2016

Analysis of Classical Encryption Techniques in Cloud Computing

Muhammad Yasir Shabir
Department of Computer Science, University of Management Science and Information, Kotli, AJK, Pakistan.

Asif Iqbal
Department of Computer Science, University of Management Science and Information, Kotli, AJK, Pakistan.

Zahid Mahmood
School of Computer & Communication Engineering, University of Science and Technology, Beijing 100083, China.

AtaUllah Ghafoor
Department of Computer Science, National University of Modern Languages, Islamabad 44000, Pakistan.

Follow this and additional works at: https://tsinghuauniversitypress.researchcommons.org/tsinghua-science-and-technology

Part of the Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Recommended Citation
Muhammad Yasir Shabir, Asif Iqbal, Zahid Mahmood et al. Analysis of Classical Encryption Techniques in Cloud Computing. Tsinghua Science and Technology 2016, 21(1): 102-113.
Analysis of Classical Encryption Techniques in Cloud Computing

Muhammad Yasir Shabir, Asif Iqbal, Zahid Mahmood*, and AtaUllah Ghafoor

Abstract: Cloud computing has become a significant computing model in the IT industry. In this emerging model, computing resources such as software, hardware, networking, and storage can be accessed anywhere in the world on a pay-per-use basis. However, storing sensitive data on un-trusted servers is a challenging issue for this model. To guarantee confidentiality and proper access control of outsourced sensitive data, classical encryption techniques are used. However, such access control schemes are not feasible in cloud computing because of their lack of flexibility, scalability, and fine-grained access control. Instead, Attribute-Based Encryption (ABE) techniques are used in the cloud. This paper extensively surveys all ABE schemes and creates a comparison table for the key criteria for these schemes in cloud applications.

Key words: cloud computing; access control; fine-grained access; weighted attribute

1 Introduction

Cloud computing is becoming ubiquitous as it offers fast and efficient on-demand services for storage, network, hardware, and software through the internet. Cloud computing offers new facilities to enterprises, companies, and the general public, and provides low-cost computing infrastructure for IT-based solutions. Cloud computing is not new; organizations such as universities, research laboratories, and the military in developed countries have long used networks for communication, but the term cloud is more recent[1]. Cloud computing is being increasingly offered on the web as web technology has become faster and more complex. It is now used by a large number of users to store sensitive data on third party servers, either for cost saving or for simplicity of sharing. Cloud computing is now considered the fifth utility[2] after gas, water, electricity, and telephony. There are a range of service-oriented cloud computing service models, including Infrastructure (e.g., Amazon’s EC2, Amazon S3, IBM Blue cloud), Platform (e.g., Yahoo Pig, Google App Engine), and Software (e.g., saleforce.com, Gmail, Microsoft online) as a service. Users have no need to hire IT professionals or to invest in their own software/hardware systems.

Applications that run in the cloud can balance several factors including size of data, load balancing, bandwidth, and security. One of the major barriers to cloud adoption is data security and privacy, because the data owner and the service provider are not within the same trusted domain[3]. Security issues are increasingly significant in lower layer Infrastructure as a Service (IaaS) to higher Platform as a Service (PaaS). These cloud layers are in deployed models (public, private, community, and hybrid) in high end Mobile Cloud Computing (MCC). Users hesitate to move into the cloud because certain loopholes in its architecture make cloud computing insecure. On-demand applications available in the cloud have increased; cybercrime has
also increased to launch passive and active attacks. A range of different techniques or security algorithms are used to maintain the security and privacy of the cloud. These include encryption, limited service access, stringent access, and data backup and recovery to make data retrieval easy. To ensure the confidentiality and privacy of data from a cloud service provider, a key source is an encryption technique that provides sufficiently robust security as illustrated in Fig. 1.

Attribute-Based Encryption (ABE) is newly invented public key cryptographic technique that works in a one-to-many fashion and is also called fuzzy encryption. Public key encryption methods store encrypted data on third party servers, while distributing decryption keys to authorized users. However, there are many drawbacks to this. First, it is difficult to efficiently manage the distribution of secret keys to authorized users. Second, there is a lack of flexibility and scalability. Third, data owners must be online whenever encrypting or re-encrypting data, or distributing the secret keys. ABE minimizes the above limitations by reducing the communication overhead of the internet and increasing scalability, flexibility, and fine-grained access control for large scale systems[4].

This paper provides a literature review of preliminary schemes in Section 2. A review of different ABE schemes are discussed along with analysis of strengths and weaknesses of these schemes is discussed in Section 3. Finally Section 4 concludes our work.

2 Literature Review

Cloud service providers determine the access control mechanisms for data on the cloud. Access control is a procedure that restricts, denies, or allows access to system. In the cloud, data security is crucial to protect against inside attack, denial of service attack, and collision attack. Traditionally, different expressive access control policies are used to protect data stored locally and data stored remotely[5]. The approaches include Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), and Attribute-Based Access Control (ABAC). In DAC, users are given complete control over resources on the basis of user identity. The use of DAC is not feasible when the size of the network and the number of users increase or when data is distributed across different servers. MAC is based on lattices and on the MAC decision on network configuration. In RBAC, access is based on particular roles (a set of objects related to the subject) and varies depending on the user. A role is assigned to different tasks, for example, members of staff have different roles[6].

RBAC is not feasible because all entities have the right to access and large groups would have same type of access. ABAC considers attributes based on user requests, including names and value pairs, and are associated with actions, users, subjects, objects, contexts, and policies. ABAC is more flexible, secure, and scalable and works in a hierarchical fashion. ABAC solved the RBAC problem of assigning privileges to a user. However, such access control schemes, or the use of a server as a reference monitor, cannot be applied in cloud environments because clouds have a large amount of resources, a lot of dynamic users and flexible construction because every autonomous system has its own security policy. As networks grow and the number of users increases, a more complex structure must be created, to improve the performance and reliability of stored data. The data are replicated across several locations and stored in distributive fashion across many servers. This creates a lack of confidentiality and security. The only method for protecting sensitive data across multiple sites is to encrypt the data before uploading to the server. Data stored on the cloud must be protected through different mechanisms. One of the vital techniques is public key encryption. In the traditional public key infrastructure, the data owner encrypts the data with the data user’s public key, before uploading it to the cloud. When a data user sends a request to access data on the cloud, the cloud returns a corresponding cipher text to the data user. The user then
decrypts this cipher text with the private key. There are two major disadvantages with this technique. First, for encryption, the data owner must obtain the data user’s public key before uploading. Second, because the same plaintext is used with different public keys, the storage overhead becomes excessive.

To overcome these disadvantages, ABE was introduced by Punithasurya and Priya[7]. In this approach, an attribute or set of attributes were used to encrypt and decrypt data. The user’s identity was an attribute. This technique minimized the public key encryption of all authorized users. ABE, also called fuzzy identity encryption, is similar to a previous identity-based cryptosystem introduced by Shamir[8]. In this technique, the generation of public/secret keys was based on user identity parameters such as name, network address, city, or street number, rather than by random pairing of public/secret keys. Three major entities participated in the ABE architecture: the data owner or sender of data, the user or the receiver of data, and the authority that generated the keys for both sender and receiver according to predefined attributes. If a new data user without predefined attributes was added to the system, the authority defined the attributes and then generated the public key and master key. The data owner encrypted data with the public key and a set of descriptive attributes. The data user decrypted the data with the private key, which was provided if and only if attributes of the user’s secret key matched those of the cipher text threshold values of at least “d”, where d is a threshold value.

Besides other prospective Internet of things becoming a novel and new research areas, the key enabling technologies, including the management and infrastructure in Ref. [9] and privacy and data security, will launch an innovation for academia and industry communities. Physical, information, and management security are particular crucial in Internet of Things. A new “full public verifiability” concept was proposed for hybrid public-key encryption schemes. It is a new hybrid public-key encryption scheme that has this feature, which is based on the decisional bilinear Diffie-Hellman assumption[10]. It had been proved that this hybrid public-key encryption scheme is secure against adaptive chosen cipher text attack in the standard model.

A smart XOR-based Key Freshness Scheme (XKFS) refreshes the key without inter node message transmission. It ensures the key revocation to restrict the accessibility of user to existing knowledge after node removal from the network[11]. Ideal ABE scheme[12] covers a public key based mechanism where a secret key is dependent on attribute count.

Following definitions provide a preliminary overview of confidentiality, accountability, revocation, secure access control, and collision resistance.

(1) Data confidentiality: Data is encrypted by the data owner before uploading to cloud. Unauthorized users cannot access the data.

(2) Fine-grained access control: It provides the secure accessibility to the resources. During accessibility, within the same group, users’ access rights are not the same as shown in Fig. 2.

(3) Scalability: The performance of the system is not affected if the number of authorized users increases.

(4) User accountability: Honesty can be checked to ensure that authorized users never share their private keys with illegal users.

(5) User revocation: If any user quits the system, the system revokes the access rights directly, and the user has no access to any stored data.

(6) Collision resistance: Users cannot decipher encrypted data by combining their attributes because each attribute is related to a polynomial or random number.

Pairing is based on cryptographic techniques and establishes a relationship among cryptographic groups. The ABE algorithm uses bilinear maps to establish group relationships[13, 14]. A pairing is defined as a bilinear map from two cyclic groups, $G_1$ and $G_2$ to a third group $G_t$ where each group has a large prime order $m$. Let $p$ and $q$ be the generator of $G_1$ and $G_2$, respectively. Choose two random number $a, b \in Z_m$. A bilinear map has following properties:

![Fig. 2 Access control techniques.](image-url)
1. Bilinearity: \( e(p^a, q^b) = e(p, q)^{ab} \);
2. Non degeneracy \( e(p, q) \neq 1 \).

3 Analytical Review of Schemes

3.1 ABE

ABE is a public key cryptography technique that uses one-to-many encryption. ABE uses attributes as identities for both encryption and decryption of data. The cipher text and a user’s secret key depend on attributes. If the attributes of a user key match those of the cipher text, then decryption is allowed. For example, assume that there are three attributes \{std, fac, cs\} and that the threshold value is 2, then the private key will need at least two descriptive attributes to decrypt data. This model was first proposed by Sahai and Waters\(^\text{[15]}\) to provide fine-grained access control, flexibility, and scalability in access control mechanisms in the cloud. ABE uses a set of four algorithms: setup, key generation, encryption, and decryption. Its limitations are as follows\(^\text{[16]}\):

1. Lack of an express ability in the sense of a threshold value.
2. Different categories of users create a computational overhead.

3.2 Key Policy ABE (KP-ABE)

KP-ABE was proposed by Goyal et al.\(^\text{[17]}\) as a modified form of basic ABE. Initially security parameters are setup to encrypt the message \(M\) and descriptive attribute \(S\) using \(PK\) to produce Cipher Text (CT), as shown in Algorithm 1. In KP-ABE decryption\(^\text{[18]}\), a key is embedded with an access structure and CT is annotated.

The decryption of the cipher text is only possible if the attributes of the CT satisfy the access structure of the user’s secret key as illustrated in Fig. 3. In KP-ABE, a policy is assigned to users when the authority to create key and attributes is assigned to the cipher text during its creation. KP-ABE reduces the computational overhead in a cloud server by enabling the data owner to express the access structure\(^\text{[19]}\).

Algorithm 1

\[
\begin{align*}
\text{Setup(security parameter)} & \rightarrow \text{PK, MK} \\
\text{Encrypt(PK, } M, S) & \rightarrow \text{CT} \\
\text{KeyGen(MK, } A) & \rightarrow D \\
\text{Decrypt(CT, } D,) & \rightarrow \text{M if } S \in A \\
& \perp \text{otherwise}
\end{align*}
\]

\(A = \text{access structure}\) \(D = \text{secret key}\) \(S = \text{descriptive attribute}\) \(M = \text{message}\)

KP-ABE has the following limitations:
1. A sender cannot decide who can decrypt the data.
2. It is not suitable in certain applications like sophisticated broadcast encryption.
3. It lacks flexibility and scalability.

3.3 Expressive Key Policy ABE (EKP-ABE)

EKP-ABE, as shown in Algorithm 2, is an extension of KP-ABE in which non-monotonic access structures are used. A non-monotonic access structure contains negated attributes\(^\text{[20]}\). It uses Monotonic Access structure and additional NOT gate. For example, “CS AND Std NOT graduate” means that “a student of computer science but not graduate”. EKP-ABE sets a more flexible access structure by adding a negative word in front of an attribute, meaning that a person who has such attributes cannot decrypt the data. The main limitation of EKP-ABE is that it requires many negative attributes that are not related to the encrypted data but may exist in the encrypted data (useless attributes). This may cause huge overheads.

3.4 Cipher text Policy ABE (CP-ABE)

CP-ABE is a reversed model of KP-ABE. It is another

Algorithm 2

\[
\begin{align*}
\text{Setup(security parameter)} & \rightarrow \text{PK, MK} \\
\text{Encrypt(PK, } M, S) & \rightarrow \text{CT} \\
\text{KeyGen(MK, } \tilde{A}_u) & \rightarrow D \\
\text{Decrypt(CT, } D,) & \rightarrow \text{M if } S \in \tilde{A}_u \\
& \perp \text{otherwise}
\end{align*}
\]

\(\tilde{A}_u = \text{non monotonic access structure}\) \(D = \text{secret key}\) \(S = \text{descriptive attribute}\) \(M = \text{message}\)
modified form of ABE that was best described by Rifki et al.\cite{21} The CP-ABE access structure is linked with a cipher text while the decryption key is annotated with a set of descriptive attributes, as shown in Algorithm 3. Therefore, the roles of the decryption key and cipher text are switched with respect to key policy ABE. In this scheme, encryption specifies the monotonic access structure with a threshold value for relevant attributes.

The key can be used to decrypt the cipher text if and only if the decryption key attributes satisfy the access policy in a given cipher text as illustrated in Fig. 4. This approach is more robust even if the trusted server is compromised. The concept of CP-ABE is closer to traditional RBAC. It is superior to KP-ABE in terms of enforced access control of the encrypted data.

Its main limitations are as follows:

1. The decryption key only supports logically organized user attributes in a single set.
2. CP-ABE cannot satisfy the requirements of enterprises that need flexibility and efficiency in their access control.

### 3.5 Cipher text Policy Attribute-Set-Based Encryption (CP-ASBE)

CP-ASBE is an extended form of CP-ABE, which, unlike existing CP-ABE schemes that use a monolithic set of user attributes in a key, uses a structure based on a recursive set of user attributes. In CP-ABE, a decryption key supports only a logically organized single set of attributes and to satisfy cipher text, users can use combination of all the attributes from single set issued in their key. CP-ABE is cumbersome when an enterprise has naturally occurring compound attributes with multiple numerical values for each attribute. For example, “Faculty” in a “college of information technology” serving as the “committee chair” of a “university committee” in “fall 2014” are valid attributes that describe a user. This presents a significant challenge to policies that consist of such compound attributes. Numerical attributes are limited to one value within a key. However, in many real-world systems, multiple numerical values are assigned to a single attribute as shown in Fig. 5.

To solve this challenge, a CP-ASBE scheme was introduced by Bobba et al.\cite{22} that organized user attributes with keys and allowed users to impose dynamic constraints on how attributes combined to satisfy the access policy. To achieve this, CP-ASBE organized user attributes as a recursive family set and selectively restricted decrypting users to a single set of attributes or allowed them to combine attributes from multiple sets within the given key while preventing attributes from multiple keys from being combined. Similarly, the assignment of multiple numerical values to given attributes was supported by placing each assignment in a separate set. For example, consider a user who has two values: marks 34 in binary (100010) and 32 in binary (10000). For these two numbers of 6 bits each, the user obtains values of all 12 Boolean attributes effectively and pretends to have any marks he wants. The main limitations of this approach are as

![Diagram of CP-ABE cryptography.](image1)

![Diagram of Multiple numerical attribute.](image2)
follows:\[^{23}\] (1) Combining attributes from multiple sets of attributes within a given key is a real challenge.

(2) Preventing collision by avoiding users from combining attributes from multiple keys is another challenge.

### 3.6 Hierarchical Identity-Based Encryption (HIBE)

HIBE is an extended form of IBE. In regular identity-based encryption schemes, each private key is distributed by a single private key generator, and public keys are their Primitive ID (PID), which is also called 1-HIBE\[^{24}\]. One of the vital drawbacks of this technique is its key management overhead. To minimize this, a 2-HIBE scheme was introduced that provided a precise definition of the security. A 2-HIBE scheme consists of a domain Private Key Generator (PKG), a root PKG, and users, all of which are associated with an arbitrary string of PID. A user’s public key is the combination of PID and domain PID, which is also called address. The domain PKG can compute any private user key from users’ domain, provided they have previously requested their domain secret key from the root PKG. 2-HIBE adds sub-domains. The cryptosystem includes a root certificate authority called a trusted third party that allows a hierarchy of certificates. HIBE can significantly reduce the workload on the root server and allows key escrow at several levels.

### 3.7 Hierarchical Attribute-Based Encryption (HABE)

The HABE scheme was derived by Wan et al.\[^{25}\]. This scheme offers fine-grained access control, scalability, and full delegation by combining the features of HIBE and CP-ABE. HABE works in a disjunctive clause fashion and assumes that all attributes in one conjunctive clause are administered by the same domain master. The limitations of HABE are as follows:

(1) Although the same attribute may be administrated by multiple domain masters, this is difficult to implement in practice\[^{26}\].

(2) It cannot efficiently support compound attributes.

(3) It lacks support for multiple-value assignments.

### 3.8 Hierarchical Attribute-Set-Based Encryption (HASBE)

HASBE was first proposed by Hephzi Rachel and Prathiba\[^{27}\] that combines the features of ASBE and HIBE. In HASBE, each data consumer or data owner is managed by a domain authority\[^{28}\]. There are five types of party that can participate in the system: data owner, data consumer, domain authority, parent/trusted authority, and cloud service provider. These are rearranged in a hierarchical structure as shown in Fig. 6.

The scheme builds the hierarchical structure of system users by applying the delegation algorithms of CP-ASBE as illuminated in Fig. 7. HASBE works as a recursive-set-based attribute and uses a bilinear mapping system for both encryption and decryption, as well as providing efficient user revocation to assign multiple values to users’ attributes. The limitations of HASBE are as follows:

(1) If a lower level authority is on leave or absent from work, operation is completely stopped.

(2) The domain hierarchy is very complex and the excessive time taken to fetch and execute a query degrades system performance\[^{29}\].

### 3.9 Cipher text Policy Weighted Attribute-Based Encryption (CP-WABE)

CP-WABE is a generalized form of traditional CP-ABE. In real applications, the importance of each attribute has a different weight and may not be treated as identical. For example, suppose that a head of department wishes to encrypt a document concerning a 40-year-old lecturer in the department of commerce. The access structure...
Fig. 7 Hierarchical attribute-set-based encryption.

{“lecturer” AND “CS department” AND “Age 40”} is used to encrypt the document. A user with the private key must have all three attributes in order to decrypt the document. If the categories were expanded into professor, assistant professor, and associate professor and added to the access structure, the structure becomes too complex, even when not all the possibilities are taken into account. To avoid this, a CP-WABE scheme was proposed by Liu et al.[30] in which attributes were weighted according to their importance in the access control system.

The data owner can then encrypt the data with a certain set of attributes with a weighted structure. In the decryption process the set of weighted attributes with the cipher text must match the weighted access structure. For instance, the levels “professor”, “assistant professor”, and “associate professor” can be given weights of “professor (1)”, “professor (2)”, and “professor (3)”, respectively. If the access structure is {“professor (1)” AND “CS department”}, everyone who is a professor in the CS department can decrypt the document. Both professor and associate professor cannot decrypt if the access structure is {“professor (2)” AND “CS department”} because professor (1) has a higher weighting. CP-WABE provides fine-grain access and is mainly used in distribution systems. This scheme can be considered as four types of algorithm as follows.

1. Setup \((1^\lambda, U)\) $\rightarrow$ PK, MK where
   \(1^\lambda = \text{security parameter}, U = \text{attribute universe}\)
2. Encrypt \((M, A, \text{PK})\) $\rightarrow$ CT
   (CT is associated with a weighted attribute)
3. Key Gen (MK, \(S\)) $\rightarrow$ SK
   (Where \(S\) is a weighted attribute)
4. Decrypt (CT, SK) $\rightarrow$ M

If a set of attributes contained in SK satisfies the access structure.

The limitations of CP-WABE are as follows:
(1) The computation cost is very high.
(2) The length of cipher text makes it unsuitable in some applications.

3.10 Key Policy Weighted Attribute-Based Encryption (KP-WABE)

In a traditional KP-ABE scheme, the characteristics of specified attributes are treated at the same level. In real environments each attribute has a different weight according to its importance[31]. KP-WABE overcomes the drawbacks of CP-WABE by reducing computation overhead and the size of the cipher text. In KP-WABE, the data receiver private key has a certain kind of weighted access structure and the data owner encrypts the data for all of receivers who have a certain set of weighted attributes. KP-WABE consists of four algorithms as follows.

1. Setup \((1^\lambda, U)\) $\rightarrow$ PK, MK where
   \(1^\lambda = \text{security parameter}, U = \text{attribute Universe}\)
2. Encrypt \((M, S^0, \text{PK})\) $\rightarrow$ CT
   (CT is associated with weighted attribute \(S^0\))
3. Key Gen (MK, \(A\)) $\rightarrow$ SK
   (Where \(A\) is a weighted access structure and SK contains \(A\) as output)
4. Decrypt (CT, SK) $\rightarrow$ M

If a set of weighted attributes \(S^0\) satisfies the access structure contained in SK.

In decryption, the set of weighted attributes must satisfy the weighted access structure. For example, if a head of department wants to encrypt a document for
both staff and associate professor in the management department, he may give weights 1 and 2 to “Emp(1)” and “Emp(2)”, respectively. Both staff members and associate professors can decrypt document with access structure {“Emp(1)” AND “Management department”} while staff cannot decrypt the documents if the access structure is {“Emp(2)” & “Management department”}. An attribute may also be represented with a different name in system, but treated as a single attribute with the same weight. For example both “kitty” and “pussycat” may represent a cat. We represent this nickname as a single attribute with the same weight “cat(1)”.

The limitations of KP-W ABE are as follows:
1. A source cannot decide who can decrypt the data.
2. It is difficult to manage attributes that are issued by multiple attribute authorities.

3.11 Multi-Authority-based Weighted Attribute-Based Encryption (MA-WABE)

Most existing ABE encryption techniques have only a single authority to manage both secret keys and public keys. In many situations, however, users have attributes from multiple authorities, and data owners share data with users who are administered by a different authority. To solve this problem, many different multi-authority attribute-based access control schemes have been introduced. Yang et al.

introduced a scheme called multi-authority data access control for in-cloud storage with efficient decryption and revocation. Yang and Jia also introduced a multi-authority attribute-based access control system for in-cloud data storage. In these schemes a data owner has to be online all the time in order to update cipher text. Most existing multi-authority schemes treat attributes equally and give them the same status in the access control system. In real environments, weighting of attributes is more practical. Wang et al.

introduced a multi-authority-based weighted attribute encryption scheme that adopted the concept of weighting. The system comprised five fundamental entities: the data owner, who encrypts the data under an access control policy before uploading to the cloud, a cloud server, to provide data storage, an Attribute Authority (AA) to entitle, update, and revoke users’ attributes which are assigned different weights according to their importance, a Central Authority (CA), which assigns a global user identifier to each user and a user public key to the AA, and the users, or data consumers as illustrated in Fig. 8.

In this scheme, a weighted threshold access structure is defined. The leaf node represents the weight of the attribute and the root node assigns a threshold value. If the summed leaf node weights of a user’s secret key attributes exceed a threshold value, the users can decrypt the document. For example, if User A = {CS, Kotli, Age} and User B = {IT, Lahore, Age 40}, the system assigns weight values of {6, 5, 3} and {4, 3, 2} to User A and User B, respectively. Assuming that the threshold value in the access structure is 10, User A may decrypt the cipher text because his summed weight value at leaf node is 14. In contrast, User B cannot decrypt the cipher text because his summed weight value does not exceed the threshold value as shown in Fig. 9. This scheme is more reliable, efficient, and secure, and is also closer to real cloud computing applications than existing schemes. Multi-authority weighted-attribute-based encryption offers fine-grained access control, collusion resistance, and multi-authority security.

Table 1 elucidates the comparison of the features and limitations of traditional access control schemes including DAC, MAC, RBAC, and ABAC. Table 2 comprises of comparison of features and identified limitations of ABE encryption techniques. Table 3 provides access-technique-based ABE comparison for DAC, MAC, RBAC, and ABAC. Table 4 illustrates a comparison between different algorithms under following parameters including fine-grained access control, computation overhead, user revocation efficiency, scalability and efficiency, collision resistance, association of attributes, and association of access policy. These are measured in low, average, above average, high, and very high. Moreover, association of access policy is highlighted by showing
Threshold value

W(dept.) W(City) W(Age)

User A
Kotli Age 25 CS

User B
Lahore Age 35 IT

(a) (b) (c)

Fig. 9 Weighted threshold access structure shown in (a) and for User A in (b) and for User B in (c).

Table 1 Features’ comparison of traditional access control schemes.

| Scheme | Feature | Limitation |
|--------|---------|------------|
| DAC    | User oriented and based on identity of requestor. Complete authority over all resource. Owner discretion access through administrator. | Possibility to filch the copy of the original message. Cumbrous and time consuming in large environment |
| MAC    | Access control based on security labels like secret, top secret, confidential. Used in environment where paramount importance is confidentiality. More secure and easy to scale. | Limited user functionality and high admin overhead |
| RBAC   | Access based on user’s role, not identity. User’s right can change and scalable to some degree. | Possibility of role explosion. Unable to accommodate in real-time context. Roles are static. |
| ABAC   | Access is based on user’s attribute. More secure, flexible and scalable. Attributes describe role that is built dynamically at run time. | Organization change necessary to manage attributes. |

“with key” and “with cipher” options.

4 Conclusions

ABE is an extensively used encryption technique for access control in cloud computing. The main advantage of ABE is that it gives users access to stronger encryption and allows key strength distribution. This paper has analyzed several different ABE techniques and categories, and reviewed their functionality and limitations. We extended the survey to weighted attribute based encryption techniques that perform better by offering fine-grained access control. Based on its fine-grained access control, flexibility, and scalability in cloud computing, we conclude that WABE performs as well as or better than the other schemes.

References

[1] J. Krogsie, Model-Based Development and Evolution of Information Systems: A Quality Approach. Springer, 2012.
[2] Z. Wan, J. E. Liu, and R. H. Deng, A hierarchical attribute-based solution for flexible and scalable access control, IEEE Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 743–754, 2012.
[3] B. Balasubraman and P. Venkata Krishna, Extensive survey on usage of attribute based encryption in cloud, Journal of Emerging Technologies in Web Intelligence, vol. 6, no. 3, pp. 263–272, 2014.
[4] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption, IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, pp. 131–143, 2013.
[5] B. R. Purushothama and B. B. Amberker, Access control mechanisms for outsourced data in cloud, in Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on, IEEE, 2012.
[6] A. Vijayalakshmi and R. Arunpriya, Authentication of data storage using decentralized access control in clouds, Journal of Global Research in Computer Science, vol. 5, no. 9, pp. 1–4, 2014.
[7] K. Punithasurya and J. Priya, Analysis of different access control mechanism in cloud, International Journal of Applied Information Systems, vol. 4, 2012. Doi: 10.5120/ijais12-450660.
[8] A. Shamir, Identity-based cryptosystems and signature schemes, in Advances in Cryptology, G. R. Blakley and D. Chaum, eds. Springer Berlin Heidelberg, 1985.
[9] H. Ning and H. Liu, Cyber-physical-social-thinking space based science and technology framework for the Internet of Things, SCIENCE CHINA Information Sciences, vol. 58, no. 3, pp. 1–19, 2015.
Table 2 Features’ comparison of ABE encryption techniques.

| Author & contribution | Feature | Limitation |
|-----------------------|---------|------------|
| Wan et al.[1]: Hierarchical ABE | Computation tasks are fully delegated by combining HIBE and CP-ABE | Lack of support compound attributes and multiple attribute allocation |
| Sahai and Waters[1]: Fuzzy IBE | One-to-many encryption technique for the fine grained access control | Lack of express ability in their threshold value. Create computation overhead, when users vary |
| Goyal et al.[17]: Key policy | Reducing the computation overhead by defining the access control structure with the private key | Lack of flexibility and scalability |
| Bethencourt et al.[20]: Cipher text policy | Better than key policy in the terms of enforce access control with encrypted data | Not suitable in enterprise environment because only supports logically organized single set of attributes |
| Rifki et al.[21]: Non monotonic access structure | More flexibility in their access control structure by defining negated attributes | Huge overhead in the sense of useless attributes |
| Bobba et al.[22]: Attribute set base ABE | Useful in enterprise level and support recursive family set of attribute instead of single set | Preventing collusion and combining attribute from multiple set is real challenge. |
| Wang et al.[24]: Hierarchical ASBE | Achieved scalability, flexibility, and fine grained access by supporting compound attribute with a hierarchical structure of the users | Leave or absent of any low level authority cause for delay for that duration also domain hierarchy is very complex |
| Liu et al.[30]: Weighted cipher text access structure | Improve fine grained access control by defining attribute weight according to their importance in the system. | Cipher text size is too long |
| Liu et al.[31]: Weighted key access structure | Reduce cipher text size by annotated weighted access structure with the private key also simplifying different nicknames of single attribute | Encryption cannot decide who can decrypt the encrypted data |
| Wang et al.[34]: Weighted threshold access structure | Efficient multi-authority scheme that used weighted threshold access structure and issued different attribute related keys also avoid collusion resistance | System model of this scheme is complex and how to achieve more significant access structure needs to be studied in future work |

Table 3 Access-technique-based ABE comparisons.

| Access techniques | User-oriented | Assigning role | Policy | Application | Performance |
|-------------------|---------------|----------------|--------|-------------|-------------|
| DAC               | High          | Not mentioned  | Fixed  | Administrator driven | Low         |
| MAC               | Varies from user to user | Single node assigning | Fixed  | Administrator driven | Based on security level |
| RBAC              | High          | Multi          | Flexible | Administrator driven | Above average |
| ABAC              | High          | Not mentioned  | Highly flexible | Automated adaptive | High         |

[10] K. Li, T. X. Hu, and L. J. Fen, Tight chosen ciphertext attack (CCA)-secure hybrid encryption scheme with full public verifiability, *SCIENCE CHINA Information Sciences*, vol. 57, no. 11, pp. 1–14, 2014.

[11] A. Ghafoor, M. Sher, M. Imran, and K. Saleem, A lightweight key freshness scheme for wireless sensor networks, in *12th International Conference on Information Technology - New Generations*, 2015.

[12] C. C. Lee, P. Chung, and M. S. Hwang, A survey on attribute-based encryption schemes of access control in cloud environments, *International Journal of Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[13] S. Garg, C. Gentry, and S. Halevi, Candidate multilinear maps from ideal lattices, in *Eurocrypt*, 2013.

[14] S. Garg, C. Gentry, S. Halevi, A. Sahai, and B. Waters, Attribute-based encryption for circuits from multilinear maps, *Lecture Notes in Computer Science*, vol. 8043, pp. 479–499, 2013.

[15] A. Sahai and B. Waters, Fuzzy identity based encryption, in *Proc. Advances in Cryptology-Eurocrypt*, 2005, pp. 457–473.

[16] P. Singh and S. Singh, Cross bread role based access control for extended security at Azure in cloud computing, *International Journal of Application or Innovation in Engineering and Management*, vol. 2, no. 2, pp. 205–208, 2013.

[17] 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*, 2006.

[18] C. Vinoth Et-Al, Secure data storage in cloud by using asymmetric key management based encryption, in *Ijseret*, 2015.

[19] R. Gaikwad, D. M. Dakhane, and R. L. Pandhi, Implementation and analysis of network security using Hasbe, *International Journal of Emerging Trends &
### Table 4 Feature-based comparison table of ABE.

| Algorithms | Fine-grained access control | Computation overhead | User revocation efficiency | Scalability and efficiency | Collision resistance | Association of attributes | Association of access policy |
|------------|-----------------------------|----------------------|----------------------------|---------------------------|---------------------|--------------------------|-----------------------------|
| ABE        | Low                         | AVG                  | AVG                        | AVG                       | Below AVG           | With cipher              | With key                    |
| KP-ABE     | Low if re-encryption than high | Most of computation overhead | Low                         | AVG high if broadcast type | AVG                 | With cipher              | With key                    |
| EKP-ABE    | Better than KP-ABE          | Reduced computation overhead | AVG                        | Higher than KP-ABE        | Above AVG           | With cipher              | With key                    |
| CP-ABE     | AVG realization of complex access structure | AVG                  | Low                        | AVG not efficient in modern enterprise environment | Good                | With key                 | With cipher                 |
| CP-ASBE    | Higher than CP-ABE          | Lower than CP-ABE    | Above AVG                  | Better than CP-ABE        | Good                | With key                 | With cipher                 |
| HIBE       | Comparative low             | Most of computation overhead | ...                        | Better: lower when compare with ABE | Good                | ...                      | ...                         |
| HABE       | High                        | Some of overhead     | AVG                        | Above AVG                 | Good                | With key                 | With cipher                 |
| HASBE      | High                        | Less than all of above | Above AVG                  | High                      | Good                | With cipher              | With cipher                 |
| CP-WABE    | Very High                   | High                 | Above AVG                  | High                      | Good                | With cipher              | With cipher                 |
| KP-WABE    | Very High                   | Low                  | Above AVG                  | High                      | Good                | With cipher              | With key                    |
| MA-WABE    | Very High                   | Low                  | Very high                  | Very high                 | Very good           | With cipher              | With cipher                 |

*Technology in Computer Science*, vol. 3, no. 2, pp. 170–174, 2014.

[20] J. Bethencourt, A. Sahai, and B. Waters, Cipher text-policy attribute based encryption, in *Proceeding IEEE Symposium Security and Privacy*, 2007.

[21] S. Rifki, Y. Park, and S. Moon, A fully secure cipher text-policy attribute-based encryption with a tree-based access structure, *Journal of Information Science and Engineering*, vol. 31, pp. 247–265, 2015.

[22] R. Bobba, H. Khuranaaand, and M. Prabhakaran, Attribute-sets: A practically motivated enhanced to attribute-based encryption, in *Proc. Esorics*, Saint Malo, France, 2009.

[23] B. Poornima and T. Rajendran, Improving cloud security by enhanced Hasbe using hybrid encryption scheme, in *2014 World Congress on Computing and Communication Technologies*, 2013.

[24] 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 Communication Security*, 2010.

[25] Z. Wan, J. Liu, and H. Deng, Hasbe: A hierarchical attribute based solution for flexible and scalable access control in cloud computing, *IEEE Transaction on Information Forensics and Security*, vol. 7, no. 2, pp. 743–754, 2012.

[26] R. L. Gaikwad, D. M. Dakhane, and R. L. Pardhi, Implementation of network security model in cloud computing using encryption technique, *International Journal of Recent Advances in Engineering & Technology*, vol. 1, no. 2, pp. 51–56, 2013.

[27] D. Hephzi Rachel and S. Prathiba, An enhanced Hasbe for cloud computing environment, *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 4, pp. 396–401, 2013.

[28] S. S. Aravinth and M. Ramkumar, Empowered service delegation with attribute encryption for distributed cloud computing, in *Current Trends in Engineering and Technology (ICCTET), 2013 International Conference on*, Coimbatore, India, 2013.

[29] N. Krishna and L. Bhavani, Hasbe: A hierarchical attribute set based encryption for flexible, scalable and fine grained access control in cloud computing, *International Journal of Computer & Organization Trends*, vol. 3, no. 9, pp. 294–301, 2013.

[30] X. Liu, H. Zhu, J. Ma, X. Xiong, Q. Li, and M. Jun, Cipher text-policy weighted attribute based encryption for fine-grained access control, in *2013 5th International Conference On Intelligent Networking And Collaborative Systems*, 2013.

[31] X. Liu, H. Zhu, J. Ma, and S. Ma, Key-policy weighted attribute based encryption for fine-grained access control, in *Icc14-W5: Workshop on Secure Networking and Forensic Computing*, 2014.
[32] K. Yang, X. Jia, K. Ren, and B. Zhang, Dac-Macs: Effective data access control for multi-authority cloud storage systems, in Proceedings of IEEE Infocom, 2013.

[33] K. Yang and X. Jia, Attributed-based access control for Multi authority systems in cloud storage, in Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on, 2012, pp. 536–545.

[34] Y. Wang, D. Zhang, and H. Zhong, Multi-authority based weighted attribute encryption scheme in cloud computing, in 2014 10th International Conference on Natural Computation, 2014.

Muhammad Yasir Shabir received his BSIT degree in 2009 from University of Azad Jammu and Kashmir, Mirpur Campus and MS in computer sciences from international Islamic University, Islamabad in 2012. His major research area is wireless sensor networks. He has been working in International Islamic University, Islamabad after that he joined Mohi-ud-din Islamic University Neriyan, Azad Kashmir (Pakistan) as a lecturer. He had also worked as a lecture in Campus of Mandi Baha-ud-din, University of Sargodha, Pakistan. Now he is working as a lecturer in University of Management Sciences and Information Technology Kotli, Azad Kashmir, Pakistan.

Ata Ullah Ghafoor received his BS and MS degrees in 2005 and 2007 respectively from COMSATS Islamabad Pakistan. He will complete PhD degree in 2016 from Department of Computer Science and Software Engineering, IIUI, Pakistan. From 2007-2008, he was software developer at Streaming Networks Islamabad. He is working as assistant professor/Head Project Committee in Department of Computer Science at NUML Islamabad since 2008. He has supervised 100 projects at undergraduate level and won one international and twenty national level software competitions. He is awarded ICT funding for the development of projects. He is the faculty partner for MicroSoft and Software Houses. He has software development expertise in C, C#, Java, and PHP. His interests are WSN security, steganography, IoT, NGN, VOLTE, VoIP, mobile App development for Android, and Windows 8.

Asif Iqbal received his BSc degree in 2000 from University of Azad Jummu & Kashmir Muzaffarabad and MSc in computer science from Mohi-ud-din Islamic University, Islamabad, Pakistan in 2003. His major research areas are privacy and security of cloud data storage, key management issues in cloud paradigm, data protection, privacy and authentication on untrusted domain and classical attribute based encryption techniques (ABE). He is currently the master student at University of Azad Jummu & Kashmir Muzaffarabad. He is doing research project on “Dynamically update access control structure of cloud data storage” at University of Management Science & Information Technology Kotli.

Zahid Mahmood received his BS degree in 2007 from University of Baluchistan Quetta and MS degree in computer sciences from International Islamic University, Islamabad, Pakistan in 2012. His major research area is key management techniques in wireless sensor network and lightweight cryptography techniques for Internet of Things (IoT), authentication, privacy and secure communication for wearable devices. He has been working in International Islamic University, Islamabad after that he joined the Mohi-Ud-Din Islamic University Nerian Sharif, as a lecturer. He is currently pursuing the PhD degree in computer science at School of Communication Engineering of University of Science and Technology Beijing, China.