Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Attribute-based data access control in mobile cloud computing: Taxonomy and open issues

Mehdi Sookhak [a,*], F. Richard Yu [a], Muhammad Khurram Khan [b], Yang Xiang [c], Rajkumar Buyya [d]

[a] Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
[b] Center of Excellence in Information Assurance (CoEIA), King Saud University, Saudi Arabia
[c] School of Information Technology, Deakin University, Australia
[d] Department of Computing and Information Systems, The University of Melbourne, Doug McDonell Building, Parkville Campus, Melbourne, Vic. 3010, Australia

HIGHLIGHTS

- Studying the fundamental of ABE cryptosystem based on different criteria.
- Taxonomizing the ABE access control in cloud computing.
- Analysing the existing techniques critically based on taxonomy to identify the gaps.
- Identifying open issues and challenges of the existing ABE access control techniques.

ARTICLE INFO

Article history:
Received 15 November 2015
Received in revised form
28 June 2016
Accepted 22 August 2016
Available online xxxx

Keywords:
Cloud computing
Access control
Attribute-based encryption
Thematic taxonomy

ABSTRACT

With the thriving growth of the cloud computing, the security and privacy concerns of outsourcing data have been increasing dramatically. However, because of delegating the management of data to an untrusted cloud server in data outsourcing process, the data access control has been recognized as a challenging issue in cloud storage systems. One of the preeminent technologies to control data access in cloud computing is Attribute-based Encryption (ABE) as a cryptographic primitive, which establishes the decryption ability on the basis of a user's attributes. This paper provides a comprehensive survey on attribute-based access control schemes and compares each scheme's functionality and characteristic. We also present a thematic taxonomy of attribute-based approaches based on significant parameters, such as access control mode, architecture, revocation mode, revocation method, revocation issue, and revocation controller. The paper reviews the state-of-the-art ABE methods and categorizes them into three main classes, such as centralized, decentralized, and hierarchal, based on their architectures. We also analyzed the different ABE techniques to ascertain the advantages and disadvantages, the significance and requirements, and identifies the research gaps. Finally, the paper presents open issues and challenges for further investigations.

1. Introduction

Cloud computing as the next generation of computing has become extremely popular these days and received significant interest from both academia and business. Even though there is no unique definition for cloud computing, however, one typical definition by many researchers comes from the National Institute of Standards (NIST): model for enabling ubiquitous, convenient, on-demand network access to shared pool of configurable computing resources (network, servers, storage, application, and services) that can be rapidly provisioned and released with minimal effort [1]. Cloud computing enables users to store their data at remote storage servers. These robust servers are, however, managed by a third party often called as a cloud service provider (CSP) [2,3]. Besides, computer hardware such as memory, disk space, and processor are virtualized and delivered to the end users as a service via the public Internet [4,5]. A cloud facility composed of thousands of virtual machines dispersed over a set

* Corresponding author.
E-mail address: m.sookhak@ieee.org (M. Sookhak).

http://dx.doi.org/10.1016/j.future.2016.08.018
0167-739X/© 2016 Elsevier B.V. All rights reserved.

of powerful data centers with diverse geographical points, which are interconnected using the telecommunication links. In addition, the cloud users are charged based on the actual amount of service they have used as analogous to water or electricity bill [6,7].

The cloud computing model offers a number of advantages for both users and service provider. For an end user, the benefits are as follows: rapid elasticity, measured service, minimal upfront investment, less maintenance cost and ubiquitous access to cloud services [8,9]. On the other side, the virtualization technology used in cloud computing results in a higher level of resource utilization and therefore, imposes fewer electricity costs to service providers.

Although clouds are more reliable and have more powerful infrastructure than personal computers, there are still security concerns that prevent users to deploy their businesses in the cloud and thus reduces the growth of cloud computing. The most apparent reason why individuals and businesses are not willing to delegate management of data to a cloud service provider as an untrusted third party is because they lose their physical control over the outsourced data [10,11]. Apart from that, the sensitive data in the cloud storage must be protected from unauthorized access. Consequently, the data owner needs to ensure the confidentiality of the outsourced data remains protected by using cryptographic access control systems.

Recently, researchers have proposed several data access control schemes to protect the stored data in the cloud computing. Such schemes empower the data owner to securely handle authorized users and revoke their permission rights. Attribute-based encryption (ABE) is an important technique rendering the different attributes of the data owner, user, or cloud environment to implement the data access control [12–15].

This paper comprehensively reviews the state-of-the-art attribute-based access control schemes used to protect the confidentiality of the outsourced data. We also study and classify the characteristics of attribute-based approaches by devising thematic taxonomy into six groups, namely access control mode, architecture, revocation mode, revocation method, revocation issue, and revocation controller. The main contributions of the paper are as follows: (1) studying the fundamental of ABE cryptosystem based on attributes, data access, and policies (2) classifying the attribute-based access control in cloud computing on the basis of their architecture the authority center into three groups, such as centralized, decentralized, and hierarchical access control and reviewing each group of such techniques critically, and (3) presenting a taxonomy for attribute-based access control in cloud computing and analyzing the existing ABE techniques based on the taxonomy to determine the advantages and disadvantages, the significance and requirements, and identifies the research gaps. Moreover, we identify open issue and challenges for attribute-based data access control to guide prospective researchers and scholars to choose an appropriate domain for future research and acquire ideas for further investigations. To the best of our knowledge, this is the first effort that studies attribute-based access control applied in cloud and distributed computing.

The rest of the paper is organized as follows. Section 2 presents the fundamental concepts of identity-based encryption, proxy re-encryption, and role-based access control. Section 3 discusses the concept of attribute-based encryption, attribute policy, and general circuit access structure. In Section 4, we categorize the ABE methods on the basis of their architecture and presents a comprehensive survey on the state-of-the-art ABE methods in each category. Section 5 presents our proposed taxonomy of ABE and compare current ABE schemes by using several significant parameters presented in the taxonomy. Section 6 focuses on the issues and challenges in current ABEs. Finally, the paper is concluded in Section 7.

2. Background

This section thoroughly explores the idea and the architecture of the three main cryptosystems, such as identity-based encryption, proxy re-encryption, and role-based access control, that are used to define secure and reliable access control techniques in cloud and distributed computing.

2.1. Identity-based encryption (IBE)

Shamir [16] was the first to propose the identity-based encryption to communicate securely and verify the signatures without exchanging the public or private key. The IBE scheme is constructed based on a public key cryptosystem in which the user is able to select an arbitrary string that provides a unique identity for him and is available to the other party as a public key (i.e., any combination of his name, social security number, address, phone number). The corresponding private key, however, is generated by using a Private Key Generator Center (PKG) instead of by users because if a user can compute his private key, he is able to compute the other party private key. Fig. 1 shows the difference between private key, public key and identity-based encryption.

The original motivation of Shamir for designing IBE was to simplify the certificate management in e-mail systems. When Alice sends an email to Bob, the message will be encrypted by the bob's e-mail address as the public key (bob@company.com). Upon receiving the email, Bob needs to authenticate himself to PKG for obtaining his private key from the PKG to read the mail.

Boneh and Franklin [17] improved the Shamir scheme [16] and proposed a fully-fledged IBE method based on bilinear maps between groups and computational Diffie–Hellman assumption. This scheme can help a user to delegate the duty to third parties by giving one private key to each of them in accordance to their responsibility.

2.2. Proxy re-encryption (PRE)

Proxy re-encryption (PRE) is a cryptographic primitive to turn a ciphertext encrypted under one key into an encryption of the same plaintext under different key by using a proxy. For example, Alice receives emails from many clients, and wants to leave for vacation and delegate her email access to Bob without sharing her secret key with him. The simplest way for Alice to implement the PRE scheme is to store her private key at the proxy on the email server. When a proxy receives a ciphertext for her, it is decrypted by using Alice's private key and re-encrypted using Bob's public key. The main problem of this method is that the proxy should be a trusted centre [18].

Blaze et al. [19] were the first to propose PRE scheme – BBS approach – without requiring to learn the plaintext and secret key based on the ElGamal cryptosystem [20]: Let G be a group of prime order p and let g be a generator of G, Alice and Bob distribute their public keys $PK_a = g^a$ and $PK_b = g^b$ and keeps their discrete logarithms as a secret key (a, b). The sender chooses a plaintext $(m \in G)$ and a random number $(r \in Z_p)$, generates a ciphertext $C_a = (C_1, C_2)$ where $C_1 = g^{ar}$ and $C_2 = m.g^r$, and then transmits it to the Alice. The proxy uses the given re-encryption key ($R_{a,b} = b/a \bmod q$) to divert the ciphertext from Alice to Bob by

$$\begin{aligned} C_b &= ({C_1}^{Rk_{a,b}}, C_2) \\ &= ((g^{ar})^{b/a}, m.g^r) \\ &= (g^{br}, m.g^r). \end{aligned} \tag{1}$$

Although, this scheme is semantically secure under the Decision Diffie–Hellman assumption in G, it has several issues such as follows: (1) Bidirectionality: The Proxy is able to divert the Bob's

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

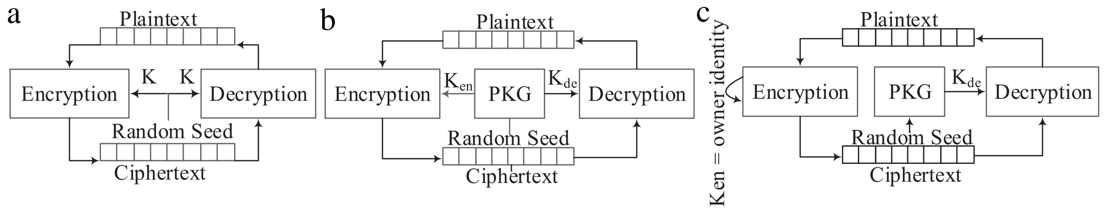


Fig. 1. Comparison private-key, public key and identity-based encryption schemes.

message to Alice by computing $(Rk_{a,b})^{-1}$ without getting permission from Bob, (2) Collusion: The proxy and Alice can collude to expose the Bob's private key ($sk_b = sk_{a,b}.sk_a$), (3) Re-encryption key generation: To compute the re-encryption key, this method needs a trusted third party, share their secret keys or to generate some secure multi-party computation by the proxy.

Ivan and Dodis [21] presented unidirectional proxy encryption by using standard public key cryptosystems in which Alice's secret key is divided by two parts ($a = a_1 + a_2$) and distributed between Proxy and Bob. Upon receiving the ciphertext ($m.g^{ar}, g^r$), the proxy computes ($m.g^{ar}/g^{a_1}$) and transfer it to another party. Therefore, Bob is able to decrypt this message by $((m.g^{ar}/g^{a_1})/g^{a_2})$. Although this method solves the bidirectional problem of the BBS approach, it also has some drawbacks as follows: this method needs a pre-secret-sharing, which does not change the ciphertext for Alice to ciphertext for Bob and Bob requires to store the additional secret key.

Ateniese et al. [22] addressed these problems and proposed a unidirectional proxy re-encryption method based on bilinear maps ($e : G_1 \times G_1 \rightarrow G_2$). They also designed a master key security without any required pre-sharing of secret keys between parties to prevent any collusion attack. After choosing a plaintext ($m \in G_2$) and a random number (r), the sender computes and transfers her ciphertext ($C_a = (z^r.m, g^{ra})$), where $z = e(g, g)$. When the message arrives at the proxy, it is re-encrypted by using the re-encryption key $\left(Rk_{a,b} = g^{b/a}\right)$ and diverted to Bob by computing the following equation:

$$\begin{aligned} C_b &= (z^r.m, e(g^{ra}, g^{b/a})) \\ &= (z^r.m, e(g, g)^{ra \times b/a}) \\ &= (z^r.m, z^{br}). \end{aligned} \tag{2}$$

In general, the PRE schemes have to meet the following requirements: (1) Unidirectional: Re-encrypting from Alice to Bob does not permit to delegate from Bob to Alice, (2) Noninteractive: The sender should be able to generate the re-encryption key by Bob's public key without requiring trusted third party, (3) Proxy invisibility: This is an important feature of PRE schemes in which the sender of a plaintext and the recipients should not be aware of the existence of the proxy, (4) Original access: The sender should be able to decrypt the re-encrypted ciphertexts that were originally sent to her, (5) Optimal Key: The size of receiver's secret key should be constant irrespective of the number of accepted delegations, (6) Collusion safeness: The scheme must be protected against collusion attack, which allows the sender for delegating the decryption rights to other party, while keeping signing rights for the same public key, (7) Nontransitive: The proxy should not be able to re-delegate the decryption right without obtaining permission from sender, (8) Nontransferable: The decryption right should not be re-delegated by the proxy and a set of colluding recipients, (9) Temporary right: the receiver can only decrypt the sender's messages that were generated during some specific time period t, and (10) Chosen-ciphertext safeness: Bob and proxy should not be able to find Alice's secret key by comparing some known ciphertext and plaintext [22–25].

2.2.1. *Applications of proxy re-encryption*

The PRE scheme has many potential applications for secure e-mail forwarding, law enforcement, digital rights management (DRM), and performing cryptographic operations on devices with storage and computation limitations. The main potential application of proxy re-encryption is securing the distributed storage where an untrusted access control server is capable of controlling access to encrypted files stored on distributed storage.

Ateniese et al. [22] implemented a distributed file system based on proxy cryptography to reduce the trust needed in the key server. In this approach, the data owner encrypts the files using a fast symmetric key cryptosystem such as AES under content keys. These keys are also encrypted under a master public key using a unidirectional proxy re-encryption scheme and are stored with files in a set of lockboxes. When an authorized client fetches the encrypted file from storage, s/he extracts the lockbox and transfers it to the access control server to re-encrypt it from the master key to the client's public key. The access control server is able to re-encrypt the lockbox and returns it to the client if it has the capability to possess an appropriate re-encryption key. The client can then decrypt the re-encrypted block using his/her secret key and decrypt the file block by it. The operation of the proxy re-encryption for securing access to distributed file system is shown in Fig. 2.

2.3. *Role-based access control (RBAC)*

The design of access control and privilege management model is a basic challenge of the large scale secure database systems and mobile distributed applications because of the dynamic nature of privileges and the fine-grained nature of entities. Role-based access control (RBAC) has emerged as a well-known approach in contrast with traditional mandatory access control, which can regulate the access of users to resources and applications based on identifying roles and activities of users in the system [26]. In general, the role is described as a semantic construct, which includes a set of tasks, authority and responsibility related to a particular working activity. Therefore, in RBAC, the access authorizations on resources are assigned to the roles instead of each individual user [27].

The RBAC can be used as follows: (1) one of the best solutions to provide the security features in multi-domain digital government infrastructure [28] and meets the complex requirement of web-based application [29], (2) a centralized policy management, which allows an organization to implement a central control over its resources, (3) a decentralized policy management, which allows many administrators with different authorizations and privileges to inhabit different locations, and (4) an efficient way to simplify security administration on the basis of roles to organize access authorization, for example, when the responsibility of a user to be changed, it only needs to assign a new role to user and revoke the old role [30].

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

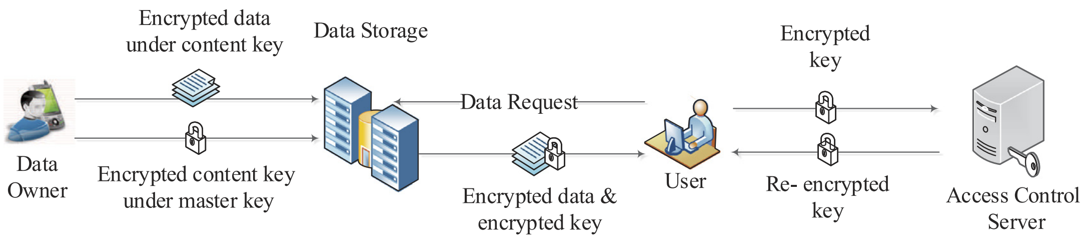


Fig. 2. Operation of the proxy re-encryption for securing access to distributed file system.

3. Attribute-Based Encryption (ABE)

Sahai and Waters [13] introduced Attribute-Based Encryption (ABE) as a new type of IBE scheme in which the identities are viewed as a set of descriptive attributes. The KPG of ABE scheme generates a user's private key considering with the attributes associated with a user's identity. In an ABE model, Bob is able to decrypt the message encrypted with a set of Alice's attributes, ω, if he has a certain set of attributes as measured by the "set overlap" distance metric. In other words, Bob can decrypt a ciphertext by his secret key, ω', if and only if at least d components of the encrypted message are matched with Bob's private key components $\left(|\omega \cap \omega'| \geq d\right)$ where d shows the error-tolerance in terms of minimal set overlap. As a result of this feature, the data owners are able to store the data on an untrusted server (i.e., cloud service provider) without the need to check authentication before delivering a document. However, this cryptosystem can only be applied for error-tolerant encryption with biometrics, which restricts it to design more general systems.

ABE schemes are classified into two main types, as follows:

1. *Key-Policy Attribute-Based Encryption (KP-ABE)*: is a public key cryptography scheme for one-to-many communications in which the user's private keys are associated with policies, while ciphertexts are labeled by sets of attributes [14]. The user's private key in KP-ABE is identified by an access-tree structure where the user's attributes are located in the leaves. The interior nodes of this access-tree are the threshold gates, which are described by their children and a threshold value k_x where $0 < k_x \leq num_x$. A user is able to decrypt a ciphertext with a given key if and only if the data access structure is satisfied by the attributes associated with a ciphertext. One of the important application of KP-ABE techniques is to ensure the confidentiality of outsourced data, for example in [31]. The main idea behind this method is to divide the data into two parts as a header and body so that the body is encrypted by using an encryption key of header part. Moreover, the data owner is able to delegate the decryption privilege to the user group by using the type-based proxy re-encryption protocol.
2. *Ciphertext-Policy Attribute-Based Encryption (CP-ABE):* is the second type of ABE cryptosystem in which ciphertexts are associated with policies, whereas the user's private key is identified with a set of descriptive attributes as a string. An encryptor specifies a policy that private keys must satisfy to decrypt the message by using an access tree structure [32]. A user is able to decrypt a ciphertext with a given key if and only if the data access structure is satisfied by the attributes associated with the private key to nodes of the tree. Fig. 3 illustrates the encryption and decryption procedure of KP-ABE and CP-ABE.

CP-ABE is more suitable to control data access in cloud storage systems than KP-ABE because it gives data owners the ability to select an access structure based on attributes and to encrypt data under this structure regarding to the corresponding public attributes [33,34]. However, applying CP-ABE method to control data access in cloud storage systems leads into two main security problems in terms of attribute revocation, such as backward issue and forward issue. (1) Backward security issue indicates a new user is able to access and decrypt the messages, which are encrypted and transmitted before joining to the system, and (2) Forward security issue indicates once a user leaved a group or an attribute of user is revoked, s/he still is able to access to future data. Since each attribute can be shared by multiple users, revocation of an attribute or a user affects the other users. In the rest of this section, we elaborate the fundamental of attribute based encryption method.

3.1. Monotone and non-monotone circuits access structure/policy

The attribute-based cryptosystem comprises two important components, such as attribute and objects. An attribute involves a unique identifying string and its hash $(x, H(x))$, and objects indicate the encrypted or recovered data based on ABE method [35,36]. An attribute policy refers to a specification of a set of attributes and threshold (as cryptographic operations) that can be used to encrypt an object. Attribute policy (P) generally can be defined as follows:

$$P = T_k(S)\,|S \subseteq A, \quad S \neq \emptyset, \; 1 \leq n \leq |S| \tag{3}$$

where S indicates a subset of all attributes, and $T_k(S)$ indicates a policy is encrypted using a set of attributes under threshold of n.

On the other hand, logical conjunction and disjunction ("AND logic" and "OR logic" policy) are also applicable to attribute policy using the threshold primitive, in which the threshold n-out-of-n attributes and the threshold 1-out-of-n attributes are necessary to decrypt an objective in the logic policy, respectively [37].

When the input of policies does not belong to a sub-set of attributes, the policy expression is more complex. For example, to encrypt an object (o_i) under a policy $P_1 \vee P_2 \vee P_3$ including three complex policies, the data owner needs to encrypt the input object (o_i) under each policy $(P_1, P_2,$ and $P_3)$ and concatenate them together $E(o_i, P_1).E(o_i, P_2).E(o_i, P_3)$. Moreover, for encrypting an objective (o_i) under a policy $P_1 \vee P_2 \vee P_3$, it requires to sequentially encrypt the object (o_i) under each policy $E(E(E(o_i, P_1), P_2), P_3)$ [35].

There are two different expression for policies, such as Monotonic Boolean policy, and Non-monotonic Boolean policy. (1) Monotonic Boolean policy: Let the set $\{a_1, a_2, \ldots, a_m\}$ be the universe of all attributes, a subset of the attribute set $(S \in 2^{\{a_1,a_2,\ldots,a_m\}})$ is monotonic if $\forall B, C\ if\ :\ B \in C\&B \subseteq C$ then $C \in S$ [38]. The main characteristic of the monotonic Boolean policy is that the policy has to be constructed from "AND logic" and "OR logic" to arbitrarily combine attributes. (2) Non-monotonic Boolean policy: Supporting a "NOT gate" as a logical primitive is a great extension for decelerating the policies especially when the user attributes are mutually exclusive. However, the monotonic policies are unable to support this feature. A naive way to overcome this problem is to define a negation for all individual attributes as a primitive attribute, which results in an additional management burden for both users and the authority. This is because each user must hold a non-negative or negative of all attributes in the system. A number of monotonic and non-monotonic ABE methods are summarized in Table 1.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

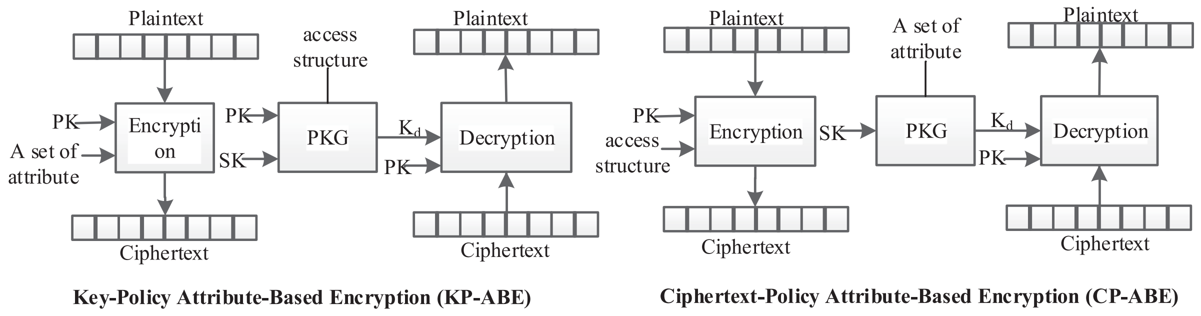


Fig. 3. The comparison between key-policy and ciphertext-policy attribute-based encryption methods.

Table 1
A review on policy expression in different type of ABE methods.

Schemes	ABE type	Pol. Exp.	Description	Drawbacks
GPSW [14]	KP-ABE	Monotonic	Using an absence of attribute to support non-monotonic policy	Increasing the total number of attributes
OSW [40]		Non-monotonic	Using the broadcast revocation scheme [41] in which a user will be revoked by giving revoked users redundant information	The private key size is increasing by a multiplicative factor of *log* (number of attributes)
LSW [42]			Optimizing the broadcast revocation scheme [41] by defining several local revocation equations instead of a global polynomial.	Losing all access right by revoking a single attribute of a user
ALP [43]			Achieving very short ciphertexts using a new identity-based revocation	The private key size is increasing a factor of number of attributes
BSW [32]	CP-ABE	Monotonic	Expressing an access predicate f in terms of any monotonic formula over attributes	Lack of satisfaction with generic group model proofs
CN [44]			It creates a direct construction for constructing a policy using a single AND gate	Restricted to a fixed number of system attributes
W [45]			Any attribute access structure can be expressed by Linear Secret Sharing Scheme (LSSS) matrix M.	More computation cost for attribute decryption
YAHK [46]	CP-ABE/KP-ABE	Non-monotonic	Supporting unbounded size of attribute set and access policies by using [42]	Proving the security only based on q-type assumptions

3.2. *General circuit access structure*

Access structure can be specified on the basis of Boolean circuits with one output wire in which each attribute relates to an input wire of the Boolean circuit [39]. When a set of attributes leads the circuit to a trust value, this set of attributes is considered as an authorized set. An access structure is defined based on all of the authorized sets.

Goyal et al. [14] were the first to propose KP-ABE method on the basis of monotonic access structure (consisting only of AND, and OR gates). Since, the key's access formula in this method is unable to support negative constraints, a new KP-ABE method is designed based on non-monotonic Boolean formulas in [40]. However, the existing KP-ABE methods [14,40–46] are vulnerable to backtracking attack when the access structure is defined by general circuit because of applying secret sharing techniques and bilinear maps together.

In other words, the main reason for creating backtracking attack is that any value calculated at the input wire of OR-gate, has to be similar to the other value, which calculated at the other input wire due to the way of secret sharing in OR-gate. As a result, if the value at one of the input wire of OR-gate is disclosed, the value at the other input wire can be computed implicitly by the attacker. Since this value can be transferred to other gates, there is a possibility to calculate the value at the output wire of the circuit, which is started from values related to unauthorized set of attributes.

Garg et al. [47] addressed backtracking problem by proposing a KP-ABE method for general circuits based on leveled multilinear maps [48] (under Decisional Bilinear Diffie–Hellman assumption) where the keys are associated with the circuits that are layered and monotonic. The work in [49]presented an ABE for circuits under the standard Learning With Errors (LWE) assumption in which the public parameters and ciphertext grow linearly with the depth of the circuit. The main idea behind this method is to overcome backtracking attack by applying Two-to-One Recoding (TOR) scheme for assessing general monotone Boolean circuits. The TOR scheme prevents the values at input wires of OR-gate to be used in other gates and instead, the key components are related to the input wires of the circuit and to the output wire of each gate. The circuit in this method is assessed bottom-up and the values related to output wires of gates in level j are powers of $g_{(j+1)}$.

Dragan et al. [50] proposed an efficient KP-ABE method on the basis of secret sharing and chained multilinear maps to reduce the number of decryption key components in [47]. The chained multilinear maps also can be defined easily and has smaller size than the circuit depth. This method divides the logical gates into two categories: logical gates of fan-out, and FANOUT-gates that multiplies the output of a logic gate. The secret sharing technique that is used to broadcast the secrete among the input wires of the circuit is as follows: (1) Sharing the output wires of a FANOUT-gate by randomizing a value rated to the input wire of the gate and transferring this random value to downward logic gates for sharing; and (2) Sharing the output wire of a logic gate among its input wires based on the input wire levels of the gate. When the shared values are received by all input wires of the circuit, a secret reconstruction procedure calculates values to each wire for bottom-up assessing the circuit. Since the bilinear map has a forward direction, this scheme can prevent the backtracking attack.

4. Attribute-based access control in cloud storage

The ABE methods play an important role in fine-grained access control in cloud computing. In the rest of this paper, we critically

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

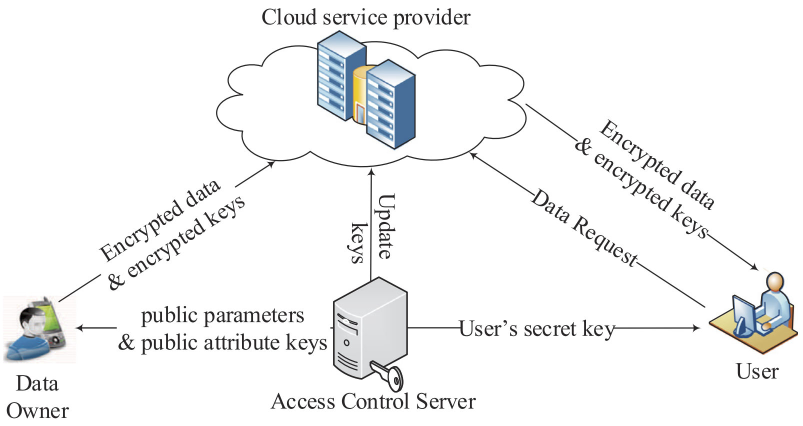

Cloud service provider
Encrypted data & encrypted keys
Update keys
Encrypted data & encrypted keys
Data Request
public parameters & public attribute keys
User's secret key
Data Owner
Access Control Server
User

Fig. 4. The architecture of centralized attribute-based access control in cloud storage.

review the existing attribute-based access control techniques in cloud computing. This paper categorizes such ABE techniques according to their architecture – the authority center – into three groups, such as centralized, decentralized, and hierarchical access control. The authority center is an important component in the architecture of ABE protocols.

4.1. Centralized attribute-based access control in cloud storage

The architecture of access control systems under the centralized ABE cryptosystem consists of four main components: (1) Data owner (Do): Who is responsible for determining attribute-based access policy, dividing his own data into different parts, encrypting each part by using symmetric encryption methods under different content keys, and then encrypting these keys by applying CP-ABE method considering with the access policy structure before storing them in the cloud storage, (2) User: The individual or enterprise that has a set of attributes depending on its roles in the system. The user is authorized to access the stored data if the access policy associated with the ciphertext which was defined by data owner is satisfied by his/her attributes, (3) Cloud service provider (CSP): This entity provides data outsourcing services and data access services for data owners and users. The CSP consists of data servers to control data access, and a data service manager to handle the attributes of users, and (4) Central authority: This is a fully trusted party that is in charge of entitling, revoking, and updating the attributes of users. It also generates public and private parameters for the systems and grants the different access to users based on their attributes. Fig. 4 shows the architecture of centralized attribute based access control in the cloud computing.

The most important challenge in designing centralized attribute based data access methods in cloud computing is to support the attribute and user revocation. This is because joining of a new user or revoking of one of the existing users frequently takes place in cloud computing. These processes have a considerable side effect on the efficiency of ABE mechanism in cloud computing. For example, when a user is revoked from the list of the authorized users, other users who have the shared attribute with the revoked user, have to update their private keys. Performing this update task incurs high computation cost on the users, especially when they are using the mobile devices with the limited computing power. Recently, some researchers have focused on a attribute based access control with efficient revocation technique. In the rest of this section, we analyze the centralized ABE methods in cloud computing and classify them based on a revocation model into four groups: timed rekeying revocation, proxy re-encryption revocation, lazy revocation, and revocable-storage ABE.

4.1.1. Timed rekeying revocation

The usual solution to address the attribution revocation issue is to use a timed rekeying mechanism. In some works such as [32,35,51], the authority is responsible to append an expiration time to each of the attributes. For example, Bob can decrypt a message, which is encrypted by Alice, iff Bob's expiration time is greater than or equal to Alice's expiration time. However, these approaches are vulnerable to forward and backwards issues. The authority also generates and periodically broadcasts the key update of users for updating the keys of non-revoked users without considering whether they are a member of the group and have the share attribute.

Boldyreva et al. [52] built an attribute-based method on the Fuzzy IBE primitive [13] and binary tree data structure to improve the efficiency of attribute revocation. In this method, each message is decrypted based on two attributes, such as the receiver identity and the expiration time. The decryption key also consists of two parts as private key and key update that are corresponding with identity and time period. Since the trusted authority is in charge of generating the private key and the key update, it can simply revoke the attribute by holding the key update distribution. To reduce the number of computation key update by the authority from linear to logarithmic, the binary tree is used in which users are the leaves of tree and the intermediate nodes are the polynomial of a decryption key. Each user is able to obtain the identity key by computing polynomials of all nodes on the path from the user node to the root node. However, the main weak point of this method is that it is defenseless against forward and backward issues.

4.1.2. Proxy re-encryption revocation

Hur and Noh [53] proposed an efficient attribute revocation scheme based on CP-ABE in which the CSP is responsible to overcome the forward and backward issues using the re-encryption method. Hence, before transferring data to users within the attribute revocation phase, the encrypted data association with data access structure and a set of attribute groups will be re-encrypted under a set of the membership information for each attribute group and the new set of group attribute keys. Upon receiving the re-encrypted ciphertext, the user needs to compute the attribute group key from the data access structure and then decrypt the ciphertext by using the secret keys. However, the performance of this method is dependent on the trustworthy level of the CSP.

Yang et al. [54] considered the problem of semi-trust cloud server for attribute revocation scheme and designed an attribute-based fine-grained access control by assigning a number to each attribute as a version key. When a user's attribute is dropped, a new version key and an update key will be generated by the trusted authority to update the secret key of all the non-revoked users (to overcome backward issue). Furthermore, the users who newly joined to the group and have a sufficient attribute need to access the previous data by updating all the ciphertexts associated with the revoked attribute. Since preforming the modification by the data owners incurs a heavy overhead on them, the ciphertext update is carried out by PRE method in CSP (to overcome forward issue).

Cheng et al. [55] proposed an efficient method to optimize the revocation scheme based on CP-ABE. To reduce the cost of revocation operation, data owner selects a random block of the file as the dynamic data instead of whole file to perform the revoke operation. However, the main drawback of this method is that the data publication and retrieval overhead are increasing due to the usage of a secret sharing scheme [56] for dividing the input file into n blocks and sharing the blocks.

Tysowski and Hasan [57] designed a key management to ensure the security of outsourcing data in the cloud for resource-constrained mobile devices. To reduce the computational and communication overhead on mobile devices of the data owner or the user, the authors considered two following ways: (1) associating the mobile devices with the cloud to generating the keys, and (2) combining the proxy-based re-encryption with the CP-ABE to delegate the required computation of the attribute revocation process to the cloud.

4.1.3. Lazy revocation

Although the proxy re-encryption is an important technique to perform the access revocation in several methods [53,54], it imposes a huge overhead on the client side and server side. To reduce the overhead of access revocation, Fu [58] presented a crucial method, namely lazy revocation in which the required re-encryption is postponed until the next write access request. In other words, the lazy revocation greatly decreases the number of required re-encryption to improve the performance of the system. Since the lazy revocation leads to fragmentation of encryption keys, the lazy revocation needs key regression [59] or key-updating schemes [60] to generate the old version of keys for decrypting the files that are not yet re-encrypted. Zarandioon et al. [61] improved the backes key-updating schemes by proposing a new hierarchical key updating method on the basis on Bilinear Diffie–Hellman problem to support lazy revocation.

Barsoum and Hasan [62] introduced a method, which allows the data owner to dynamically outsource the data to the cloud storage and manage data access and user access revocation. The authors also designed a data structure, namely Block Status Table (BST), to support dynamic data update. The BST is stored in the local storage of the DO, who is responsible for updating the table during modify, insert, and delete operations. This table contains three columns, namely: Serial Number (SN), Block Number (BN), and Version Number (VN). The SN is the actual (or physical) position of the block in the file while the BN shows the logical location of the block in the file. The VN of each block indicates the number of dynamic operations applied to the block so far. Upon outsourcing a data block for the first time, the VN is set to one and for every dynamic operation of this block, the VN is incremented by one. Furthermore, to support access control over dynamic data, three cryptographic techniques are combined in this approach, namely broadcast encryption, lazy revocation, and key rotation. The broadcast encryption allows the data owner to encrypt the rotation secret key to enable the authorized users to access the outsourced data. The lazy revocation is also used to efficiently support user revocation. Finally, the key rotation technique is in charge of enabling the authorized users to access the updated or new block that is encrypted by using a new key. It is because after updating a block or generating a new block, the block must be encrypted under a new key that is generated by the rotation secret key.

4.1.4. Revocable-storage attribute-based encryption (RS-ABE)

Since the user's credentials can be changed over the time in the organizations, the ABE data access methods have to support user revocation technique for revoking the private keys of some of the users. When a user is revoked at time T, he/she cannot decrypt the messages, which will be encrypted after the revocation time. However, the revoked user can still learn information about the messages that were created before the revocation time (T). The first idea to overcome the revocation issue is to update all decrypted messages by using the publicly available information when some users' credentials are revoked. However, by increasing the number of revoked users, the decryption time is linearly increasing and the ciphertext may grow enormously because of the requirement to re-encrypt all messages after revoking a user.

Sahai et al. [63] was the first to address this issue by proposing a revocable storage ABE method (RS-ABE) that enables an untrusted third party to store ciphertexts for revoking access on previously encrypted data. The main idea behind this method is to introduce a new concept, namely ciphertext delegation, in which a ciphertext with access policy C_p is delegated to a more restrictive policy $C_{p'}$ by adding the time to a set of attributes and using publicly available information (public keys).

Lee et al. [64] reduced the length of ciphertext by recommending a new ABE method, namely self-updatable encryption (SUE), in which a user's private key and ciphertext are associated with time. As a result, a user has capability to decrypt the ciphertext with policy p when the time of private key is a head of the time of ciphertext and the attributes of his private key satisfies the policy. The user revocation mechanism in SUE is constructed based on [52] in which the complete subtree method is used to update the keys of non-revoked users securely.

4.2. Decentralized attribute-based encryption

The main restriction of the centralized ABE protocols is the existence of only one central authority to generate the private keys for the users based on the verified attributes. Therefore, these protocols can be used to share information on the basis of a policy of a unique entity or organization. However, most of the time, there are different entities or organizations with different policies to share the information. As a result, we require designing a protocol to support different entities who monitor the attributes of all users.

Chase [65] was the first to introduce a multi-authority ABE protocol by assigning a global identifier (GID) for each of the users. The GID is a unique name, serial number, or any identifying string that helps the authorities to distinguish the users beyond their attributes and prove the users' credentials. In the multi-authority system, one of the authorities is considered as a central authority (CA) who is responsible for generating a setup keys for each user on the basis of the GID of user. The CA also hold a master secret key to decrypt the messages. Each of the authorities uses a pseudorandom function (PRF) to generate the random secret keys for each user based on the corresponding GID. It is because to make the secret key independent from the other users' keys and prevent collusion. If the user has the sufficient attributes for satisfying each of the authorities, the CA allows the user to decrypt the message by computing the additional value, which must be combined with the user secret key. However, the Chase scheme is unable to distribute control over multi untrusted authorities because of the existence of the central authority that has the capability to decrypt every message. Furthermore, the privacy of users are not preserved against the authorities due to the usage of a consistent GID.

Lin et al. [66] addressed these issues and designed a threshold based scheme, namely multi authority fuzzy identity based encryption (MA-FIBE) scheme, without requiring a central authority.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

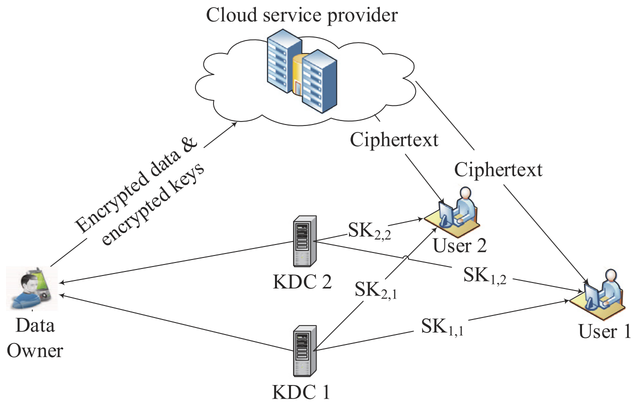


Fig. 5. The architecture of DACC method.

The authors employed the Distributed key generation (DKG) protocol [67] and joint zero secret sharing (JZSS) protocol [68] to remove the central authority. However, the main difficulty of this method is that the security of the system will be compromised with collusion of more than t users, where t is a system parameter selected at setup phase and directly determine the efficiency of the system.

In [69], the Chase scheme [65] is improved by using a distributed PRF to remove the central authority. The authors designed an anonymous key issuing protocol to preserve the security of users against the collusion of authorities. The users are also able to communicate with the authorities through pseudonyms instead of using the GID. Since at least one attribute from each of the authorities should be released for the users, this method is impractical.

Lewko and Waters [70] proposed the fully decentralized ABE protocol in which any party is able to act as an authority to issue secret keys for the users. The previous decentralized ABE protocols [65,69] prevent the collision of authorities by randomizing the users' secret keys. However, such randomizing is unable to simultaneously provide the autonomous key generation and collusion resistance goals due to the lack of an entity to compile all pieces. As a result, the authors employed a hash function on the users' GID to provide collision resistance across multiple key generations in various authorities. On the other hand, the user is able to encrypt the data by employing Linear Secret Sharing scheme (LSSS) that is issued on the basis of any set of authorities. Although the Lewko and Waters method supports the attribute revocation feature, it incurs high computation overhead on the users because this method is constructed based on bilinear groups of composite order.

Ruj et al. [71] extended the Lewko and Waters scheme [70] by designing a distributed access control scheme for cloud computing (DACC) on the basis of bilinear pairings on elliptic curves. The DACC scheme supports collision resistance and user revocation without having to redistribute the keys to all users. The main novelty of such method is considering a new entity as key distribution centers (KDCs) to generate and distribute secret keys to the users. In other words, the KDCs can be managed by different companies to issue various credentials for the users on the basis of the policy of organizations. The architecture of the DACC method is illustrated in Fig. 5. To encrypt the data in DACC method, the user, firstly, converts the Boolean access tree to the LSSS matrix and outsources the encrypted data along with the LSSS matrix to the cloud. When a user u_n with a set of attributes I_{u_n} is revoked, all users must change the stored data that have the attributes of the revoked user. However, this method incurs high communication overhead on the users because they must send a new ciphertext to all of the non-revoked users. Furthermore, such the method is unable to support user authentication. On the other hand, the users do not have permission to modify the outsourced data in the cloud.

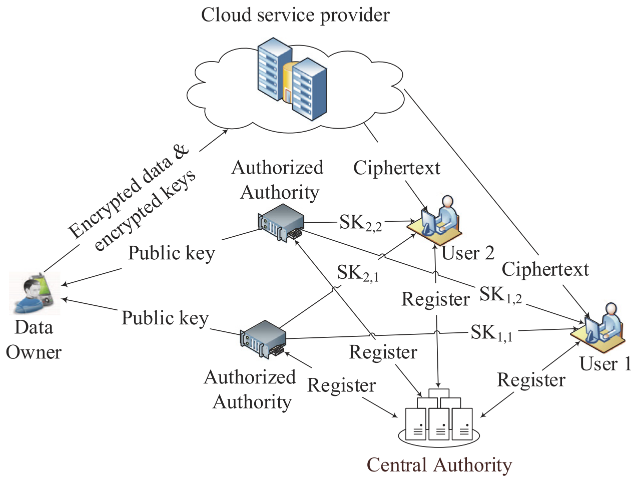


Fig. 6. The architecture of DAC-MACS method.

Yang et al. [72] proposed an efficient data access control for multi-authority cloud storage (DAC-MACS). The authors considered two types of authorities in the DAC-MACS scheme: (1) Global certificate authority that is responsible for generating a global identity for each user and authority, and (2) Attribute authorities who are in charge of generating secret key for each users based on their global identity and public key for each attribute; and revoking and updating users' attributes. The DAC-MACS scheme achieves the collision resistance by using the user and authority identity because all of the attributes can be easily recognizable. The authors also implemented a token-based decryption outsourcing method based on outsourcing the decryption of ABE ciphertexts method [37] to improve the efficiency of decryption function on the user side. However, the revoked user is still able to decrypt new ciphertexts. Furthermore, if the recently join user has got enough attributes, she can decrypt the previously published ciphertexts (Forward Security) [72]. Fig. 6 shows the architecture of the DAC-MACS scheme.

In [73], Yang and Jia suggested a revocable data access control scheme for multi-authority in the cloud storage by improving the DAC-MACS method and extending the single-authority method [74]. For example, to improve the efficiency of the attribute revocation algorithm, instead of updating all of the ciphertexts that associated with any attribute within the authority in the DAC-MACS method, the ciphertexts of the revoked attributes need to be updated only. Moreover, unlike the DAC-MACS method, each attribute is able to be appeared more than one time in a ciphertext. Similar to the DAC-MACS method, the authors divided the authorities into the global certificate authority and the attribute authorities. However, to prevent the global certificate authority from decrypting the ciphertext, the attribute authorities are responsible to generate the required public keys for encrypting the data together. Furthermore, the attribute authorities generate a version number for each attribute to address the attribute revocation problem. The ciphertext updating process during the attribute revocation function is also delegated to the servers for improving the efficiency of the method.

The main limitation of the DAC-MACS and Revocable DAC-MACS scheme is that the authenticated users are unable to access the outsourced data anonymously. Ruj et al. [75] overcame this issue by proposing a decentralized access control scheme that supports the anonymous authentication characteristic. In other words, the cloud has the capability to verify the authentication of users without having to know their identities by using attribute based signature scheme [76]. To prevent the replay attack in the attribute based signature scheme, the data owner must attach the

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

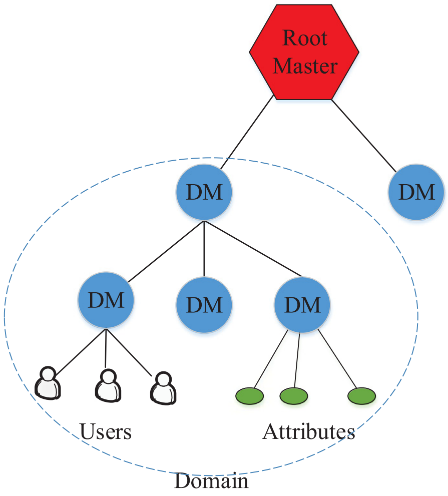


Fig. 7. Hierarchical attribute-based encryption scheme architecture.

time stamp t to the signed message before sending to the cloud. As a result, the revoked users are unable to create a new signature with new time stamp to modify the data.

4.3. Hierarchical attribute-based encryption

Gentry and Silverberg [77] were the first to propose a Hierarchical ID-based encryption (HIDE) to support one-to-many encryption. In other words, a root private key generator is able to securely delegate its duty to lower-level private key generator to generate the private keys for users in each domain.

Wang et al. [78] designed a hierarchical attribute-based encryption (HABE) scheme on the basis of the HIBE and the CP-ABE systems to provide the fine-grained access control in the cloud computing. The HABE scheme consists of the following entities: (1) a root master who is in charge of generating the system parameters and domain keys, (2) multiple domains who are responsible for transferring the keys to the domain masters and secret keys to the users, (3) domain masters (DMs) who are known as the users' administrator in each domain, and (4) the users and attributes that are connected to leftmost and rightmost DMs, respectively. Fig. 7 shows a simple architecture of HABE scheme.

The HABE method includes three main phases: (1) Creating a file: Before sharing the generated file, the data owner needs to define a disjunctive normal form policy, encrypt the file using BGHP method [79] and transferring the divided file to the cloud. After receiving the file, the CSP verifies the sender and distributes the file among servers. (2) Adding a new user: Upon joining a new user to the system, a unique ID and a set of attributes are selected for the user. Then, by using the Extraction algorithm in the HIBE system [77], a private key is generated for the user. (3) User revocation: In the first step of user revocation, the attribute list of DMs should be updated. After receiving the user ID and the corresponding list of attributes, the CSP deletes the user ID from user list and then the user secret key should be updated by applying the proxy re-encryption and lazy revocation techniques. However, using the disjunctive normal form policy for creating a file and putting all attributes under a specific DM make the implementation of the HABE difficult. This is because each attribute is administrated by several DM. Moreover, the combination of attribute is not efficiently supported by this method.

Zhiguo et al. [80] addressed the limitations of HABE method by extending the CP-ABE method to propose a hierarchical attribute-set-based encryption (HASBE) scheme. Furthermore, instead of using proxy re-encryption and lazy revocation techniques to support the user revocation, the authors used multiple values as access expiration time to improve the efficiency of the method. To encrypt the data in the HASBE method, the data owner uses a tree access control [81] in which the leaf nodes are attributes and the intermediate nodes are threshold gates. On the other hand, the authorized user is able to decrypt a ciphertext if a recursive set based key structure of the user satisfies the tree access structure where each element of the key structure can be a set or an element corresponding to an attribute. The HASBE method consists of two main phases: (1) Creation file: the data owner must encrypt the file using a symmetric data encryption key (DEK) and then defines an access tree structure to encrypt the DEK. (2) Key revocation: a key-expiration attribute is embedded in the user's key to show the validation of the key. The policy associated with the data is used for checking the validation of the key.

Most of the existing HABE methods suffer from hierarchical relation between attributes in the same category [78,80,82,83]. To overcome this issue, Wang et al. [84] proposed a new ciphertext-policy hierarchical attribute-based encryption (CP-HABE) method on the basis of sets of attribute paths that are defined in attribute trees. The authors leveraged Linear Integer Secret Sharing (LISS) [85] for presenting the access control policies by using a distribution matrix.

In [86], the author focused on the scalability and flexibility issue in access control area and proposed an efficient CP-HABE method by making the size of ciphertext and the computation of bilinear pairing constant in cloud computing. The authors used a system model with hierarchical structure, which consists of root-level authority, top-level domain authority, and low-level domain authority for managing the attributes (Fig. 8). The root authority is responsible of generating the system parameters and authorizing the top-level domain authorities. The low-level domain authorities are also responsible for managing the other low-level authorities or the data owner and users in their authorities. In other words, each data owner is managed by her low-level authority, which this authority is managed by its parent authority. This hierarchical structure with inherent properties reduces the computation cost of encryption and decryption of the data. Before outsourcing a new file to the cloud servers, the data owner encrypts it by using a symmetric key and then the key will be encrypted based on the data owner attributes. As a result, if the user's attributes satisfied the access control, the user is able to obtain the symmetric key in order to decrypt the outsourced file.

Wang et al. [87] developed a file hierarchy ciphertext-policy attribute-based encryption (FH-CP-ABE) scheme by extending the CP-ABE method [32] and using the layered model of access policy as a hierarchical structure. The main idea behind this method is to integrate the different access structures of files in to a single access structure, which can be used to encrypt the files in the same hierarchical structure. The core advantage of the FH-CP-ABE method is that the computation cost of encryption and decryption of the files decreases as well as the storage cost of ciphertext. For example, to share k hierarchical files with k access levels in the cloud computing, the data owner is able to encrypt the different level of the files by using an integrated access structure. Since the data owner needs to compute an integrated ciphertext and the common attributes only one time, the computation cost of the system is increasing dramatically. Fig. 8 depicts the architecture of the FH-CP-ABE method.

5. Taxonomy and comparison of attribute-based encryption data access

Fig. 9 shows the thematic taxonomy of ABE methods for accessing data in the cloud computing. The methods are categorized

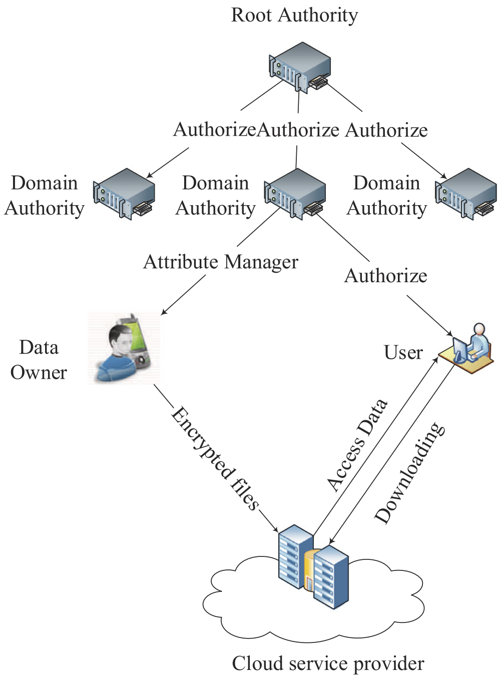


Fig. 8. Ciphertext policy hierarchical attribute-based encryption (CP-HABE) scheme architecture.

based on the following characteristics: type of data access, architecture, type of revocation, revocation method, revocation security, and revocation controller.

(1) Type of data access: there are two different type for ABE access control, namely CP-ABE and KP-ABE. (i) In CP-ABE, the user encrypts the data based on an associated access structure over attributes while the user's private key includes a number of attributes. Therefore, to decrypt a ciphertext, the user's attributes must fulfill the ciphertext's access structure. (ii) In KP-ABE, the user's private key is associated with access structure and the ciphertext is labeled with a set of attributes. As a result, the user can decrypt the message, if the key's access structure is satisfied by the attributes associated with a ciphertext.

(2) The architecture of ABE protocols falls into three categories, such as centralized, decentralized and hierarchical. (i) The centralized architecture only has a central authority center who is in charge of issuing the key for the users. The centralized architecture protocols discussed in [1–5]. (ii) The decentralized architecture consists of multi authorized authorities to share information on the basis of policies of various organizations. The decentralized architecture protocols discussed in [6–10]. (iii) Hierarchical: To improve the scalability and flexibility of ABE methods, and to support one-to-many encryption feature, the users should be in a hierarchical structure.

(3) Type of revocation: one of the important issue in ABE techniques is how to prevent the users from accessing the ciphertext by revoking the user's permissions or attributes. There are two types of revocation, such as user revocation and attribute revocation. (i) In the user revocation, the revocation controller unit prevents a user from accessing a data by using the user revocation mechanism. The algorithm discussed in [75,78,80]. (ii) In the attribute revocation, the revocation controller unit eliminates the attribute from a list of attributes of the user. The algorithm discussed in [52–55].

(4) Revocation method: This characteristic indicates the various methods that are used to revoke the user or attribute in the ABE protocols. These revocation methods include time re-king, proxy re-encryption, lazy revocation, LSSS matrix, and update key.

(5) Revocation Issue: applying CP-ABE method to control data access in cloud storage systems leads into two main security problems in terms of attribute revocation, such as backward issue and forward issue. (i) Backward issue means the user who newly joins system and has sufficient attributes is able to decrypt the ciphertext and access to the plaintext of previous data published before he holds the attribute, and (ii) Forward issue means the user who revokes an attribute is able to access any new plaintext and decrypt any new ciphertext that needs the dropped attribute to decrypt. Since each attribute can be shared by multiple users, revocation of an attribute or a user affects the other users.

(6) Revocation controller attribute who is responsible to performing the attribute or user revocation mechanism. Most of the time, data owner should revoke the user or attributes, for example [54,55,70]. However, the data owner is able to delegate the revocation task to the server [35,53] [35, 53] or the authorized authority [52,72,78,80].

The comparison summary of attribute based encryption methods based on the important characteristics and requirements that are presented in the thematic taxonomy are publicized in Table 2.

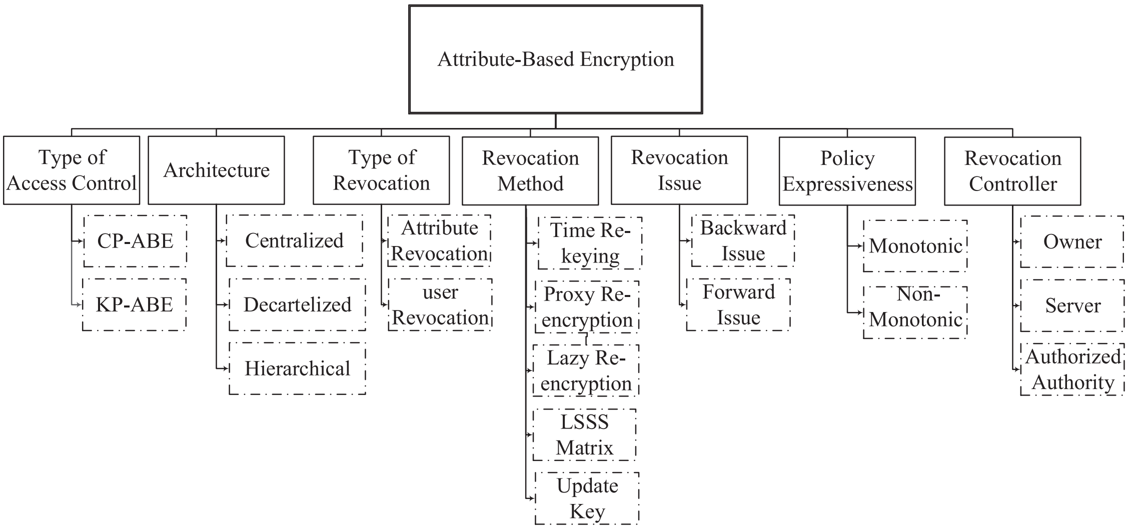


Fig. 9. Taxonomy of attribute-based access control in cloud computing.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

Table 2
Comparison of attribute based access control methods based on the thematic taxonomy.

Schemes	ABE type	Architecture	Backward security	Forward security	Revocation			
					Attribute	User	Method	Controller
BSW [32]	CP-ABE	Centralized	No	No	No	Yes	Time rekeying	Authorized authority
PTMW [35]			No	Yes	Yes	No	Time rekeying	Server
BGK [52]			No	No	No	Yes	Time rekeying; Binary tree	Authorized authority
HN [53]			Yes	Yes	Yes	Yes	PRE	Server
YJK [54]			Yes	Yes	Yes	No	PRE	Owner
CWM [55]			No	No	Yes	No	PRE	Owner
TH [57]			No	No	No	Yes	PRE	Owner
K2C [60]	KP-ABE		No	No	No	Yes	LRE	Owner
BH [62]	Broadcast encryption		No	No	No	Yes	LRE	Owner
C [65]	KP-ABE	Decentralized	No	No	No	No	–	–
MA-FIBE [66]			No	No	No	No	–	–
LCLS [69]			No	No	No	No	–	–
LW [70]	CP-ABE		No	No	No	No	–	–
DACC [71]			Yes	No	No	Yes	LSSS matrix	Owner
DACMACS [72]			Yes	Yes	Yes	No	Update key	Authorized authority
RSN [75]			Yes	Yes	No	Yes	LSSS matrix	Owner
HABE [78]		Hierarchical	No	No	No	Yes	PRE, LRE	Authorized authority
HASBE [80]			No	Yes	No	Yes	PRE, Time rekeying	Authorized authority
CP-HABE [86]			No	No	No	No	–	Authorized authority
FH-CPABE [87]			No	No	No	No	–	Authorized authority

Table 3
Performance comparison of the existing attribute-based access control methods for cloud computing.

Schemes	Ciphertext size	Rekeying size	Private key size	Public key size	Computation cost (ms)	
					DO	User
H [15]	$(2t+1)\times C_0+C_1+C_T$	$(m+2)\times C_0$	$(2k+2)\times C_0$	C_0+C_1	$2t+1.2$	$(5.8+m)\times k+0.2\log t+5.8$
BSW [31]	$(2t+1)\times C_0+C_1+C_T$	$m\times C_0$	$(2k+1)\times C_0+C_{Kek}$	C_0+C_1	$2t+1.2$	$5.8k+0.2\log t+2.9$
YWRL [33]	$(u+1)\,C_0+C_1+C_t$	$2umC_0+2uC_p$	$(2u+1)\times C_0+C_k$	$(3u+1)\times C_0+C_1$	$u+1.2$	$2.9u+k+2.9$
LLLS [36]	$(C_T+3)\,C_0+C_1+C_T$	$2r\times\log\frac{n}{r}\times C_0$	$\left(k+3+\frac{C_T}{t}\right)\times\log C_0$	$\left(\frac{C_T}{t}u+6\right)C_0+C_1+C_P$	$2kt+3.2$	$2k^2+5.8k+14.5$
OSW [37]	$(2t+1+3r)\times C_0+C_1+C_T$	–	$\log n\times(2k+1)\times C_{kek}$	$(\log n+2n+2)\times C_0+\log n\times C_1$	$2t+4.1$	$k^2+8.9k$
HN [51]	$(2t+1)\times C_0+C_1+C_T$	$(n-m)\times\log\frac{n}{n-m}\times C_P$	$(2k+1)\times C_0+\log n\times C_{kek}$	C_0+C_1	$2t+1.2$	$6.8k+0.2\log t+2.9$
SSW [61]	$(2l+1)\times C_0+C_1$	–	$(4t+1)\times C_0$	$(l+2)\times C_0+C_1$	$2t+1$	–
HABE [76]	C_t+r+1	–	$(\lvert s\rvert+2)\times n$	$(\lvert s\rvert+2)\times 2n$	–	–

C_0: bit size of an element in G, m: the number of users in an attribute group, C_1: bit size of an element in G_T, n: the number of all users in the system, C_T: bit size of an access tree T in the ciphertext, r: the number of revoked users, C_kek: bit size of a KEK, k: the number of attributes associated with private key of a user, C_P: bit size of an element in Z_P^*, t: the number of attributes appeared in T, C_k: bit size of the attribute set associated with the attribute secret key of a user, l: the number of columns of an access structure (in [61]), $|s|$: the size of an attribute set, u: the size of the attribute universe.

6. Performance evaluation

There are different parameters to evaluate the performance of the existing attribute-based encryption methods in cloud computing as follows: (1) Ciphertext size (communication cost): The size of file that the data owner has to send to the cloud service provider or the size of file that cloud service provider sends to users, (2) Private key size (storage cost): It indicates the required storage for each user to store the private key, (3) Public key size: It shows the required storage to store the public key of authorities in the ABE method, (4) Re-keying size: It indicates the size of the re-keying message that can be used to recognize the user revocation for each attribute in the ABE system, (5) Computation cost on the data owner: It indicates the required time to encrypt data by a data owner, and (6) Computation cost on the user: It shows the required time to decrypt data by a user. Table 3 depicts the performance comparison of some of the existing method based on the aforementioned parameters.

Table 4 compares the complexity of the existing ABE methods for cloud computing based on communication cost (size of ciphertext); storage cost (public and private key size); computation cost of data encryption and decryption on data owners and users; and communication and computation cost of adding a new user, deleting an existing user who leaves the group, or updating the list of users, which are incurred on the cloud service provider.

7. Open issues and challenges

Data access control is a very important issue, specifically in cloud computing since the data owners' delegate the management of data to an un-trusted party. This section highlights some of the most important challenges in deploying and utilizing the attribute-based data access control in cloud and distributed computing. Table 1 clearly shows the comparative analysis of the different ABE techniques to discover the advantages and disadvantages, the significance and requirements, and identifies the research gaps. The three main issues and open challenges for further investigation are deliberated as follows:

1. *Lightweight data auditing approach for cloud and mobile cloud computing:* Recently, the majority of businesses and enterprise consumers have outsourced their data to a remote cloud

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

Table 4
Comparison of computation, communication, and storage cost of the existing attribute-based access control methods for cloud computing.

Schemes	Commun. cost	Storage cost		Computation cost		Communication and computation cost on ABE center		
	Ciphertext size	Private key size	Public key size	DO	User	Adding user	Updating user	Deleting user
GPSW [14]	$O(n_c)$	$O(n_u)$	$O(m)$	$O(n_c)$	$O(n_c)$	$O(n_u)$	$O(n)$	$O(n \times n_u)$
BSW [31]	$O(n_c)$	$O(m)$	$O(l)$	$O(n_c)$	$O(n_c)$	$O(m)$	$O(n \times m)$	$O(n \times m)$
OSW [37]	$O(n_c)$	$O(n_u)$	$O(m)$	$O(n_c)$	$O(n_c + n_u)$	$O(n_u)$	$O(n \times n_u)$	$O(n \times n_u)$
ALP [40]	$O(1)$	$O(m \times n_u)$	$O(m)$	$O(m)$	$O(n_c)$	$O(m \times n_u)$	$O(n \times m \times n_u)$	$O(n \times m \times n_u)$
[42]	$O(n_c)$	$O(n_u)$	$O(m)$	$O(m)$	$O(n_u)$	$O(n_u)$	$O(n \times n_u)$	$O(n \times n_u)$
[63]	$O(n_c)$	$O(m)$	$O(m)$	$O(n_c)$	$O(n_a \times n_c)$	$O(m)$	$O(n)$	$O(n \times m)$
LCLS [64]	$O(n_a \times n_c)$	$O(n_a \times n_u)$	$O(m \times n_a)$	$O(m)$	$O(n_c \times n_u)$	$O(n_a \times n_u)$	$O(n \times n_a \times n_u)$	$O(n \times n_a \times n_u)$
CC [67]	$O(n_c)$	$O(m)$	$O\left(m^2\right)$	$O(n_c)$	$O(n_c)$	$O\left(m^2\right)$	$O\left(n \times m^2\right)$	$O\left(n \times m^2\right)$
LW [68]	$O(n_c)$	$O(n_u)$	$O(m)\ O(n_c)$	$O(n_c)$	$O(n_u)$	$O(n \times n_u)$		$O(n \times n_u)$

n: the number of all users in the system, n_c: the number of attributes associated with a ciphertext, m: the number of attributes in the system, n_u: the number of users attribute appeared in the system, n_a: the number of authorities.

storage server. To access the outsourced data, they mainly use the resource-constrained mobile devices [57]. A significant challenge in cloud and mobile cloud computing environment is to develop a lightweight access control method to improve the security of the outsourced data without any further limitation and requirement. The lightweight data access control must be subjected to minimize the number of transmissions and computation overhead. Moreover, another important issue is to provide the flexibility. The lightweight method needs to support the fine-grained access control from multi-users. One of the important solutions with respect to lightweight access control in cloud and distributed computing is to employ the computation delegation methods, such as [88–90]. By Employing this technique, the user may be able to reduce the computation cost and delegate the huge part of computation to the trusted third party in cloud computing.

2. *User revocation and attribute revocation:* The ABE enables data owners to encrypt the data based on a set of their attributes before outsourcing the data to the cloud. The ABE method needs to support the user revocation in which the data owner is able to revoke a specific user. Moreover, attribute revocation characteristic allows the data owner to revoke a group of users who already have the revoked attribute. However, the majority of the decentralized and hierarchical data access schemes are not able to support the attribute revocation. The lazy revocation is an applicable technique to achieve this goal with minimum computation and communication overhead on the client and server side and is applicable in cloud and distributed computing [58].
3. *Backward and forward security:* The users who newly join the system must not be able to decrypt the previously published plaintext, although they have sufficient attributes to decrypt such plaintext. On the other hand, when a specific attribute is revoked, a group of users who had such attribute must be unable to access any new plaintext and decrypt any new ciphertext by using the dropped attribute. These are two fundamental issues for proposing an ABE scheme, which are called backward and forward issue, respectively. Although there are some methods to prevent forward issue in centralized ABE [63,64], most of the decentralized and hierarchical ABE methods are not yet able to prevent the forward and backward issues and therefore these issues have not been addressed yet.

8. Conclusion

This paper thoroughly discussed access control systems and a wide range of attribute-based access control mechanisms applied in cloud and distributed computing. We began by presenting the concept of IBE and explaining the PRE and RBAC as three fundamental cryptographic techniques to provide a background for attribute-based access control systems. The paper explored ABE as a new type of IBE scheme. Moreover, we comprehensively surveyed the state-of-the-art attribute-based access control schemes to devise a thematic taxonomy on the basis of several parameters and characteristics. Furthermore, we classified the common frameworks and highlight the similarities and differences of different ABE techniques. We analyzed ABE schemes to discover some of the advantages and disadvantages, the significance and requirements, and identified the research gaps. The paper also drew attention to open issues and challenges more specifically in ABE schemes for further investigations.

Acknowledgment

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

References

[1] P. Mell, T. Grance, The NIST definition of cloud computing, Commun. ACM 53 (6) (2010) 1–145.

[2] Z. Zhibin, H. Dijiang, Efficient and secure data storage operations for mobile cloud computing, in: 8th International Conference and Workshop on Systems Virtualiztion Management Network and Service Management, 2012, pp. 37–45.

[3] M. Sookhak, H. Talebian, E. Ahmed, A. Gani, M. Khan, A review on remote data auditing in single cloud server: Taxonomy and open issues, J. Netw. Comput. Appl. 43 (2014) 121–141. http://dx.doi.org/10.1016/j.jnca.2014.04.011.

[4] K. Kumar, L. Yung-Hsiang, Cloud Computing for Mobile Users: Can Offloading Computation Save Energy? Computer 43 (4) (2010) 51–56. http://dx.doi.org/10.1109/mc.2010.98.

[5] M. Sookhak, A. Akhunzada, A. Gani, M. Khurram Khan, N. Anuar, Towards dynamic remote data auditing in computational clouds, Sci. World J. (2014) http://dx.doi.org/10.1155/2014/269357.

[6] M. Whaiduzzaman, M. Sookhak, A. Gani, R. Buyya, A survey on vehicular cloud computing, J. Netw. Comput. Appl. 40 (1) (2014) 325–344. http://dx.doi.org/10.1016/j.jnca.2013.08.004.

[7] M. Sookhak, A. Gani, M.K. Khan, R. Buyya, Dynamic remote data auditing for securing big data storage in cloud computing, Inform. Sci. (2015) 1–16. http://dx.doi.org/10.1016/j.ins.2015.09.004.

[8] Q.A. Wang, C. Wang, K. Ren, W.J. Lou, J. Li, Enabling public auditability and data dynamics for storage security in cloud computing, IEEE Trans. Parallel Distrib. Syst. 22 (5) (2011) 847–859. http://dx.doi.org/10.1109/Tpds.2010.183.

[9] M. Sookhak, A. Gani, H. Talebian, A. Akhunzada, S.U. Khan, R. Buyya, A.Y. Zomaya, Remote data auditing in cloud computing environments: A survey, taxonomy, and open issues, ACM Comput. Surv. 47 (4) (2015) 65:1–65:34. http://dx.doi.org/10.1145/2764465.

[10] W. Cong, R. Kui, L. Wenjing, L. Jin, Toward publicly auditable secure cloud data storage services, IEEE Netw. 24 (4) (2010) 19–24. http://dx.doi.org/10.1109/MNET.2010.5510914.

[11] M. Shiraz, M. Sookhak, A. Gani, S. Shah, A study on the critical analysis of computational offloading frameworks for mobile cloud computing, J. Netw. Comput. Appl. 47 (2015) 47–60. http://dx.doi.org/10.1016/j.jnca.2014.08.011.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

[12] S.D.C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, P. Samarati, A data outsourcing architecture combining cryptography and access control, 2007, http://dx.doi.org/10.1145/1314466.1314477.

[13] A. Sahai, B. Waters, Fuzzy identity-based encryption, (EUROCRYPT 2005), in: Advances in Cryptology, vol. 3494, Springer Berlin, Heidelberg, 2005, pp. 457–473. http://dx.doi.org/10.1007/11426639_27, (Chapter 27).

[14] V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, 2006. http://dx.doi.org/10.1145/1180405.1180418.

[15] J. Hur, Improving security and efficiency in attribute-based data sharing, IEEE Trans. Knowl. Data Eng. 25 (10) (2013) 2271–2282. http://dx.doi.org/10.1109/TKDE.2011.78.

[16] A. Shamir, Identity-based cryptosystems and signature schemes, in: G. Blakley, D. Chaum (Eds.), Advances in Cryptology, Vol. 196, Springer Berlin, Heidelberg, 1985, pp. 47–53. http://dx.doi.org/10.1007/3-540-39568-7_5, (Chapter 5).

[17] D. Boneh, M. Franklin, Identity-based encryption from the weil pairing, in: J. Kilian (Ed.), (CRYPTO 2001), in: Advances in Cryptology, vol. 2139, Springer Berlin, Heidelberg, 2001, pp. 213–229. http://dx.doi.org/10.1007/3-540-44647-8_13, (Chapter 13).

[18] J. Shao, Z. Cao, Multi-use unidirectional identity-based proxy re-encryption from hierarchical identity-based encryption, Inform. Sci. 206 (2012) 83–95. http://dx.doi.org/10.1016/j.ins.2012.04.013.

[19] M. Blaze, G. Bleumer, M. Strauss, Divertible protocols and atomic proxy cryptography, in: Advances in Cryptology EUROCRYPT'9, Vol. 1403, Springer Berlin, Heidelberg, 1998, pp. 127–144. http://dx.doi.org/10.1007/BFb0054122, (Chapter 10).

[20] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, in: G. Blakley, D. Chaum (Eds.), Advances in Cryptology, Vol. 196, Springer Berlin, Heidelberg, 1985, pp. 10–18. http://dx.doi.org/10.1007/3-540-39568-7_2, (Chapter 2).

[21] A. Ivan, Y. Dodis, Proxy cryptography revisited, in: Proceedings of the Network and Distributed System Security Symposium, NDSS, 2003, pp. 1–20.

[22] G. Ateniese, K. Fu, M. Green, S. Hohenberger, Improved proxy re-encryption schemes with applications to secure distributed storage, ACM Trans. Inf. Syst. Secur. 9 (1) (2006) 1–30. http://dx.doi.org/10.1145/1127345.1127346.

[23] R. Canetti, S. Hohenberger, Chosen-ciphertext secure proxy re-encryption, 2007. http://dx.doi.org/10.1145/1315245.1315269.

[24] B. Libert, D. Vergnaud, Unidirectional chosen-ciphertext secure proxy re-encryption, IEEE Trans. Inform. Theory 57 (3) (2011) 1786–1802. http://dx.doi.org/10.1109/TIT.2011.2104470.

[25] H. Wang, Z. Cao, L. Wang, Multi-use and unidirectional identity-based proxy re-encryption schemes, Inform. Sci. 180 (20) (2010) 4042–4059. http://dx.doi.org/10.1016/j.ins.2010.06.029.

[26] R.S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman, Role-based access control models, IEEE Comput. 29 (2) (1996) 38–47. http://dx.doi.org/10.1109/2.485845.

[27] G.-J. Ahn, R. Sandhu, Role-based authorization constraints specification, ACM Trans. Inf. Syst. Secur. 3 (4) (2000) 207–226. http://dx.doi.org/10.1145/382912.382913.

[28] J. Joshi, A. Ghafoor, W.G. Aref, E.H. Spafford, Digital government security infrastructure design challenges, IEEE Comput. 34 (2) (2001) 66–72. http://dx.doi.org/10.1109/2.901169.

[29] J.B.D. Joshi, W.G. Aref, A. Ghafoor, E.H. Spafford, Security models for web-based applications, Commun. ACM 44 (2) (2001) 38–44. http://dx.doi.org/10.1145/359205.359224.

[30] D.F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, R. Chandramouli, Proposed NIST standard for role-based access control, ACM Trans. Inf. Syst. Secur. 4 (3) (2001) 224–274. http://dx.doi.org/10.1145/501978.501980.

[31] D. Jeong-Min, S. You-Jin, P. Namje, Attribute based proxy re-encryption for data confidentiality in cloud computing environments, in: 2011 First ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering, CNSI, 2011, pp. 248–251. http://dx.doi.org/10.1109/CNSI.2011.34.

[32] J. Bethencourt, A. Sahai, B. Waters, Ciphertext-Policy Attribute-Based Encryption, 2007. http://dx.doi.org/10.1109/SP.2007.11.

[33] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, W. Jonker, Mediated ciphertext-policy attribute-based encryption and its application, in: Information Security Applications, Vol. 5932, Springer Berlin, Heidelberg, 2009, pp. 309–323. http://dx.doi.org/10.1007/978-3-642-10838-9_23, (Chapter 23).

[34] S. Yu, C. Wan, K. Ren, W. Lou, Attribute based data sharing with attribute revocation, 2010. http://dx.doi.org/10.1145/1755688.1755720.

[35] M. Pirretti, P. Traynor, P. McDaniel, B. Waters, Secure attribute-based systems, J. Comput. Secur. 18 (5) (2010) 799–837. http://dx.doi.org/10.3233/JCS-2009-0383.

[36] X. Liang, R. Lu, X. Lin, X.S. Shen, Ciphertext policy attribute based encryption with efficient revocation, Tech. Rep., 2010, URL http://bbcr.uwaterloo.ca/~x27liang/papers/abewithrevocation.pdf.

[37] M. Green, S. Hohenberger, B. Waters, Outsourcing the decryption of ABE ciphertexts, 2011.

[38] J. Han, W. Susilo, Y. Mu, J. Zhou, M. Au, PPDCP-ABE: Privacy-preserving decentralized ciphertext-policy attribute-based encryption, in: Computer Security - ESORICS 2014, Vol. 8713, Springer International Publishing, 2014, pp. 73–90. http://dx.doi.org/10.1007/978-3-319-11212-1_5, (Chapter 5).

[39] D.R. Stinson, Cryptography: Theory and Practice, third ed., Chapman and Hall/CRC, 2005.

[40] R. Ostrovsky, A. Sahai, B. Waters, Attribute-based encryption with non-monotonic access structures, 2007. http://dx.doi.org/10.1145/1315245.1315270.

[41] M. Naor, B. Pinkas, Efficient trace and revoke schemes, in: Financial Cryptography, Vol. 1962, Springer Berlin, Heidelberg, 2001, pp. 1–20. http://dx.doi.org/10.1007/3-540-45472-1_1, (Chapter 1).

[42] A. Lewko, A. Sahai, B. Waters, Revocation systems with very small private keys, in: IEEE Symposium on Security and Privacy, SP, Oakland, CA, USA, 2010, pp. 273–285. http://dx.doi.org/10.1109/SP.2010.23.

[43] N. Attrapadung, B. Libert, E. de Panafieu, Expressive key-policy attribute-based encryption with constant-size ciphertexts, PKC 2011, in: Public Key Cryptography, vol. 6571, Springer Berlin, Heidelberg, 2011, pp. 90–108. http://dx.doi.org/10.1007/978-3-642-19379-8_6, (Chapter 6).

[44] L. Cheung, C. Newport, Provably secure ciphertext policy ABE, 2007. http://dx.doi.org/10.1145/1315245.1315302.

[45] B. Waters, Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization, PKC 2011, in: Public Key Cryptography, vol. 6571, Springer Berlin, Heidelberg, 2011, pp. 53–70. http://dx.doi.org/10.1007/978-3-642-19379-8_4, (Chapter 4).

[46] S. Yamada, N. Attrapadung, G. Hanaoka, N. Kunihiro, A framework and compact constructions for non-monotonic attribute-based encryption, PKC 2014, in: Public-Key Cryptography, vol. 8383, Springer Berlin, Heidelberg, 2014, pp. 275–292. http://dx.doi.org/10.1007/978-3-642-54631-0_16, (Chapter 16).

[47] S. Garg, C. Gentry, S. Halevi, A. Sahai, B. Waters, Attribute-based encryption for circuits from multilinear maps, in: Advances in Cryptology CRYPTO 2013, vol. 8043, Springer, Berlin, Heidelberg, 2013, pp. 479–499. http://dx.doi.org/10.1007/978-3-642-40084-1_27, (Chapter 27).

[48] D. Boneh, A. Silverberg, Applications of multilinear forms to cryptography, Contemp. Math. 324 (2003) 72–91.

[49] S. Gorbunov, V. Vaikuntanathan, H. Wee, Attribute-based encryption for circuits, 2013. http://dx.doi.org/10.1145/2488608.2488677.

[50] C.C. Dragan, F.L. Tiplea, Efficient key-policy attribute-based encryption for general Boolean circuits from multilinear maps, Tech. rep., Cryptology ePrint Archive, Report 2014/462, 2014.

[51] M. Pirretti, P. Traynor, P. McDaniel, B. Waters, Secure attribute-based systems, 2006. http://dx.doi.org/10.1145/1180405.1180419.

[52] A. Boldyreva, V. Goyal, V. Kumar, Identity-based encryption with efficient revocation, 2008. http://dx.doi.org/10.1145/1455770.1455823.

[53] J. Hur, D.K. Noh, Attribute-based access control with efficient revocation in data outsourcing systems, IEEE Trans. Parallel Distrib. Syst. 22 (7) (2011) 1214–1221. http://dx.doi.org/10.1109/TPDS.2010.203.

[54] K. Yang, X. Jia, R. Kui, Attribute-based fine-grained access control with efficient revocation in cloud storage systems, 2013. http://dx.doi.org/10.1145/2484313.2484383.

[55] Y. Cheng, Z.-y. Wang, J. Ma, J.-j. Wu, S.-z. Mei, J.-c. Ren, Efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage, J. Zhejiang Univ. Sci. C 14 (2) (2013) 85–97. http://dx.doi.org/10.1631/jzus.C1200240.

[56] A. Shamir, How to share a secret, Commun. ACM 22 (11) (1979) 612–613. http://dx.doi.org/10.1145/359168.359176.

[57] P.K. Tysowski, M.A. Hasan, Hybrid attribute- and re-encryption-based key management for secure and scalable mobile applications in clouds, IEEE Trans. Cloud Comput. 1 (2) (2013) 172–186. http://dx.doi.org/10.1109/TCC.2013.11.

[58] K.E. Fu, Group sharing and random access in cryptographic storage file systems (Ph.D. thesis), 1999.

[59] K. Fu, S. Kamara, T. Kohno, Key regression: Enabling efficient key distribution for secure distributed storage, Computer Science Department Faculty Publication Series, 2006, p. 149.

[60] M. Backes, C. Cachin, A. Oprea, Secure key-updating for lazy revocation, (ESORICS 2006), in: 1th European Symposium on Research in Computer Security, vol. 4189, Springer, Berlin, Heidelberg, Hamburg, Germany, 2006, pp. 327–346. http://dx.doi.org/10.1007/11863908_21.

[61] S. Zarandioon, D. Yao, V. Ganapathy, K2C: Cryptographic cloud storage with lazy revocation and anonymous access, (SecureComm 2011), in: 7th International ICST Conference, vol. 96, Springer, Berlin, Heidelberg, London, UK, 2011, pp. 59–76. http://dx.doi.org/10.1007/978-3-642-31909-9_4.

[62] A. Barsoum, A. Hasan, Enabling dynamic data and indirect mutual trust for cloud computing storage systems, IEEE Trans. Parallel Distrib. Syst. 24 (12) (2013) 2375–2385. http://dx.doi.org/10.1109/TPDS.2012.337.

[63] A. Sahai, H. Seyalioglu, B. Waters, Dynamic credentials and ciphertext delegation for attribute-based encryption, in: Advances in Cryptology CRYPTO 2012, vol. 7417, Springer, Berlin, Heidelberg, 2012, pp. 199–217. http://dx.doi.org/10.1007/978-3-642-32009-5_13, (Chapter 13).

[64] K. Lee, S. Choi, D. Lee, J. Park, M. Yung, Self-updatable encryption: Time constrained access control with hidden attributes and better efficiency, in: K. Sako, P. Sarkar (Eds.), Advances in Cryptology - ASIACRYPT 2013, Vol. 8269, Springer, Berlin, Heidelberg, 2013, pp. 235–254. http://dx.doi.org/10.1007/978-3-642-42033-7_13, (Chapter 13).

[65] M. Chase, Multi-authority attribute based encryption, in: Theory of Cryptography, Vol. 4392, Springer, Berlin, Heidelberg, 2007, pp. 515–534. http://dx.doi.org/10.1007/978-3-540-70936-7_28, (Chapter 28).

[66] H. Lin, Z. Cao, X. Liang, J. Shao, Secure threshold multi authority attribute based encryption without a central authority, Inform. Sci. 180 (13) (2010) 2618–2632. http://dx.doi.org/10.1016/j.ins.2010.03.004.

[67] R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, Secure distributed key generation for discrete-log based cryptosystems, J. Cryptol. 20 (1) (2007) 51–83. http://dx.doi.org/10.1007/s00145-006-0347-3.

[68] R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, Robust threshold DSS signatures, Inform. Comput. 164 (1) (2001) 54–84. http://dx.doi.org/10.1006/inco.2000.2881.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

[69] M. Chase, S.S.M. Chow, Improving privacy and security in multi-authority attribute-based encryption, 2009. http://dx.doi.org/10.1145/1653662.1653678.

[70] A. Lewko, B. Waters, Decentralizing attribute-based encryption, in: Advances in Cryptology EUROCRYPT 2011, Vol. 6632, Springer, Berlin, Heidelberg, 2011, pp. 568–588. http://dx.doi.org/10.1007/978-3-642-20465-4_31, (Chapter 31).

[71] S. Ruj, A. Nayak, I. Stojmenovic, DACC: Distributed access control in clouds, in: IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Changsha, 2011, pp. 91–98. http://dx.doi.org/10.1109/TrustCom.2011.15.

[72] K. Yang, X. Jia, K. Ren, B. Zhang, R. Xie, DAC-MACS: Effective data access control for multiauthority cloud storage systems, IEEE Trans. Inf. Forensics Secur. 8 (11) (2013) 1790–1801. http://dx.doi.org/10.1109/TIFS.2013.2279531.

[73] K. Yang, X. Jia, Expressive, efficient and revocable data access control for multi-authority cloud storage, IEEE Trans. Parallel Distrib. Syst. PP (99) (2013) 1. http://dx.doi.org/10.1109/TPDS.2013.253.

[74] A. Lewko, B. Waters, New proof methods for attribute-based encryption: Achieving full security through selective techniques, in: Advances in Cryptology CRYPTO 2012, Vol. 7417, Springer, Berlin, Heidelberg, 2012, pp. 180–198. http://dx.doi.org/10.1007/978-3-642-32009-5_12, (Chapter 12).

[75] S. Ruj, M. Stojmenovic, A. Nayak, Decentralized access control with anonymous authentication of data stored in clouds, IEEE Trans. Parallel Distrib. Syst. 25 (2) (2014) 384–394. http://dx.doi.org/10.1109/TPDS.2013.38.

[76] H. Maji, M. Prabhakaran, M. Rosulek, Attribute-based signatures, in: Topics in Cryptology CT-RSA 2011, Vol. 6558, Springer, Berlin, Heidelberg, 2011, pp. 376–392. http://dx.doi.org/10.1007/978-3-642-19074-2_24, (Chapter 24).

[77] C. Gentry, A. Silverberg, Hierarchical ID-based cryptography, (ASIACRYPT 2002), in: 8th International Conference on the Theory and Application of Cryptology and Information Security Queenstown, vol. 2501, Springer Berlin, Heidelberg, New Zealand, 2002, pp. 548–566. http://dx.doi.org/10.1007/3-540-36178-2_34.

[78] G. Wang, Q. Liu, J. Wu, M. Guo, Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers, Comput. Secur. 30 (5) (2011) 320–331. http://dx.doi.org/10.1016/j.cose.2011.05.006.

[79] K. Bennett, C. Grothoff, T. Horozov, I. Patrascu, Efficient sharing of encrypted data, in: Information Security and Privacy, Vol. 2384, Springer, Berlin, Heidelberg, 2002, pp. 107–120. http://dx.doi.org/10.1007/3-540-45450-0_8, (Chapter 8).

[80] W. Zhiguo, L. Jun'e, R.H. Deng, HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing, IEEE Trans. Inf. Forensics Secur. 7 (2) (2012) 743–754. http://dx.doi.org/10.1109/TIFS.2011.2172209.

[81] R. Bobba, H. Khurana, M. Prabhakaran, Attribute-sets: A practically motivated enhancement to attribute-based encryption, (ESORICS 2009), in: 14th European Symposium on Research in Computer Security, vol. 5789, Springer Berlin, Heidelberg, Saint-Malo, France, 2009, pp. 587–604. http://dx.doi.org/10.1007/978-3-642-04444-1_36, (Chapter 36).

[82] J. Li, Q. Wang, C. Wang, K. Ren, Enhancing attribute-based encryption with attribute hierarchy, Mob. Netw. Appl. 16 (5) (2011) 553–561. http://dx.doi.org/10.1007/s11036-010-0233-y.

[83] X. Liu, Y. Xia, S. Jiang, F. Xia, Y. Wang, Hierarchical attribute-based access control with authentication for outsourced data in cloud computing, in: 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Melbourne, VIC, 2013, pp. 477–484. http://dx.doi.org/10.1109/TrustCom.2013.60.

[84] Z. Wang, J. Wang, A provably secure ciphertext-policy hierarchical attribute-based encryption, (ICCCS 2015), in: Cloud Computing and Security: First International Conference, Springer International Publishing, Nanjing, China, 2015, pp. 38–48. http://dx.doi.org/10.1007/978-3-319-27051-7_4.

[85] I. Damgård, R. Thorbek, Linear integer secret sharing and distributed exponentiation, in: 9th International Conference on Theory and Practice in Public-Key Cryptography, Springer Berlin, Heidelberg, New York, 2006, pp. 75–90. http://dx.doi.org/10.1007/11745853_6.

[86] W. Teng, G. Yang, Y. Xiang, T. Zhang, D. Wang, Attribute-based access control with constant-size ciphertext in cloud computing, IEEE Trans. Cloud Comput. PP (99) (2016) 1. http://dx.doi.org/10.1109/TCC.2015.2440247.

[87] S. Wang, J. Zhou, J.K. Liu, J. Yu, J. Chen, W. Xie, An efficient file hierarchy attribute-based encryption scheme in cloud computing, IEEE Trans. Inf. Forensics Secur. 11 (6) (2016) 1265–1277. http://dx.doi.org/10.1109/TIFS.2016.2523941.

[88] L. Xu, S. Tang, Verifiable computation with access control in cloud computing, J. Supercomput. (2013) 1–19. http://dx.doi.org/10.1007/s11227-013-1039-z.

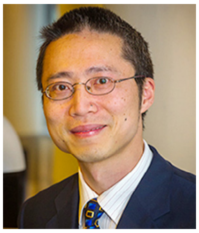
[89] B. Parno, M. Raykova, V. Vaikuntanathan, How to delegate and verify in public: Verifiable computation from attribute-based encryption, Springer, Berlin, Heidelberg, 2012, pp. 422–439. http://dx.doi.org/10.1007/978-3-642-28914-9_24, (Chapter 24).

[90] G. Xu, G. Amariucai, Y. Guan, Delegation of computation with verification outsourcing: curious verifiers, 2013. http://dx.doi.org/10.1145/2484239.2484253.

Mehdi Sookhak is a postdoctoral Research fellow at Carleton University of Canada funded by Canadian Natural Sciences and Engineering Research Council (NSERC). He was an active researcher in Centre of Mobile Cloud Computing Research (C4MCCR) at Faculty of Computer Science and Information Technology, University Malaya, Kuala Lumpur. His areas of interest include Cryptography and Information Security, Cloud Computing, Computation outsourcing, Access control, Wireless Sensor & Mobile Ad Hoc Network (Architectures, Protocols, Security, and Algorithms), and Distributed Systems.

F. Richard Yu (S'00–M'04–SM'08) received the Ph.D. degree in electrical engineering from the University of British Columbia (UBC) in 2003. From 2002 to 2004, he was with Ericsson (in Lund, Sweden), where he worked on the research and development of wireless mobile systems. From 2005 to 2006, he was with a start-up in California, USA, where he worked on the research and development in the areas of advanced wireless communication technologies and new standards. He joined Carleton School of Information Technology and the Department of Systems and Computer Engineering at Carleton University in 2007, where he is currently an Associate Professor. He received the IEEE Outstanding Leadership Award in 2013, Carleton Research Achievement Award in 2012, the Ontario Early Researcher Award (formerly Premier's Research Excellence Award) in 2011, the Excellent Contribution Award at IEEE/IFIP TrustCom 2010, the Leadership Opportunity Fund Award from Canada Foundation of Innovation in 2009 and the Best Paper Awards at IEEE ICC 2014, Globecom 2012, IEEE/IFIP TrustCom 2009 and Int'l Conference on Networking 2005. His research interests include cross-layer/cross-system design, security, green IT and QoS provisioning in wireless-based systems. He serves on the editorial boards of several journals, including Co-Editor-in-Chief for Ad Hoc & Sensor Wireless Networks, Lead Series Editor for IEEE Transactions on Vehicular Technology, IEEE Communications Surveys & Tutorials, EURASIP Journal on Wireless Communications Networking, Wiley Journal on Security and Communication Networks, and International Journal of Wireless Communications and Networking, a Guest Editor for IEEE Transactions on Emerging Topics in Computing special issue on Advances in Mobile Cloud Computing, and a Guest Editor for IEEE Systems Journal for the special issue on Smart Grid Communications Systems. Dr. Yu is a registered Professional Engineer in the province of Ontario, Canada.

Muhammad Khurram Khan is currently working as an associate professor and R&D Manager at the Center of Excellence in Information Assurance (CoEIA), King Saud University, Kingdom of Saudi Arabia. Dr. Khurram has published hundreds of research papers, received many awards. His areas of research interest are biometrics, multimedia security, digital data hiding, and authentication protocols.



Yang Xiang received the Ph.D. degree in computer science from Deakin University, Australia. He is currently a full professor at the School of Information Technology, Deakin University. He is the director of the Network Security and Computing Lab (NSCLab). His research interests include network and system security, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks. He is the chief investigator of several projects in network and system security, funded by the Australian Research Council (ARC). He has published more than 130 research papers in many international journals and conferences, such as IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Information Security and Forensics, and IEEE Journal on Selected Areas in Communications. Two of his papers were selected as the featured articles in the April 2009 and the July 2013 issues of IEEE Transactions on Parallel and Distributed Systems. He has published two books, Software Similarity and Classification (Springer) and Dynamic and Advanced Data Mining for Progressing Technological Development (IGI-Global). He has served as the program/general chair for many international conferences such as ICA3PP 12/11, IEEE/IFIP EUC 11, IEEE TrustCom 13/11, IEEE HPCC 10/09, IEEE ICPADS 08, NSS 11/10/09/08/07. He has been the PC member for more than 60 international conferences in distributed systems, networking, and security. He serves as the associate editor of the IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, Security and Communication Networks (Wiley), and the Editor of Journal of Network and Computer Applications. He is the coordinator, Asia for IEEE Computer Society Technical Committee on Distributed Processing (TCDP). He is a senior member of the IEEE.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018

Rajkumar Buyya is a Fellow of IEEE, Professor of Computer Science and Software Engineering, Future Fellow of the Australian Research Council, and Director of the Cloud Computing and Distributed Systems (CLOUDS) Laboratory at the University of Melbourne, Australia. He is also serving as the founding CEO of Manjrasoft, a spin-off company of the University, commercializing its innovations in Cloud Computing. He has authored over 450 publications and five text books including "Mastering Cloud Computing" published by McGraw Hill, Elsevier/Morgan Kaufmann, and China Machine Press for Indian, American, and Chinese markets respectively. He is one of the highly cited authors in computer science and software engineering worldwide (h-index = 88, g-index = 176, 38,700+ citations). Software technologies for Grid and Cloud computing developed under Dr. Buyya's leadership have gained rapid acceptance and are in use at several academic institutions and commercial enterprises in 40 countries around the world. Dr. Buyya has led the establishment and development of key community activities, including serving as foundation Chair of the IEEE Technical Committee on Scalable Computing and five IEEE/ACM conferences. These contributions and international research leadership of Dr. Buyya are recognized through the award of "2009 IEEE TCSC Medal for Excellence in Scalable Computing". Manjrasoft's Aneka Cloud technology developed under his leadership has received "2010 Frost & Sullivan New Product Innovation Award" and recently Manjrasoft has been recognized as one of the Top 20 Cloud Computing companies by the Silicon Review Magazine. He is currently serving as Co-Editor-in-Chief of Journal of Software: Practice and Experience, which was established 40+ years ago. For further information on Dr. Buyya, please visit his cyberhome: www.buyya.com.

Please cite this article in press as: M. Sookhak, et al., Attribute-based data access control in mobile cloud computing: Taxonomy and open issues, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.08.018