack against any encryption scheme is data modification. Its indeed the basic security requirement of PP-ABE because the adversary can modify the IDs, ratings, feedback and trust values which can cost the entire collaborative environment. The PP-ABE is a hybrid encryption which makes it secure against data tampering. If an adversary is somehow able to steal the private keys, he/she can not modify the plain-text message because of the additional symmetric key encryption.

3) ID Spoofing: The adversary can impersonate as a a real cloud user or a cloud service provider to get registered with the RA. Once it does so it can gain access to the system and make falsified service requests and provisions. The PP-ABE is protected against the ID-spoofing attacks because of the application of formally robust methods for secret key generation. Whenever a user registers with the RA it submits a list of attributes L. Each of the attribute is mapped to a random number in cyclic group of prime order p, all those numbers are then summed up together. It then goes through other group operations and finally the cloud user ID CU_{ID} is tied to another random element in G_0. A similar process is followed to generate the secret key K_S. In any case, if some adversary knows the attribute list of the user it can not submit a registration request to get exactly the same user ID CU_{ID} and secret key K_S. This prevents the PP-ABE from ID-spoofing attack.

4) White Washing: The PP-ABE is protected against the white-washing attack due to mathematically sound nature of applied techniques. The attack is facilitated by ID spoofing attack which is not possible because of the reasons already discussed above.

VI. Conclusion

In this paper, we have presented a privacy preserving attribute-based encryption(PP-ABE) scheme for a MCCE. Our proposed scheme is novel in a twofold manner: first, it preserves the privacy of cloud users and providers; and second, it curtails the malicious and colluding behavior of the brokering service. The MCCE is protected against any security and privacy risks if the brokering service is compromised. In future, we will further work on the trust and privacy issues of the MCCE. We aim to explore the trust from user's perspective and see its correlation with privacy. We will also investigate the effects of human factors in establishment of trust.

References

[1] I. I. T. T. F. (ITTF), "Information technology – cloud computing – overview and vocabulary, iso/iec 17788:2014," ISO, Tech. Rep., Oct 2014.

[2] X. Li, H. Ma, F. Zhou, and W. Yao, "T-broker: A trust-aware service brokering scheme for multiple cloud collaborative services," *Information Forensics and Security, IEEE Transactions on*, vol. 10, no. 7, pp. 1402–1415, July 2015.

[3] M. Guzek, A. Gniewek, P. Bouvry, J. Musial, and J. Blazewicz, "Cloud brokering: Current practices and upcoming challenges," *Cloud Computing, IEEE*, vol. 2, no. 2, pp. 40–47, Mar 2015.

[4] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, M. Ben-Yehuda, W. Emmerich, and F. Galan, "The reservoir model and architecture for open federated cloud computing," *IBM Journal of Research and Development*, vol. 53, no. 4, pp. 4:1–4:11, July 2009.

[5] S. Sotiriadis, N. Bessis, and N. Antonpoulos, "Decentralized meta-brokers for inter-cloud: Modeling brokering coordinators for interoperable resource management," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on*, May 2012, pp. 2462–2468.

[6] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption," *Computers, IEEE Transactions on*, vol. 64, no. 1, pp. 126–138, Jan 2015.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the 2007 IEEE Symposium on Security and Privacy*, ser. SP '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 321–334. [Online]. Available: http://dx.doi.org/10.1109/SP.2007.11

[8] L. Cheung and C. Newport, "Provably secure ciphertext policy abe," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, ser. CCS07. New York, NY, USA: ACM, 2007, pp. 456–465. [Online]. Available: http://doi.acm.org/10.1145/1315245.1315302

[9] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in *Theory of Cryptography*, ser. Lecture Notes in Computer Science, S. Vadhan, Ed. Springer Berlin Heidelberg, 2007, vol. 4392, pp. 535–554. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-70936-7_29

[10] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles," in *Advances in Cryptology - EUROCRYPT 2004*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds. Springer Berlin Heidelberg, 2004, vol. 3027, pp. 223–238. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24676-3_14