



## Improving Flexibility, Scalability and Fine-Grained Access Control using Hierarchical Attribute Set Based Encryption (HASBE) and Security in Cloud Computing

Prashant A. Kadam\*, Dinesh M. Yadav

Department of Computer Engineering &  
Savitribai Phule Pune University,  
Maharashtra, India

**Abstract**— Cloud Computing is going to be very popular technology in IT enterprises. For any enterprise the data stored is very huge and invaluable. Since all tasks are performed through network it has become vital to have the secured use of legitimate data. In cloud computing the most important matter of concern are data security and privacy along with flexibility, scalability and fine grained access control of data are the other requirements to be maintained by cloud systems. Access control is one of the prominent research topics and hence various schemes have been proposed and implemented. But most of them do not provide flexibility, scalability and fine grained access control of the data on the cloud. In order to address the issues of flexibility, scalability and fine grained access control of remotely stored data on cloud the proposed scheme uses hierarchical attribute set-based encryption (HASBE) which is the extension of attribute- set-based encryption (ASBE) with a hierarchical structure of users. The proposed scheme achieves scalability by handling the authority to appropriate entity in the hierarchical structure, inherits flexibility by allowing easy transfer and access to the data in case of location switch. It provides fine grained access control of data by showing only the requested and authorized details to the user thus improving the performance of the system. In addition, it provides efficient user revocation within expiration time, request to view extra-attributes and privacy in the intra-level hierarchy. Thus the scheme is implemented to show that it is efficient in access control of data as well as security of data stored on cloud with comprehensive experiments performed.

**Keywords**— Fine-grained access control; attribute-set-based encryption(ASBE); hierarchical attribute-set-based encryption(HASBE);user revocation ; intra-level hierarchy

### I. INTRODUCTION

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services. The services themselves have long been referred to as Software as a Service (SaaS). The datacenter hardware and software is what we will call a Cloud.

Cloud computing is a web-based application that provides computation, software, infrastructure, platform, devices and other resources to users on the basis of pay as you use. Clients can use cloud services without any installation and the data uploaded on cloud is accessible from anywhere in the world, the only requirement is the computer with active internet connection. As a customizable computing resources and a huge amount of storage space are provided by internet based online services, the shift to online storage has contributed greatly in eliminating the overhead of local machines in storage and maintenance of data. Cloud provides a number of benefits like flexibility, disaster management and recovery, pay-per-use and easy to access and use model which contribute to the reason of switching to cloud. Cloud gives the provision for storage of important data of users. Thus cloud helps to free up the space on the local disk.

The prominent security concern is data storage security and privacy in cloud computing due to its Internet-based data storage and management. The data security issue becomes vital when the data is a confidential data. In cloud computing, users have to give up their data to the cloud service provider for storage and business operations, while the cloud service provider is usually a commercial enterprise which cannot be totally trusted. So the data integrity and privacy of data is at risk [12].

Flexible and fine-grained access control is strongly desired in the service-oriented cloud computing model. Various schemes which provide access control models have been proposed. But the problem related with these schemes is that they are limited to data owners and service providers which exist in the same trusted domain [12].

Features offered by Cloud Computing: 1. Scalability and on-demand services. Cloud computing provides resources and services for users on demand. The resources are scalable over several data centers. 2. User-centric interface Cloud interfaces are location independent and can be accessed by well-established interfaces such as Web services and Internet browsers. 3. Guaranteed Quality of Service (QoS) Cloud computing can guarantee QoS for users in terms of hardware/CPU performance, bandwidth, and memory capacity. 4. Autonomous system: The cloud computing systems are autonomous systems managed transparently to users. However, software and data inside clouds can be automatically reconfigured and consolidated to a simple platform depending on user's needs. 5. Pricing Cloud computing does not require up-front investment. No capital expenditure

## II. RELATED WORK

This section contains the comparative study and overall analysis of the existing systems. Literature Survey of these systems is as follows:

### A. Literature Survey

[1] Vipul et al. published "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data." which states that as more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encrypting data is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). A new cryptosystem is developed for fine-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE). In the cryptosystem, ciphertexts are labeled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. It demonstrates the applicability of the construction to sharing of audit-log information and broadcast encryption. The construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE).

[2] Rakesh et al. published "Attribute-Sets: A Practically Motivated Enhancement to Attribute-Based Encryption." elaborates that in distributed systems users need to share sensitive objects with others base on the recipients ability to satisfy a policy. Attribute- Based Encryption (ABE) is a new paradigm where such policies are specified and cryptographically enforced in the encryption algorithm itself. Ciphertext-Policy ABE (CP- ABE) is a form of ABE where policies are associated with encrypted data and attributes are associated with keys. In this work focus is on improving the flexibility of representing user attributes with keys. Specifically, it proposes the Ciphertext Policy Attribute Set Based Encryption (CP-ASBE) - a new form of CP-ABE - which, unlike existing CP- ABE schemes that represent user attributes as a monolithic set in keys, organizes user attributes into a recursive set based structure and allows users to impose dynamic constraints on how those attributes may be combined to satisfy a policy. It shows that the proposed scheme is more versatile and supports many practical scenarios more naturally and efficiently. It provides a prototype implementation of the scheme and evaluates its performance overhead.

[3] Pankaj et al. published "Cloud Computing Security Issues in Infrastructure as a Service." explains that cloud computing promises to cut operational and capital costs and the more important thing is it lets IT departments focus on strategic projects instead of keeping datacenters running. It is much more than simple internet. It is a construct that allows user to access applications that actually reside at location other than users own computer or other Internet-connected devices. There are numerous benefits of this construct. For instance other company hosts user application. This implies that they handle cost of servers, they manage software updates and depending on the contract user pays less i.e. for the service only. Confidentiality, Integrity, Availability, Authenticity, and Privacy are essential concerns for both Cloud providers and consumers as well. Infrastructure as a Service (IaaS) serves as the foundation layer for the other delivery models, and a lack of security in this layer will certainly affect the other delivery models, i.e., PaaS, and SaaS that are built upon IaaS layer. It presents an elaborated study of IaaS components security and determines vulnerabilities and countermeasures. Service Level Agreement should be considered very much importance.

[4] John et al. published "Ciphertext-Policy Attribute-Based Encryption (CP-ABE)." explains that in several distributed systems a user should only be able to access data if a user possesses a certain set of credentials or attributes. Currently, the only method for enforcing such policies is to employ a trusted server to store the data and mediate access control. However, if any server storing the data is compromised, then the confidentiality of the data will be compromised. In this paper we present a system for realizing complex access control on encrypted data that we call Ciphertext-Policy Attribute-Based Encryption. By using this technique encrypted data can be kept confidential even if the storage server is untrusted; moreover, the methods are secure against collusion attacks. Previous Attribute-Based Encryption systems used attributes to describe the encrypted data and built policies into users keys; while in this system attributes are used to describe the users credentials, and a party encrypting data determines a policy for who can decrypt. Thus, these methods are conceptually closer to traditional access control methods such as Role-Based Access Control (RBAC). In addition, it provides an implementation of system and gives performance measurements.

[5] Suhair et al. Published "Designing a Secure Cloud-Based EHR System using Ciphertext-Policy Attribute-Based Encryption. "which shows that as more and more healthcare organizations adopt electronic health records (EHRs), the case for cloud data storage becomes compelling for deploying EHR systems; not only is it inexpensive but it also provides the flexible, wide-area mobile access increasingly needed in the modern world. However, before cloud-based EHR systems can become a reality, issues of data security, patient privacy, and overall performance must be addressed. As standard encryption (including symmetric key and public-key) techniques for EHR encryption/decryption caused increased access control and performance overhead, the scheme proposes the use of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) to encrypt EHRs based on healthcare providers' attributes or credentials; to decrypt EHRs, they must possess the set of attributes needed for proper access. It motivates and presents the design and usage of a cloud-based EHR system based on CP-ABE, along with preliminary experiments and analysis to investigate the flexibility and scalability of the proposed approach.

[6] Ayad et al. published "Enabling Data Dynamic and Indirect Mutual Trust for Cloud Computing Storage System." which proposes a cloud-based storage scheme that allows the data owner to benefit from the facilities offered by the CSP and enables indirect mutual trust between them. The proposed scheme has four important features: (i) it allows the owner to outsource sensitive data to a CSP, and perform full block-level dynamic operations on the outsourced data, i.e., block modification, insertion, deletion and append, (ii) it ensures that authorized users (i.e., those who have the right to access the owners file) receive the latest version of the outsourced data, (iii) it enables indirect mutual trust between the owner

and the CSP, and (iv) it allows the owner to grant or revoke access to the outsourced data. The security issues of the proposed scheme are discussed. Besides, it justifies its performance through theoretical analysis and experimental evaluation of storage, communication, and computation overheads.

[7] Chandana et al. published “GASBE: A Graded Attribute-Based Solution for Access Control in Cloud Computing.” which states that cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. As promising as it is, this paradigm also brings forth many new challenges for data security and access control when users outsource sensitive data for sharing on cloud servers, which are not within the same trusted domain as data owners. To keep sensitive user data confidential against un-trusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users. However, in doing so, these solutions inevitably introduce a heavy computation overhead on the data owner for key distribution and data management when fine-grained data access control is desired, and thus do not scale well. The problem of simultaneously achieving fine-graininess, scalability, and data confidentiality of access control still remains unresolved. It addresses this challenging open issue by defining and enforcing access policies based on data attributes on one hand and allowing the data owner to delegate most of the computation tasks involved in fine-grained data access control to untrusted cloud servers without disclosing the underlying data contents, on the other hand. It achieves this goal by exploiting and uniquely combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption.

[8] Guojun et al. published “Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers.” which explains that with rapid development of cloud computing, more and more enterprises will outsource their sensitive data for sharing in a cloud. To keep the shared data confidential against untrusted cloud service providers (CSPs), a natural way is to store only the encrypted data in a cloud. The key problems of this approach include establishing access control for the encrypted data and revoking the access rights from users when they are no longer authorized to access the encrypted data. It aims to solve both problems. First, it proposes a hierarchical attribute-based encryption scheme (HABE) by combining a hierarchical identity-based encryption (HIBE) system and a ciphertext-policy attribute-based encryption (CP-ABE) system, so as to provide not only fine-grained access control, but also full delegation and high performance. Then, it proposes a scalable revocation scheme by applying proxy re-encryption (PRE) and lazy re-encryption (LRE) to the HABE scheme, so as to efficiently revoke access rights from users.

[9] Qin et al. published “Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services.” which states that cloud computing, as an emerging computing paradigm, enables users to remotely store their data into a cloud so as to enjoy scalable services on-demand. Especially for small and medium-sized enterprises with limited budgets, they can achieve cost savings and productivity enhancements by using cloud-based services to manage projects, to make collaborations, and the like. However, allowing cloud service providers (CSPs), which are not in the same trusted domains as enterprise users, to take care of confidential data, may raise potential security and privacy issues. To keep the sensitive user data confidential against untrusted CSPs, a natural way is to apply cryptographic approaches, by disclosing decryption keys only to authorized users. However, when enterprise users outsource confidential data for sharing on cloud servers, the adopted encryption system should not only support fine-grained access control, but also provide high performance, full delegation, and scalability, so as to best serve the needs of accessing data anytime and anywhere, delegating within enterprises, and achieving a dynamic set of users. In this it propose a scheme to help enterprises to efficiently share confidential data on cloud servers. The goal is achieved by first combining the hierarchical identity-based encryption (HIBE) system and the ciphertext-policy attribute-based encryption (CP-ABE) system, and then making a performance-expressivity tradeoff, finally applying proxy re-encryption and lazy re-encryption to our scheme.

[10] Patrick et al. published “Methods and Limitations of Security Policy Reconciliation.” which explains that a security policy is a means by which participant session requirements are specified. However, existing frameworks provide limited facilities for the automated reconciliation of participant policies. It considers the limits and methods of reconciliation in a general-purpose policy model. It identifies an algorithm for efficient two-policy reconciliation, and show that, in the worst-case, reconciliation of three or more policies is intractable. Further, it suggests efficient heuristics for the detection and resolution of intractable reconciliation. Based upon the policy model, it describes the design and implementation of the Ismene policy language. The expressiveness of Ismene, and indirectly of the model, is demonstrated through the representation and exposition of policies supported by existing policy languages. It concludes with brief notes on the integration and enforcement of Ismene policy within the Antigone.

## **B. Summary of Literature Review**

Table I Summary of Literature Review

Sr. No.	Parameters vs. ABE techniques	Author	Year	Fine-Grained Access Control	Efficiency
1	KP-ABE (Key Policy Attribute based encryption)	V. Goyal, O. Pandey, A. Sahai, and B. Waters	2006	Low	Average
2	EKP-ABE (Expressive Key Policy Attribute Based)	S. Yu, C. Wang, K. Ren, and W. Lou	2010	Better Access control than that of KP-ABE	Higher than KP-ABE, allows constant cipher text

	Encryption)				only
3	CP-ABE (Ciphertext Policy Attribute Based Encryption)	J. Bethencourt, A. Sahai, and B. Waters	2007	Average Realization of complex Access Control	Average Not efficient for modern enterprise environments
4	CP-ASBE (Ciphertext Policy Attribute Set Based Encryption)	R. Bobba, H. Khurana, and M. Prabhakaran	2009	Better Access Control than that of CP-ABE	Better than CP-ABE as there is Less collusion attacks
5	HIBE (Hierarchical Identity Based Encryption)	A. Sahai and B. Waters	2005	Lower than CP-ASBE	Better, Lower as compared to ABE schemes
6	HASBE (Hierarchical Attribute Set Based Encryption)	Zhiguo Wan, Jun'e Liu, and Robert H. Deng	2012	Better Access control	Most efficient and flexible

### III. PROPOSED SYSTEM

#### A. System Architecture

Systems architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system.

The proposed system architecture is as shown in figure 1.

#### B. Attribute Based Encryption

The cloud computing system consists of five types of parties: a cloud service provider, data owners, data consumers, a number of domain authorities, and a trusted authority. The cloud service provider manages a cloud and provides data storage service. Data owners encrypt their data files and store them in the cloud for sharing with data consumers. To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them. Each data owner/consumer is administrated by a domain authority. A domain authority is managed by its parent domain authority or the trusted authority. The HASBE scheme extends the ASBE scheme to handle the hierarchical structure of system as shown in figure-1[11].

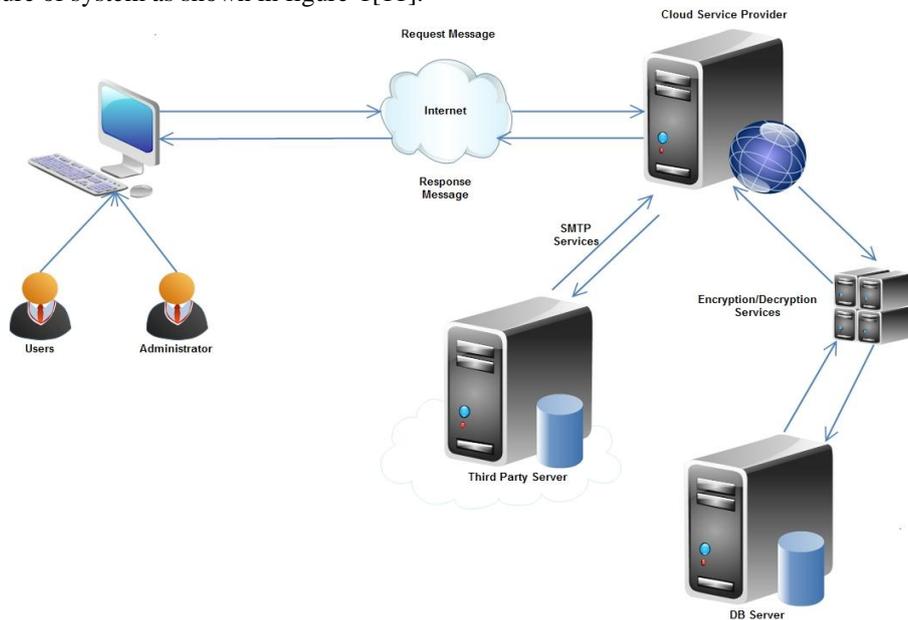


Fig. 1 System Architecture

#### C. Proposed System Architecture Description

The HASBE scheme is for realizing scalable, flexible and fine-grained access control in cloud computing. The HASBE scheme seamlessly incorporates a hierarchical structure of system users by applying a delegation algorithm to ASBE. HASBE not only supports compound attributes due to flexible attribute set combinations, but also achieves efficient user revocation because of multiple value assignments of attributes. Thus it formally proves that the security of HASBE based on the security of CP-ABE. Finally, in the implementation of the proposed scheme, and conductance of comprehensive performance analysis and evaluation, this shows its efficiency and advantages over existing schemes.

In the proposed system instead of showing complete data from cloud only those data is fetched which is essential for that user. The whole data is not fetched so it takes less time for fetching data and hence the system response time is very

less due to which system performance increases. The encryption is performed before storing data so even if data get hack by hacker data cannot be easily understand by hacker. The hierarchical structure is used, so even if lower authority is absent for particular days at that time higher authority handle all work of lower authority so work of company will not be stopped.

#### **D. Project Scope**

1. This system is designed to provide security to data stored on cloud and improve performance of system by showing only the required details requested by an employee.
2. Security is provided by generating a secret key from the various attributes stated in the form which is filled by the employee at the time of registration.
3. This system is designed to provide flexibility of the data where in case of transfer of employee, his data could be transferred to respective location with ease.
4. It also provides scalability in case when an employee is absent his work could be handled by the senior employee securely

#### **E. Methodology**

1. Registration and login by user:

In this user fill his/her own complete data. Request is sent to the CEO for confirmation. CEO confirms his/her request and assigns attribute and time period for that user. Once Account get confirm password and key is sent to that user by email so he/she can access his/her account.

2. Approve User and Assign attributes:

Out of the selected attributes according the roles defined in hierarchy of the system the attribute visibility access is decided. Each attribute is encrypted.

3. Key Generation and Verification

Key is generated based on the attributes filled by the user in registration form. In attribute key verification, when a key is used for login ,it is first checked with the key stored in the database. If a match is found then user is allowed for further process else the user is rejected for further process.

4. Encryption and decryption of data

User fills his/her data during registration. Once it is click on submit button data is send to encryption algorithm that are RSA and AES. After performing encryption data is stored in encrypted format in database.

5. Access Right:

The user can view the selected attributes of the same level as well as other levels according to the access authority using attribute key.

6. Fine Grained Access

In our propose system instead of showing complete data, the fetching of necessary data is allowed. Due to this system provides a quick response time.

7. Request for extra attribute:

The user can access attributes of same level as inter level counterparts. He can request for extra attributes in case of emergency as well as ease of work.

8. Flexibility

In this module suppose when user transfer from one location to another location at that time new location does not having rights to access data of that user .In this situation request to view attributes of required user and grant for accessing data of that user by admin is necessary. When user's data is accessible from new location then it cannot access from old location.

9. Scalability:

Since performing hierarchical structure so even if lower authority is absent for particular days at that time higher authority handles all work of lower authority so work of company will not be stopped.

10. Efficient User Revocation:

It can be done by two steps request to the admin and response to the user from admin within expiration time.

11. Privacy:

Default it is public but a user can set intra-level privacy by restricting access to attributes.

### **IV. PROPOSED ALGORITHM**

#### **A. Key Generation Algorithm**

i)Set of Attribute List= {li1, li2, li3....}

ii)Set of Employee E= {e1, e2, e3....}

a) List = List of Attribute assign to the user(E).—get list of attribute each user having

b) Foreach ( string Attribute in List ) ...retrieve one by one attribute from list

{

    Foreach (char ch in Attribute ) ...get one by one character from attribute

{

        AK =AK + ch ; ...get ASCII value of that character and make summation

    Set of attribute key AK= {ak1, ak2, ak3....} ;

- }
  - }
    - c) In the Value we get ASCII value of that character.
    - d) ASCII values save into database
    - e) Above approach generates attribute key secrete key using this secret key which has minimum length key
    - f) The RSA algorithm generates minimum lengths key
    - g) To process key an attributes with RSA it takes minimum time

**B. Algorithmic flow with mathematical equations**

The model consists of a root master(admin RM) that corresponds to the domain masters (DMs) in which the top-level DMs

- i) **CreateDM**(params,MK<sub>i</sub>,PK<sub>i+1</sub>) →(MK<sub>i+1</sub>) : To generate the master key for DM<sub>i+1</sub> with PK<sub>i+1</sub>, the RM or DM<sub>i</sub> first picks a random element  $mk_{i+1} \in Z_q$ , and then computes

$$SK_{i+1} = SK_i + mk_i P_{i+1}$$

where

$$P_{i+1} = H_1(PK_{i+1}) \in G_1, \text{ and } Q_{i+1} = mk_{i+1} P_0 \in G_1,$$

finally sets

$$MK_{i+1} = (mk_{i+1}, SK_{i+1}, Q\text{-tuple}_{i+1})$$

where

$$Q\text{-tuple}_{i+1} = (Q\text{-tuple}_i, Q_{i+1})$$

, and chosen by the RM and shared in a domain. Here, we assume that SK<sub>0</sub> is an identity element of G<sub>1</sub>, and Q-tuple<sub>0</sub> = (Q<sub>0</sub>).

- ii) **CreateUser**(params,MK<sub>i</sub>,PK<sub>u</sub>,PK<sub>a</sub>) →(SK<sub>i,u</sub>,K<sub>i,u,a</sub>) : To generate a secret key for user U with PK<sub>u</sub> on attribute a with PK<sub>a</sub>, DM<sub>i</sub> first checks whether U is eligible for a, and a is administered by itself. If so, it first sets

$$mk_u = H_A(PK_u) \in Z_q, Sk_{i,u} = mk_i mk_u P_0 \in G_1,$$

and

$$K_{i,u,a} = SK_i + mk_i mk_u P_a \in G_1,$$

Where

$$P_a = H_1(PK_a) \in G_1,$$

and then gives Q-tuple<sub>i</sub>. Otherwise, it outputs“NULL”.

iii) **Encryption**

Encryption is done using AES algorithm for search attribute and RSA algorithm is used to encrypt the user data on cloud

iv) **Decryption**

Decryption is done using AES decryption algorithm for accessing users personal credentials whereas RSA is used for accessing user data on cloud.

**C. Outcome**

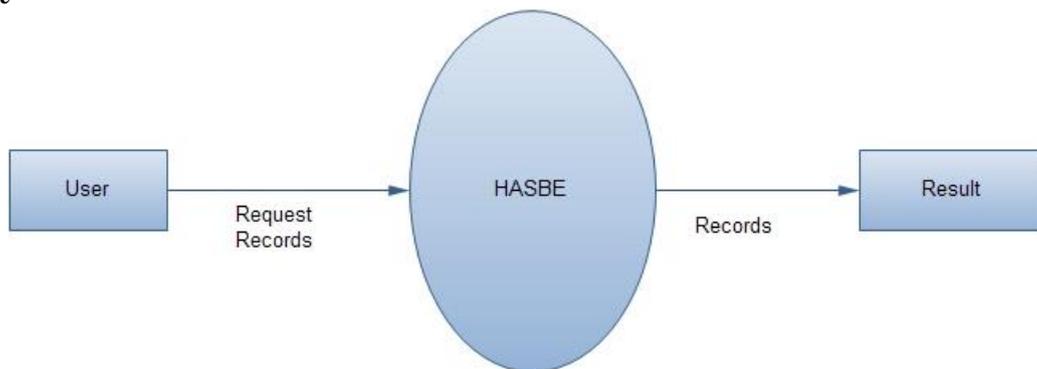


Fig. 2 Data flow diagram of HASBE

- Users: 1) Admin  
 2) Upper Level Manager  
 3) Employee.

1) Admin: - Admin approves account to every user that may be Upper Level Manager and employee. Or user whose attributes another user want to see or files (records)

He also sends a secret key to upper level user to see attributes or access files

2) Upper Level Manager: he sends request to admin for new attributes or employee of same domain.

With the help of secret key(attribute key) he can see or access file of user in hierarchy

3) Employee is end user who can request or new attributes (files) to be loaded in in cloud

He can request to upper level manager for further processing

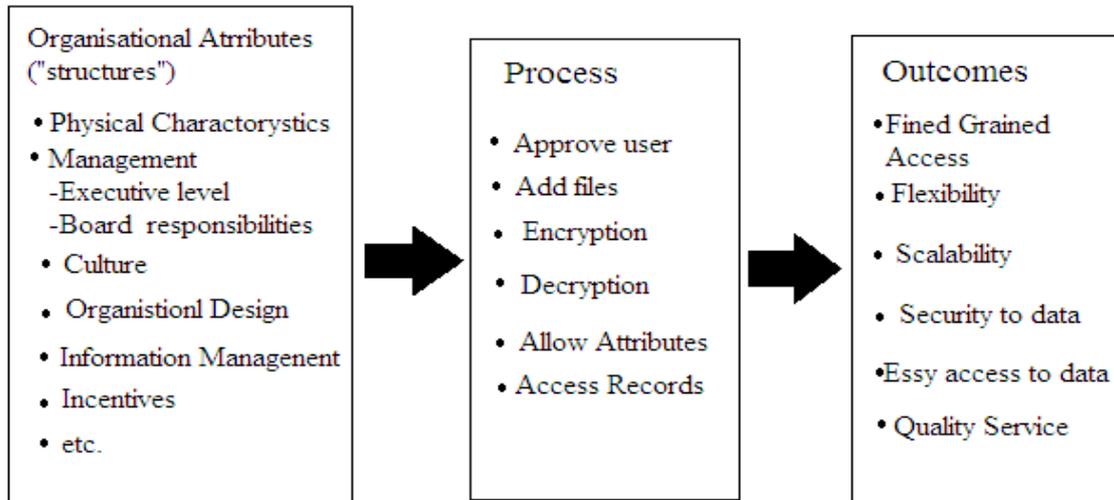


Fig. 3 Block Diagram of HASBE process flow

**D. Non Functional Requirements**

1. User may have multiple attributes.
2. Attribute may assign to multiple users.
3. Admin a Central authority

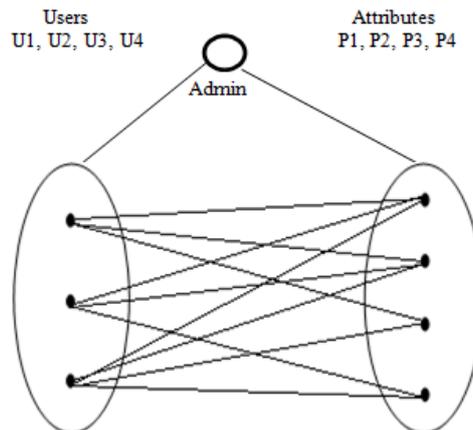


Fig. 4 Many to Many Relationship

**E. Performance requirements.**

- i) The admin acts as the root of trust and authorizes the top-level domain authorities.
- ii) A higher level manager is trusted by its subordinate domain authorities or users that it administrates, but may try to get the private keys of users outside its domain. Users may try to access data files either within or outside the scope of their access privileges, so malicious users may collude with each other to get sensitive files beyond their privileges.
- iii) The Admin is responsible for generating and distributing system parameters and root master keys as well as authorizing the top-level domain authorities.
- iv) A higher level manager is responsible for delegating keys to subordinate domain authorities at the next level or users in its domain. Each user in the system is assigned a key structure which specifies the attributes associated with the user's decryption key.

**V. EXPERIMENTAL RESULTS**

Table II Comparison result of Encryption time with AES and RSA algorithm of user attributes on cloud.

no of attributes	AES time (ms)	RSA time(ms)
0	0	0
3	0.25	0.28
6	0.35	0.42
9	0.57	0.68
12	0.67	0.79
15	0.8	0.9

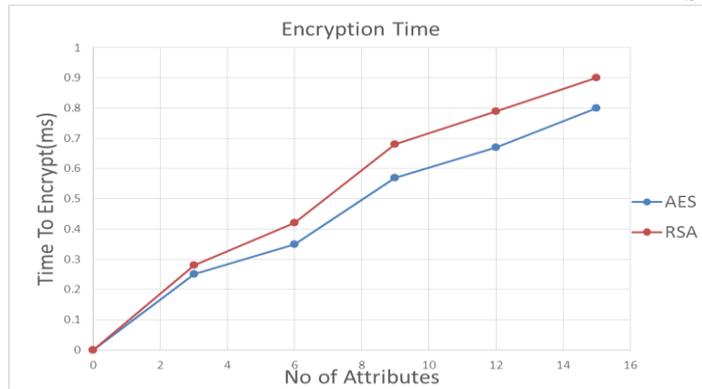


Fig. 5 Comparison graph of Encryption with AES and RSA algorithm of no. of user attributes on cloud.

The above graph shows that the encryption time required by the AES encryption algorithm requires considerably less time to encrypt the no. of user attributes so as to get access to the system.

Table III Comparison result of Decryption time with AES and RSA algorithm of user attributes on cloud.

no of attributes	AES time (ms)	RSA time(ms)
0	0	0
3	0.2	0.25
6	0.4	0.45
9	0.6	0.7
12	0.8	0.92
15	1.11	1.21

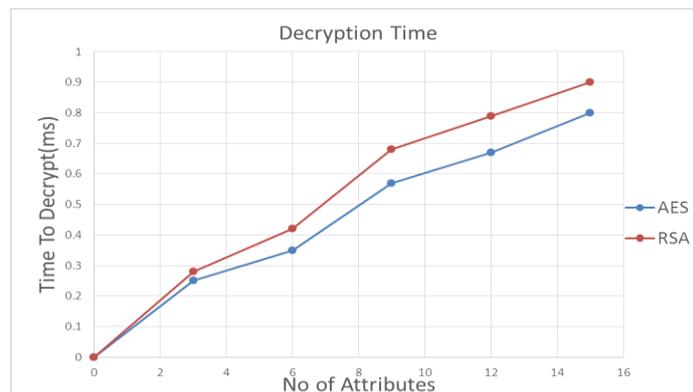


Fig. 6. Comparison graph of Decryption with AES and RSA algorithm of no. of user attributes on cloud.

The above graph shows that the decryption time required by the AES decryption algorithm requires considerably less time to encrypt the no. of user attributes so as to get access to the system.

## VI. CONCLUSION AND FUTURE WORK

Thus, the system efficiently provides a fine-grained access control with flexibility and scalability with a hierarchical structure in the HASBE system. The system will be providing security to the users from outsiders or intruders by implementing session hijacking and session fixation security in our system with SQL injection attack prevention. The core is for sure, a cloud-base thus giving users a choice of multi-user access including security from intruder attacks. Hence it benefits the users with attack handling and many advantages over the existing systems.

- We provide a secure and practical mechanism design which fulfills input/output privacy and efficiency.
- We can extend our result to any organization for secure outsourcing in cloud

## ACKNOWLEDGMENT

We express our sincere thanks to all authors whose papers in the area of cloud computing are published in various conference proceedings and journals.

## REFERENCES

- [1] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. ACM Conf. Computer and Communications Security (ACM CCS)*, Alexandria, VA, 2006.

- [2] Rakesh Bobba, Himanshu Khurana and Manoj Prabhakaran, "Attribute- Sets: A Practically Motivated Enhancement to Attribute-Based Encryption", July 27, 2009 S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.
- [3] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proc. IEEE INFOCOM 2010*, 2010, pp. 534–542.
- [4] J. Bethencourt, A. Sahai, and B. Waters. "Ciphertext-Policy Attribute- Based Encryption." In Proc. of SP'07, Washington, DC, USA, 2007.
- [5] Zhibin Zhou, Dijiang Huang" On Efficient Ciphertext-Policy Attribute Based Encryption and Broadcast Encryption".
- [6] R. Ostrovsky, A. Sahai, and B. Waters. "Attribute-based encryption with non-monotonic access structures". In Proc. of CCS'06, New York, NY, 2007
- [7] D. Boneh and M. Franklin. "Identity-Based Encryption from the Weil Pairing." In *Proc. of CRYPTO'01, Santa Barbara, California, USA, 2001*.
- [8] Guojun Wang, Qin Liu , Jie Wu "Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud services", 2011.
- [9] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in *Proc. ACM Conf. Computer and Communications Security (ACM CCS)*, Chicago, IL, 2010.
- [10] P. D. McDaniel and A. Prakash, "Methods and limitations of security policy reconciliation," in *Proc. IEEE Symp. Security and Privacy*, Berkeley, CA, 2002.
- [11] A. Sahai and B. Waters. "Fuzzy Identity-Based Encryption." In Proc. Of EUROCRYPT'05, Aarhus, Denmark, 2005..
- [12] Zhiguo Wan, Jun'e Liu, and Robert 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, April 2012.