International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 05 | May-2018 www.irjet.net p-ISSN: 2395-0072

Secured Control Technique Access for Environment in Cloud Computing Using Attribute Based Hierarchical Structure and System

Kamala Devi K

Assistant Professor of Computer Science Department, Government Arts College, Ariyalur

---

Abstract - *Cloud computing has drastically condensed the computational and storage costs of outsourced data. The existing access control techniques o er users access provisions centered on the common user attributes like Roles, which reduces the engrained access measure. The paper denes a Storage Correctness and Fine grained Access Provision (SCFAP) scheme, that provides the user an exclusive access through the use of a hierarchical structure which is a combination of users unique and common attributes. Also, we deploy the concept of Token Granting system that allows the users to verify the correctness of outsourced data without the retrieval of the respective les. The tokens are derived from the metadata containing le location that helps in the process of storage correctness verification and improvises the storage efficiency. The experimental results show SCFAP has improved storage efficiency and error recovery measures than existing techniques.*

Key Words: Access control, access structure, barrier limits, storage *efficiency*, token granting system

1. INTRODUCTION

Cloud computing is one of the widely used emerging technique that oers various methods to acquire and manage IT resources on a largescale [19, 22]. Cloud computing, in turn, provides different types of services such as Infrastructure as a service (IaaS) also sometimes called as hardware as a service (HaaS) [1, 7], Platform as a service (PaaS) and Software as a service (SaaS). Cloud computing planning promotes the resource sharing in a pure plug and provides a model that dramatically simplifies its infrastructure. The major advantage of cloud computing includes ease of use and cost-e activeness in accessing the resources over the Internet. Employing the resources in the cloud provides greater expediency to the user because of its systematic manner. Cloud helps us to make use of the existing technologies such as virtualization, service orientation and grid computing in large scale distributed environment [4, 5]. To assure the cloud data integrity and availability, Efficient approaches that enable storage correctness assurance on behalf of cloud users have to be premeditated. Hence, cloud operations should also imperatively support the dynamic features that make the system design even more challenging.

As Cloud computing is a new emergent technology despite having many binomial factors, it faces many threats in various ways. It has spread very fast due to its edibility over ease of access as it eliminates the need for extra hard drives and memory space allocation. As the cloud is a distributed system, the data stored in it is widespread in distinct locations, and it is accessed anywhere. The distributed nature of the data creates the requirement for high security over outsourced data as there exists a probability that anyone can exploit the outsourced data. The hackers [1, 2, 16], can also access the outsourced data by hacking any server virtually, and the statistical results showed that one third of the breaches happened from stolen or lost laptops exposing the data unintentionally from the users or the employee of the organization over the Internet. Further, nearly 16 percent of this data exposure is due to the insider theft. The cloud security providers were even trying to provide a solution to security problems such as security, privacy, reliability, legal issues, open standards, compliance, freedom and long term viability.

Cloud aords three major types of deployment models, which comprises of Public, Private, and Hybrid Cloud. Most common level people and some organizations make use of the public cloud model in a majority for data storage purposes because it consumes less cost and correspondingly provides utmost security over the outsourced data, but there is also a probability of data leakage in a public cloud environment. The private cloud model [9, 13], depends upon a particular firm but found to be comparatively costlier than the public cloud. The combination of either private- public or public-public or private-private infrastructure forms the Hybrid cloud environment [12, 15], providing the combined advantage of both the private and the public cloud. The significant been of the use of the hybrid cloud involves improvised security with lesser management costs.

The possession of fine-grained data access control and storage correctness verification remains to be a mandatory feature in any system, which shares the data contents among multiple users with different level of trust. To ensure the property of cloud data security, highly trusted cloud users might be allowed with full access rights while the other users were assigned partial access rights over the outsourced data. Efficient management of the fine-grained access provision in a system with users having different access privileges remains to be a challenging issue in cloud computing. To provide better security features in cloud computing environment, a novel Storage Correctness and Fine grained Access Provision Scheme (SCFAP) is given. It comprises of two parts, where the rest part designates the access structures to the users and the second presents a storage correctness scheme through the use of the access structure defined at the preliminaries. A combination of

public key, private key, and access structures is assigned to all the users of the system that is derived from the appropriate user attributes. Through the distributed keys and access structures, every single user of the system establishes the secure cloud connection and performs accesses to the cloud data. For every successful cloud data upload, the user is provided with a token, which is used to verify and validate the storage correctness associated with the outsourced data thereby improving the storage efficiency.

The paper is organized in the following manner. The section next to introduction details the literature survey, the next part, deals with the summary of limitations followed by preliminary concepts and algorithms, system design, the proposed SCFAP scheme, case study, Implementation Details, Results, and Discussion. The paper is ended with the conclusion and future work.

2. Related Works

This section describes and analyzes other approaches towards facing the challenge of fine-grained access provision to cloud users. Multiple solutions are examined, after which an overview of their works was given. This section also describes the comparison of two major approaches that is related to the fine-grained access provision techniques.

2.1 Overview

This section presents an overview of the works, which is related to the proposed SCFAP scheme.

2.1.1 Cloud based Access Control Techniques

[24] presents a data access control scheme called DAC-MAC for the multi authority cloud storage system. It provides a multi authority CPABE scheme with efficient data decryption and user revocation functions. This work further oers an Extensive Data Access Control Scheme (EDAC-MACS) that provides secured user data access even at weaker security assumptions. The security analysis results of this scheme prove that this scheme is collusion resistance but lacks at the property of fine-grained access provision to the individual users of the system. In work done by [25, 10], integration of cryptographic techniques with RBAC techniques was made and it uses role keys for data decryption. Further this work presents a hybrid cloud architecture, where the public cloud contains the basic level details and most sensitive information over the private cloud. This work separates the property of user delegation to active and passive types and establishes effective role management through the use of delegation servers and protocols. The Cipher text Policy Attribute Based Encryption was given by; it realizes the complex access control mechanisms over the encrypted data [23, 14]. Here the attributes expressed solitarily the user credentials and the person who encrypts the data could x the access limit to the users for data decryption. Through the use of this scheme, the data stored could be kept confidential even though it resides on the untrusted server. The ID based cryptographic scheme [8], makes use of the user attributes such as user id for encryption and decryption process of the outsourced data. The development of ID based cryptographic scheme provides the secured data storage over the public cloud and improved client authorization for other users to access the data content.

2.1.2 Hierarchical Based Access Control Schemes

In HASBE [17, 21], the user access rights were provided by the hierarchical access structure framed for each user of the system. This scheme ensures the property of scalability through the extension of ASBE (Attribute Set Based Encryption) technique [6]. It defines a hierarchical structure that delegates the operation of trusted authority and private key generation to the domain authorities of the lower level. Here the user attributes were converted into the stable structure of the recursive type that permits the users to de ne constraints dynamically by representing a different combination of attributes, which satisfies the user access policy. That ensures the property of flexibility and fine-grained access control over HASBE systems. The concept of Hierarchical Based Access Structure is extended to form the Hierarchical Structure used in this paper.

2.1.3 Token Based Access Verification Systems

[20] proposed a flexible distributed storage integrity auditing mechanism that consists of homomorphic tokens and erasure coded data. Tokens are provided to the users from randomly chosen block indices from each data vector space analogous to the memory location of the user requested le in the cloud. The use of erasure coded data technique protects the user data and eliminates the system errors such as data redundancy, fault tolerance and server crashes. In Privacy Preserving Public Auditing for Secure Cloud Storage by [18, 11] comprises a third party auditor (TPA) for auditing the integrity of outsourced data; this eradicates the new threats and realizes the data privacy. This scheme uses random masking technique integrated with a homomorphic authenticator that ensures the privacy of public auditing. Flexible distributed storage integrity checking mechanism is proposed by [3] using homomorphic tokens and it avoids security problems like identifying unknown users. Through the use of homomorphic tokens and distributed erasure coded data, users were permitted to audit the outsourced data. This auditing allows the users to identify both the improper data access and cloud server misbehaviors. This scheme even ensures the cloud data security, which allows the users to perform dynamic operations efficiently over the outsourced data. Experimental analysis of their proposed scheme proves that it provides high efficiency against Byzantine failure, unknown user attacks and attacks on cloud data modification. Access control schemes based on the token system were developed to provide greater security over the cloud storage systems.

2.2 Comparison of Related Works

This section presents a brief summary about two major approaches relating to the proposed SCFAP scheme. The HASBE scheme given by [17], and the flexible integrity auditing mechanism provided by Wang Cong et al, were taken into comparison, and it is described as follows:

2.2.1 Work by Wan Zhiguo et al.

To ensure the property of scalability and flexibility over outsourced data, a solution is presented in work done by [17]. This work shows a Hierarchical Attribute Set Based Encryption (HASBE) scheme to cloud users, which extends the property of Ciphertext attribute set based encryption technique. This scheme not only aims in the achievement of scalability, it even inherits the property of flexibility and fine-grained access provision through the management of compound attributes. The HASBE scheme makes use of multiple value access expiration time to deal with user revocation problems. The rest part of this work describes the extension of HASBE from ASBE technique using the hierarchical structure. Whereas the second part provides a clear demonstration of the implementation of access control scheme based on HASBE for cloud computing.

The cloud computing system considered in this work consists of five major entities. The cloud service provider provides services to users. The data owners share their data contents through the cloud in an encrypted manner. Data consumers decrypt the shared contents to perform their respective access operations. Each data owner and data consumer was assigned with a domain authority, where each domain authorities could be managed through parent domain authorities or trusted domain authorities. The major responsibility of every domain authority is to administer the domain authorities at next level or the data owner or consumer in its domain. In HASBE scheme the data users were only assumed to possess read access. All the entities associated with this scheme were organized in a hierarchical manner to accomplish their tasks.

A recursive set based key structure is formed for every user, where each element of the set is either a set or an element corresponding to a user attribute. The depth of the key structure is found using the level of recursions in the recursive set, which is similar to the definition of depth tree. For a key structure of depth 2, members of the set can either be sets or attribute elements at depth1. At depth 2 it is mandatory that all the members of the set should be of attribute elements. A unique label for the user attributes was formed using key structure. The access structure to the users in HASBE was formed in a similar way to the ASBE scheme given by [3]. In access tree structures the leaf nodes were considered to be the attributes, and non-leaf nodes represent the threshold gates. The non-leaf nodes were defined using its children and threshold values.

This work provides user access provision with the help of the hierarchical access structure, and it is formed using appropriate user key structure and access structures. It means that the user with private key corresponding to attributes in key structure would be able to access the data, only when their attributes satisfies the access policies defined by the access structure. System Setup, Top Level Domain Authority Grant, New Domain Authority/User Grant, New File Creation, User Revocation, File Access, and File Deletion are the seven major operations associated with Wan Zhiguo et al HASBE scheme. Each major system operations related to the HASBE scheme invokes the appropriate algorithms associated with it to accomplish their tasks, and it works by bilinear mapping concepts. Through the use of this operations, every user of the system shares and uses their data contents using HASBE scheme. Though Wan Zhiguo et al system provides a better solution to scalability and flexibility issues, the complete support for compound values and multiple value assignments are measured and found to be lagging in efficiency. Which reduces the level of fine-grained data access. The proposed SCFAP scheme defined users with their role based classification. Provides efficient support for compound attributes and multiple value assignments. The hierarchical structure described in SCFAP scheme improves

the level of fine-grained access provision associated with individual users of the system. The HASBE scheme further does not allow write access to the data users of the system. This makes its application inappropriate to critical systems like financial sectors, where several users require write operations to be performed. SCFAP scheme allows the users to perform write operations in an effective manner, and it is achieved through the use of token granting system, which preserves the storage correctness of the outsourced data.

2.2.2 Work by Wang Cong et al.

An approach to form solution for security risks accompanying the correctness of physical possession over outsourced data were done by [17]. This work presents a flexible distributed storage integrity auditing mechanism, which ensures the correctness of outsourced data through the use of homomorphic token and distributed erasure coded data. This scheme provides efficient user auditing of cloud data with very lightweight communication and computation cost. The auditing result provides both storage correctness guarantee as well as fast data error localization (identification of server misbehaviors). It even allows user access operations over outsourced data including block deletion, modification and appends functionalities. The overall contributions of this work is summarized as follows:

1) In comparison to many of its predecessors, this scheme achieves both the storage correctness insurance and data error localizations.

2) This scheme further supports secure and dynamic operation over data blocks including update, delete and append.

3) The work further makes an extensive security analysis that shows its resistance towards Byzantine failures and malicious data modification attack and server colluding attacks.

The flexible integrity auditing mechanism discussed in this section consists of four major entities, which includes User, Cloud Service Provider (CSP), Cloud Server (CS) and Third Party Auditor (TPA). Users share their data through cloud storage services, and a user can be either enterprise or an individual customer. Cloud Server (CS) is managed by the CSP to provide better computation and storage facilities to the users of the system. TPA is an optional entity with expertise qualities that user does not possess. TPA assesses and describes the risk of cloud storage services on behalf of users upon request.

This work provides more focus towards file oriented data rather than non-file oriented applications like social networking systems. Block level operations over user data were considered as block update, block delete, block insert and append operations. The major focus of this work is to identify the key integrity issues like unauthorized data modifications and corruptions, caused due to server compromises and random Byzantine failures.

Users store their valid credential to cloud servers through CSPs. The problem of data redundancy could be employed through the technique of erasure correcting code. This scheme further tolerates faults and server crashes that happen due to increasing data users. The users interact with cloud servers for processing le retrieval request through CSPs. As it is not feasible for the users to possess their data locally, it is necessary to verify the correctness and maintenance of the cloud data.

The users were provided with the pre computed tokens that provide correctness assurance to the users of the system. Tokens are derived from the subset of file blocks in a random manner. The verification token helps the users to ensure correctness of data operation request processed by the CSP. Tokens are issued to the user based on randomly chosen block indices from each data vector space corresponding to memory position of the requested le in the cloud and erasure coded data. In cases of in appropriate situations like insufficient resources and time the users can delegate their responsibilities to TPA. The system is designed in such a way that leakages of user? Outsourced data towards auditing protocol were prohibited. This work achieves secure data storage through five major steps, which includes le distribution preparation, challenge token pre computation, correctness verification and error localization, le retrieving and auditing and finally, towards third party auditing. The algorithms associated with each stage helps in the management of activities accompanying data storage management and correctness verification processes. This scheme provides an approach methodology that prevents CSP to process data dynamics without knowing user secret key materials and ensures users that dynamic data operation request done by CSP were processed faithfully. In this manner the property of integrity assurance and storage correctness where done in [17] scheme.

Tokens were provided to the users, based upon randomly generated block indexes and memory position of the le. This makes the property of storage correctness associated with integrity auditing scheme to be a probabilistic feature. The proposed SCFAP scheme solves this issue by granting tokens to the users in a deterministic manner. In SCFAP scheme tokens were derived from Metadata containing le locations and distributed to all the users of the system during appropriate phases. Here the major advantage is that as a result of the write operation done by the authorized system an updated token would be provided to all the users of the system, through which the property of storage correctness is achieved.

3. Construction of Storage Correctness and Fine-grained Access Provision Technique

3.1 System Design

This section presents a conceptual design of the novel scheme called Storage Correctness and Fine-grained Access Provision (SCFAP) scheme, which is described in Figure 1. The proposed SCFAP scheme consists of two parts. The rest part deals with the construction of hierarchical based user access structures and the second part depicts the algorithmic phases associated with SC-FAP scheme that helps in the achievement of fine-grained data access and improved storage efficiency across the outsourced cloud data storage. A set of appropriate cryptographic keys and access structures derived from the exact user attributes were distributed to all the users of the system. Through the use of the access structures and cryptographic keys every user of the system performs the cloud data access in a secure way. As a result of the encryption process, both the data owners and users were provided with a token, which assists in the process of integrity and security verification over the outsourced data.

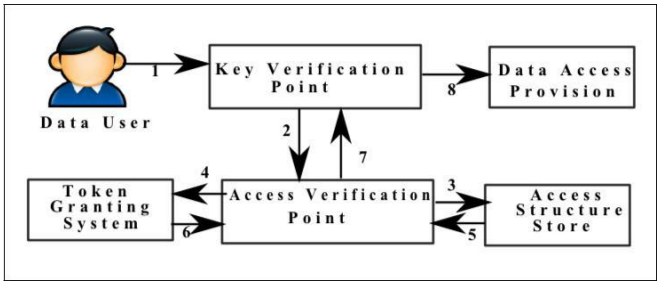

Figure 1: System design of SCFAP

The proposed system consists of five major entities and the description to the entities were given as follows, Attribute Authority (AA): The major responsibility of the Attribute Authority (AA) is to manage all the attribute related activities in specialization with the activities conning to the

management of user roles. This includes maintenance of role revocation, delegation, key allocation to users and authentication of the user given credentials like the public key, private key, etc. Cloud server (CS):

Cloud server performs all the computation related activities. This includes the computation of user given inputs and producing corresponding computational results and acknowledgments to the users. Cloud Service Provider (CSP): The Cloud Service Provider (CSP) provides services to the users of the system and performs the validation of the user given inputs and outputs during the process of encryption and decryption. Service Consumers (Users): A Service consumer is also called as the user of the system, consumes the services provided by the cloud computing environment. Data Owner: Shares valid data contents over cloud computing environment and exes data access limits across data users.

3.2 Assumptions

This work assumes an existing data access control model to build upon, and the proposed design makes use of the access control properties defined previously at related works. The hierarchical structure described in this paper is assumed to provide many to many data sharing in a secured manner through which the property of fine-grained access control, confidentiality, and non-repudiation of the outsourced data was achieved.

3.3 Key Terminologies

3.3.1 Access Assignment Structure

A summary on Access assignment structure is depicted in Figure 2.

3.3.2 Hierarchical Structure

The hierarchical structure defines the access policy associated with the individual users of the system. A hierarchy is framed from the combination of the user unique and common attributes. Each hierarchy represents the one to one relationship between the user and their access policies. The access policy defines the set of operations (read or write access) the user could perform over the outsourced data.

3.3.3 Key Structure

Key structures were designed to preserve the security of the outsourced data. Key structures are derivatives from the user common attributes like roles. The formation of key structure assigns the access privileges to the set of the common users over the outsourced data. This states that users beneath a particular role were assigned with a key structure such that they could gain access to a particular set of les.

3.3.4 Access Structure

Access structures were designed to achieve the property of fine-grained user access and it is derived from the user unique attributes like user id. It defines the extent to which an individual user could access the data.

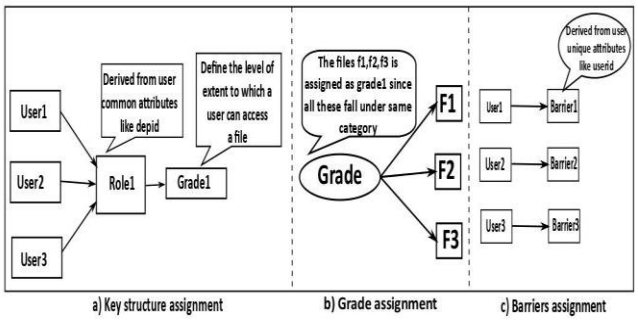

Figure 2: Access assignment structure

3.3.5 Grade

Grade denotes the level of the extent to which the set of common users could gain access to a particular set of les. Each grade formally represents a key structure, such that a user with certain grade could gain access to all the les that comes with the scope of a particular grade. Grades were derived from the user common attributes like dep id, such that it represents a set of les that belongs to the particular department. An example of the measure of a grade is described in Figure 3.

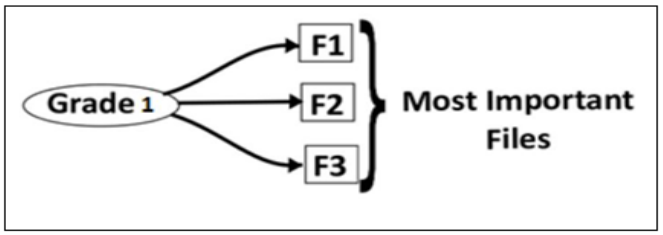

Figure 3: Access limits associated with a grade

3.3.6 Barriers

Barriers are restrictions imposed over the grades to achieve the fine-grained user access level. It has been found that it is not necessary for a user with a particular grade to access all the files that come under a particular grade. To solve this issue, barriers were designed and imposed over the user grades. From Figure 4. It is understood that though the user belongs to grade1 which provides access rights to three les F1, F2, F3. The imposition of the barrier B1 over the particular user grade G denies the user le access request to the le F2; such that the user could only be able to access the les F1 and F2. It provides the appropriate access rights to the users through which the property of fine-grained access provision is achieved.

3.3.7 Tokens

Tokens were derived from the metadata containing the le location and it acts as a user authentication entity. Tokens were issued to the data users as a result of the data encryption process. Through the use of the tokens, the user could easily verify the existence of their corresponding files in a convenient manner. Since the token represents the Metadata about the file location, it assists in the process of easier le retrieval. This improves the storage efficiency of the proposed system. Further, clear descriptions about the working of the tokens were described in phase 6, 7 and 8 of the SCFAP scheme. A new method of mathematical modelling was used to identify the functions and variables of the proposed scheme.

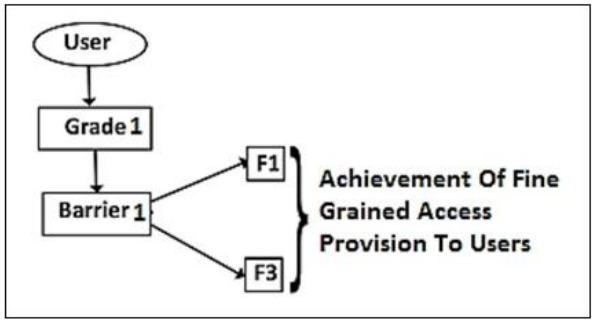

Figure 4: Barrier generations

4. Preliminary Concepts and Algorithms

4.1 KeyGen()

KeyGen() is a basic algorithm for key generation through which all the other keys associated with the data users were derived. This algorithm is invoked automatically whenever the process of key generation is required. Let us consider the set of keys k such that it contains a set of integers up to n. Such that Kn = fk1; k2; ; kng.

$$kn : \sum_{z=0}^{n} Kz \qquad \sum_{m=0}^{n} dm:$$

Where d is the set of derived keys. It could be of any other key like master key, public key and private key. In a simpler way the KeyGen algorithm generates random number of keys in accordance to the user given input parameters.

4.2 Formation of Hierarchical Access Structure

The proposed SCFAP scheme makes use of the hierarchical access structure to de ne the user access rights. The basic concept behind the hierarchical access structure was described in the previous section of the paper. In SCFAP scheme each user is assigned with a hierarchical structure, which is derived from their respective key and access structures.

Key structure is premeditated to preserve the security of the outsourced data and it represents the access rights to the group of users with a common identity. The basic concepts behind the formation of the key structure were given in the previous section. It is formed using the user common attributes like dep id. In an organization the most important or most secured les could be accessed only by the personals at the top most designation order, least important les by the low level personals and ordinary les could be accessed by the mid level individuals. In correspondence to the user designation order grades for a group of members with common identity (users under a particular role) is calculated from grade1. grade n". For every user with a role, grades were allocated with respect to their access privilege that defines the level of extent to which the users could access the data.

4.2.1 Access Structure

The access structure represents the access rights to the individual user of the system. Even though a particular user is assigned with a grade representing the key structure, it is not mandatory that the user could access all the les that come under a particular grade. The access structures associated with the SCFAP scheme were designed in such a way that solves the problem of above mentioned issue. The access structure was framed from the user barrier limits, which are derived from the user unique attributes like user id. Barriers are restrictions that were imposed over the user access grades to achieve the fine-grained access control. The assignment of the access structure defines the individual access limits over the set of les. In addition to this, phase 3 of the storage correctness scheme provides a brief summary about the algorithmic implementation of the user access structure assignment.

Through the use of the key and access structure discussed above, a hierarchical access structure is formed in the proposed SCFAP scheme, and it is illustrated in Figure 2.

4.2.2 Token Granting System

The proposed SCFAP scheme makes use of the token granting system through which the property of storage correctness is achieved. As it is described at the previous section tokens were derived from the Meta data containing the le location that assist in both ways, through which the process of storage correctness as well as the easier retrieval of the outsourced les could be made. The prime idea behind the use of token granting system in SCFAP scheme is that at the end of every successful data encryption process the data users were provided with the tokens, through which the data users verifies the existence of the outsourced data. The users could also be able to perform the decryption process only when the Meta data of the user given token points to the user requested le.

4.3 SCFAP Phases

The storage correctness phases and fine-grained access provision scheme consists of nine phases through which the property of fine-grained access provision and storage correctness verification is achieved. The SCFAP phases apply the concept of hierarchical access structure and token granting system described in preliminaries part.

4.3.1 Phase 1: SetUp()

It takes the user security parameter as an input and generates master key mk as an output. This step is done by the cloud server through automatically invoking the KeyGen algorithm.

K : mk = ./ kn: (1)

Equation 1 joins the user security parameter with the unique key generated by KeyGen() algorithm and distributes the master key to the corresponding users of the system.

4.3.2 Phase 2: GradeGen(mk; Rid)

This phase is performed by the Attribute Authority and it takes the master key mk and Role id Rid as an input, produces public key pk and grade g as an output. Public key is derived from the master key mk by manually invoking the KeyGen() algorithm. Let us consider two sets,

R = fR1; R2; R3; ; Rng and G = fg1; g2; g3; ; gng be the set of roles and grades. Such that R G (means that the role R is isomorphic to grade G).

Z : 8Rid 2 RjRid R:

Any Rid that belongs to R is the subset of R.

Z : 8Rid : P (Rid)jRid < R: (2)

At least for one value of Rid the value of Rid in R is true.

Such that Rid is covered by r where r 2 R.

) Z : 9Rid ! rjr , G:

There exists G and Rid that implies a role such that the role corresponds to a grade G.

4.3.3 Phase 3: BarrierGen(Uid; rk; pk)

It takes (Uid; rk; pk) use rid, role key, and public key as an input and as a result of computation the barrier limit bl and the private key prk is returned to the users. The private key is manually generated by role admin through the invocation of KeyGen() algorithm.

Let U be the universal set that contains all the users of the system and can be written as U = fU1; U2; ; Ung and B be the set of barriers such that can be written as B = fb1; b2; ; bng.

Z : 8Uid 2 Uid U:

8Uid : P (Uid)jUid U:

Similarly from Equation 2 this step is derived.

n n n

X kX X

Uk9Bj Uk 2 Bj Uk B:

k=1 =1 k=1

Means that for all the users there exists a barrier limit such that all the users in U belong to barriers in B. So that U is the subset of B. Such that there exists an Uid 2 B. where

a a

Z : b = bu : Ru ! G:

u2U u2U

Since, all the users U is the subset of B there exists a b corresponding to the user u, where the barriers can be calculated as a coproduct of user and barrier sets.

4.3.4 Phase 4: Encrypt(f; rk; pk)

This phase is done by the CSP and it takes the le f, role key rk and Public key pk as an input and the outputs cipher text cp to the users of the system. Data encryption is done as a part of le upload.

Z : f rk pk , f(rk;Pk)

Z : f rk pk , c:tjc:t f(rk;Pk):

Encryption is done as a combination of input parameters.

4.3.5 Phase 5: TokGen(f; rk; pk)

It takes le f, role key rk and Public key pk as an input. It is the most important part of the encryption process and it is done during the process of le upload. Since tokens were derived from the data containing le locations, here we use the concept of reduction to reduce a le to tokens. Let F = ff1; f2; ; fng be the set of les such that by property of reduction

if

9f 2 db 8R 2 F , f(R) 2 ti

then

F db ti

F db t Denotes that a le set F can be reduced to token ti and it is achieved through the data blocks. At the end of this phase tokens generated were distributed to the data users to verify the correctness of the outsourced data.

4.3.6 Phase 6: Token Computation

It is done by the CSP and cloud server as a part of the data decryption process. Let F = ff1; f2; ; fng be the set of les and

Ti = fti1; ti2; ; ting be the set of tokens associated with the les. Then,

a

[ti1; ti2; ; tin](Fi) = fdb[ti1; ti2; ; tin]; db 2 F g:

Where, Fi = f1; 2; ; ng (Only the tokens between [ti1; ti2; ; tin] can access the les in data blocks). Tokens out of this scope would be computed as corrupted and cannot be accessed. It is based on the projection property. Where,

db[ti1; ti2; ; tin] = f(ti; v) 2 d; t 2 [ti1; ti2; ; tin]g:

Means the remaining set of data blocks corresponds to some other tokens. The result of proportion [t ; t 2; ; tin](F) can be found only if [ti1; ti2; ; tin`] isi1 I (F). t means a le would be accessed only when token matches with it.

4.3.7 Phase 7: Token Update

Whenever the data user performs the write operation the tokens associated with the users were updated and distribute to all the associated system entities. This is due to the reason that the process of write operation may extend or delete some part of the le that leads to the change of the Meta data containing the le location. The process of token update is described as follows:

newti = ti ./ wc:

Where wc is the newly written content.

4.3.8 Step 8: Token Correctness

It is done as a part of data decryption during the process of le download. It takes (ti; cp) as an input. Let ti; cp be an algebraic function over F then Z : ti 2 Ti; cp 2 Cp; let us take an element ti 2 f. Such that,

Z : (ti; cp1) + (ti; cp1) (ti1 + ti2 + cp)

Z : (ti; cp1) + (ti; cp1) (ti1; cp1 + cp2)

Z : f(ti; cp) (f ti; cp) (ti; f cp) , ti cp:

It matches the values in the token and cipher text and returns the mismatch thus the token correctness is verified.

4.3.9 Phase 9: Decryption (cp; rk; bl; prk; ti)

Data decryption is done as a part of the le download process. It takes cipher text, role key, barrier limits, private key and token as input and returns the plain text to the users based on their respective access structures.

a

Z : cp ./ ti =b1(cp ./ ti)

a a

Z : cp ./ ti =b1(cp ./ ti)j(cp ./ ti) = b1(cp ./ ti) , Pt:

It combines the cipher text and token depending upon the user barrier levels and provides the plain text.

Table 1: Summary of SCFAP phases

Phase No	Phase Name	Input	Output	Doneby
1	SetUP( )		M_k	CS
2	GradeGen( )	M_k; K_{id}	P_k; G	AA
3	BarrierGen( )	U_{id}; R_k; P_k	B; P_{rk}	RA
4	Encrypt( )	F; R_k; P_k	C_t	CSP
5	TokGen( )	F; R_k; P_k	T_i	CS,CSP
6	TokenComp( )	F; T_i	C_t	CS,CSP
7	TokenUpdate( )	T_i	newT_i	CS,CSP
8	TokenCorrectness( )	T_i; C_t	File Validity	CS,CSP
9	Decrypt( )	T_i; C_t; R_k; B; P_{rk}	Plaintext	CSP

3. CONCLUSIONS

The paper defines an SCFAP scheme that solves the problem of fine-grained access provision and storage correctness associated with the existing access control techniques. The rest part of the SCFAP scheme involves the formation of hierarchical structures that fixes the appropriate access policies to the users; this improves the fine-grained ness associated with the access policy.

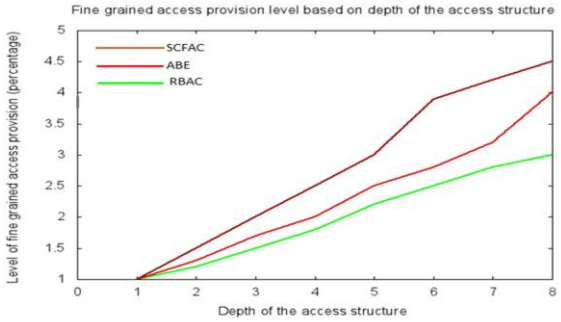

Figure 8: Comparison to fine-grained access provision measure in SCFAP scheme

The next part deals with the achievement of storage correctness related to the les, and it is made through the usage of the token granting system. In addition to this, the use of token granting system improves the storage efficiency, security, and performance of the proposed system. As this paper explains only on the key structure and Access Structure associated with the plain text but not about the Cipher Text Access Structure. In future, this work could Figure 8: Comparison to fine-grained access provision measure in SCFAP scheme be extended for outsourced data decryption techniques.

REFERENCES

[1] P. Arora, R. C. Wadhawan, and E. S. P. Ahuja, \Cloud computing security issues in infrastructure as a service," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, no. 1, pp. 1{7, 2012.

[2] V. Bhangotra and A. Puri, \Enhancing cloud security by using hybrid encryption scheme," International Journal of Advanced Engineering Technology, vol. 6, no. 4, pp. 34{40, 2015.

[3] R. Bobba, H. Khurana, and M. Prabhakaran, \Attribute sets: A practically motivated enhancement to attribute based encryption," in Computer Security (ESORICS'09), pp. 587{604, Springer, 2009.

[4] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, \Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer Systems, vol. 25, no. 6, pp. 599{616, 2009.

[5] Z. Cao, C. Mao, L. Liu, \Analysis of one secure anti collusion data sharing scheme for dynamic groups in the cloud," International Journal of Electronics and Information Engineering, vol. 5, no. 2, pp. 68{72, 2016.

[6] V. Goyal, O. Pandey, A. Sahai, and B. Waters, \Attribute based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89{98, 2006.

[7] W. Huang, A. Ganjali, B. H. Kim, S. Oh, and D. Lie, \The state of public infrastructure as a service cloud security," ACM Computing Surveys, vol. 47, no. 4, pp. 68, 2015.

[8] N. Kaaniche, A. Boudguiga, and M. Laurent, \ID based cryptography for secure cloud data storage," in 2013 IEEE Sixth International Conference on Cloud Computing, 2013.

[9] R. Ko and R. Choo, The Cloud Security Ecosystem: Technical, Legal, Business and Management Issues, Syngress, 2015.

[10] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, \Identity based encryption with outsourced revocation in cloud computing," IEEE Transactions on Computers, vol. 64, no. 2, pp. 425{437, 2015.

[11] J. Liu, X. Huang, and J. K. Liu, \Secure sharing of personal health records in cloud computing: ciphertext policy attribute based signcryption," Future Generation Computer Systems, vol. 52, pp. 67{ 76, 2015.

[12] O. Mazhelis and P. Tyrv•ainen, \Role of data communications in hybrid cloud costs," in 2011 37th IEEE EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 138{145, 2011.

[13] S. Ramgovind, M. M. Elo , and E. Smith, \The management of security in cloud computing," in Proceeding of IEEE Information Security for South Africa (ISSA'10), pp. 1{7, 2010.

[14] B. D. Revathy, M. P. Ravishankar, and C. I. T. Ponnampet, \Enabling secure and e cient keyword ranked search over encrypted data in the cloud," 2015.

[15] P. Samarati and S. De C. di Vimercati, Cloud security: Issues and concerns, Wiley, New York, 2016.

[16] J. Singh, \Cyber-attacks in cloud computing: A case study," International Journal of Electronics and Information Engineering, vol. 1, no. 2, pp. 78{87, 2014.

[17] Z. Wan, J. Liu, and R. H. Deng, \Hasbe: A hierarchical attribute based solution for flexible and scalable access control in cloud computing," IEEE Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 743{754, 2012.

[18] C. Wang, S. M. Chow, Q. Wang, K. Ren, and W. Lou, \Privacy preserving public auditing for secure cloud storage," IEEE Transactions on Computers, vol. 62, no. 2, pp. 362{375, 2013.

[19] C. Wang, K. Ren, W. Lou, and J. Li, \Toward publicly auditable secure cloud data storage services," IEEE Network, vol. 24, no. 4, pp. 19{24, 2010.

[20] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, \Toward secure and dependable storage services in cloud computing," IEEE Transactions on Services Computing, vol. 5, no. 2, pp. 220{232, 2012.

[21] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, \An efficient le hierarchy attribute based encryption scheme in cloud computing," IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, pp. 1265{1277, 2016.

[22] H. Witti, C. Ghedira, E. Disson, and K. Boukadi, \Security governance in multi cloud environment: A systematic mapping study," in 12th World Congress on Services (SERVICES'16), 2016.

[23] Y. Wu, Z. Wei, and R. Deng, \Attribute based access to scalable media in cloud assisted content sharing networks," IEEE Transactions on Multimedia, vol. 15, no. 4, pp. 778{788, 2013.

[24] K. Yang and X. Jia, \Dac macs: Effective data access control for multi authority cloud storage systems," in Security for Cloud Storage Systems, pp. 59{83, Springer, 2014.

[25] L. Zhou, V. Varadharajan, and M. Hitchens, \Achieving secure role based access control on encrypted data in cloud storage," IEEE Transactions on Information Forensics and Security, vol. 8, no. 12, pp. 1947{1960, 2013.