© 2017 IJSRSET | Volume 3 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Improving Privacy & Security in PPMA -ABE

M. Chinnasamy[1], S. Gandhimathi[2], Dr. V. Baby Deepa[3]

[1]Computer Science & Application, PGP College of Arts & Science, Namakkal, Tamilnadu, India
[2]Asst. Prof & HOD, Computer Science & Application, PGP college of Arts & Science, Namakkal, Tamilnadu, India
[3]Asst.Prof in Research Department of Computer Science, Govt. Arts College (Autonomous), karur-5, Tamilnadu, India

ABSTRACT

In previous privacy-preserving multiauthority attribute-based encryption (PPMA-ABE) schemes, a user can acquire secret keys from multiple authorities with them knowing his/her attributes and furthermore, a central authority is required. Notably, a user's identity information can be extracted from his/her some sensitive attributes. Hence, existing PPMA-ABE schemes cannot fully protect users' privacy as multiple authorities can collaborate to identify a user by collecting and analyzing his attributes. Moreover, ciphertext-policy ABE (CP-ABE) is a more efficient public-key encryption, where the encryptor can select flexible access structures to encrypt messages. Therefore, a challenging and important work is to construct a PPMA-ABE scheme where there is no necessity of having the central authority and furthermore, both the identifiers and the attributes can be protected to be known by the authorities. In this paper, a privacy-preserving decentralized CP-ABE (PPDCP-ABE) is proposed to reduce the trust on the central authority and protect users' privacy. In our PPDCP-ABE scheme, each authority can work independently without any collaboration to initial the system and issue secret keys to users. Furthermore, a user can obtain secret keys from multiple authorities without them knowing anything about his global identifier and attributes.

Keywords : CP-ABE, Decentralization, Privacy.

I. INTRODUCTION

In network society, attributes are used to distinguish different users. For instance, European electronic identity cards often comprise the attributes: nationality, sex, civil status, hair and eye colour, and applicable minority status. These attributes can be either binary or discrete numbers from pre-defined finite sets.

In particular, these attributes are required to selectively disclose as they are privacy-sensitive otherwise, a user can be identified and impersonated if some of his/her sensitive attributes are collected.

Attribute-based encryption (ABE)

In practice, we often want to share data with some expressive attributes and do not know who the recipient will be. To resolve this problem, a new public-key encryption system called attribute-based encryption (ABE) was introduced in the seminal work of Sahai and Waters.

In an ABE scheme, there is a central authority who monitors a set of universal attributes and issues secret keys to users accordingly. As a result, a user can decrypt a ciphertext if and only if there is a match between the attributes which are listed in the ciphertext and the attributes which he holds. ABE schemes have been the primary focus in the research community nowadays as it allows flexible access control and can protect the confidentiality of sensitive data.

Multi-Authority ABE (MA-ABE)

In an ABE scheme, a central authority is required. To reduce the trust on the central authority, Chase proposed a multi-authority ABE (MA-ABE) scheme. In this scheme, multiple authorities can co-exist and must cooperate with the central authority to initialize the system.

In the work, Sahai andWaters left an open problem, namely how to construct an ABE scheme where the secret key can be obtained from multiple authorities so that users can reduce the trust on the central authority.

IJSRSET1736192 | Received : 10 Oct 2017 | Accepted : 25 Oct 2017 | September-October-2017 [(3)6: 767-778]

Chase answered this question affirmatively by proposing an MA-ABE scheme. The technical hurdle in designing an MA-ABE scheme is to resist the collusion attacks. To overcome this hurdle, GID was introduced to tie all the user's secret keys together. In there is a central authority, and multiple authorities must interact to initialize the system.

Decentralized CP-ABE (DCP-ABE)

Then, Lewko and Waters proposed a decentralized CP-ABE (DCP-ABE) where a central authority is not required and multiple authorities can work independently without any cooperation. This scheme improved the previous MA-ABE schemes that require collaborations among multiple authorities to conduct the system setup. In this scheme, no cooperation between the multiple authorities is required in the setup stage and the key generation stage, and there is no central authority.

Notably, an authority in this scheme can join or leave the system freely without reinitializing the system. The scheme was constructed in the composite order (N = p1p2p3) bilinear group, and achieves full (adaptive) security in the random oracle model. They also pointed out two methods to create a prime order group variant of their scheme. Nevertheless, the authorities can collect a user's attributes by tracing his GID.

Previous Privacy-Preserving MA-ABE (PPMA-ABE)

Since the authorities can impersonate a user if they can know his attributes, privacy issues in MA-ABE are the primary concern of users. Considering this issue, some schemes have been proposed, but they cannot provide a complete solution.

In all the previous privacy-preserving MA-ABE (PPMA-ABE) schemes, only the privacy of the global identifier (GID) has been considered. Currently, no scheme addressing the privacy of the attributes in MA-ABE has been proposed. However, it is extremely important as a user can be identified by some sensitive attributes. To clarify this, we give the following example. Suppose that the Head of the Department of Computer Science is Bob.

Two sets of attributes S1={Position="Head", Department="CS", Sex="Male"} and S2={Position="PhD Student", Department="S", Sex="Male"}, we can guess that S1 is Bob's attributes even if we do not know his GID. This clearly shows that it is necessary to control the release of sensitive attributes.

GLOBALLY UNIQUE IDENTIFIER

A globally unique identifier is a unique reference number used as an identifier in computer software. The term "GUID" typically refers to various implementations of the universally unique identifier (UUID) standard. GUIDs are usually stored as 128-bit values, and are commonly displayed as 32 hexadecimal digits with groups separated by hyphens, such as:

21EC2020-3AEA-4069-A2DD-08002B30309D

They may or may not be generated from random numbers. GUIDs generated from random numbers normally contain 6 fixed bits (these indicate that the GUID is random) and 122 random bits; the total number of unique such GUIDs is 2^{122} (approximately 5.3×10^{36}). This number is so large that the probability of the same number being generated randomly twice is negligible; however other GUID versions have different uniqueness properties and probabilities, ranging from guaranteed uniqueness to likely duplicates

II. LITERATURE SURVEY

ATTRIBUTE-BASED ENCRYPTION

Sahai and Waters introduced the first attribute-based encryption (ABE) where both the ciphertext and the secret key are labeled with a set of attributes. A user can decrypt a ciphertext if and only if there is a match between the attributes listed in the ciphertext and the attributes held by him. ABE schemes can be classified into two types: key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE).

KP-ABE: In a KP-ABE scheme, the ciphertext is associated with a set of attributes, while an access structure is embedded in the secret keys.

CP-ABE: In a CP-ABE scheme, an access structure is embedded in the ciphertext, while the secret keys are associated with a set of attributes.

MULTI-AUTHORITY ATTRIBUTE-BASED ENCRYPTION

In the seminal work, Sahai and Waters left an open problem, namely how to construct an ABE scheme where the secret keys can be extracted from multiple authorities so that users can reduce the trust on the central authority. Chase answered this question affirmatively by proposing an MA-ABE scheme. As mentioned in, the technical hurdle in constructing an MA-ABE scheme is to resist the collusion attacks. To overcome this hurdle, all secret keys of a user are tied to his GID. Multiple authorities must interact to initialize the system, and a central authority is required.

Lin et al. proposed an MA-ABE scheme where the central authority is not required. This scheme was derived from the distributed key generation (DKG) protocol and the joint zero secret sharing (JZSS) protocol. To initialize the system, the multiple authorities must collaboratively execute the DKG protocol and the JZSS protocol twice and k times, respectively, where k is the degree of the polynomial selected by each authority.

Each authority must keep k+2 secret keys. Furthermore, this scheme is k-resilient, namely the scheme is secure if and only if the number of the compromised users is no more than k, and k must be fixed in the setup stage.

Muller et al. proposed a distributed CP-ABE scheme. This scheme was proven to be secure in the generic group, instead of reducing to a complexity assumption. In this scheme, a central authority is required to generate the global key and issue secret keys to users.

A fully secure multi-authority CP-ABE (MACP-ABE) scheme in the standard model was proposed by Liu et al. This scheme was based on the previous CP-ABE scheme. In this scheme, there are multiple central authorities and attribute authorities. The central authorities distribute identity related keys to users, while the attribute authorities distribute attribute-related keys to users. Prior to possessing attribute keys from the attribute authorities, the user must obtain secret keys from the multiple central authorities. This scheme was constructed in the bilinear group with Composite order

(N = p1 p2 p3).

Lekwo and Waters proposed a new MA-ABE scheme called decentralizing CP-ABE (DCP-ABE) scheme. This scheme improved the previous MA-ABE schemes that require collaborations among multiple authorities to initial the system. In this scheme, no cooperation between the multiple authorities is required in the setup stage and the key generation stage, and a central authority is not required.

Notably, an authority in this scheme can join or leave the system dynamically without the need to reinitialize the system. The scheme was constructed in the bilinear group with Composite order (N = p1 p2 p3),and achieved full (adaptive) security in the random oracle model. Furthermore, they also proposed two methods to create a prime order group variant of their scheme. Nevertheless, the authorities can collect a user’s attributes by tracing his GID.

Chase and Chow first proposed a privacy-preserving MA-ABE (PPMA-ABE) scheme which improved the previous scheme and removed the need of a central authority. In previous MA-ABE schemes, to obtain the corresponding secret keys, a user must submit his GID to each authority. Hence, multiple authorities can collaborate to collect the user’s attributes by his GID.

Chase and Chow provided an anonymous key issuing protocol for the GID by using the 2-party secure computing technique. As a result, a group of authorities cannot collaborate to collect the user’s attributes by tracing his GID. Nevertheless, the multiple authorities must cooperate to initial the system. Meanwhile, each pair of authorities must execute the 2-party key exchange protocol to share the seeds of the selected pseudo random functions (PRFs).

This scheme is N −2 tolerant, namely the scheme is secure if and only if the number of the compromised authorities is no more than N − 2, where Nis the number of the authorities in the system. The authorities cannot know any information about the user’s GID, but they can know the user’s attributes. Chase and Chow also left an open challenging research problem on how to construct a PPMA-ABE scheme without the need of cooperation’s among authorities.

Li proposed a MACP-ABE scheme with account ability. In this scheme, the anonymous key issuing protocol was employed. Specifically, a user can be identified when he shared his secret keys with others. Likewise, the multiple authorities must cooperate to initialize the system.

Recently, a privacy-preserving decentralized KP-ABE (PPDKP-ABE) scheme was proposed by Han et al. In this scheme, multiple authorities can work independently without any collaboration. Especially, a user can obtain secret keys from multiple authorities without releasing anything about his GID to them, and the central authority is not required.

Qian et al. proposed a privacy-preserving decentralized CP-ABE (PPDCP-ABE) scheme where simple access structures can be implemented. Nevertheless, similar to that in, the authorities in these schemes can also collect the user's attributes.

DISADVANTAGES

- The collusion attacks must be resisted. Since the DCP-ABE scheme was constructed in the random oracle model, the collusion attacks can be easily resisted by tring the user's secret keys to his GID.
- The user must convince each authority that the attributes for which he is obtaining secret keys are monitored by the authority as the authority cannot know his attributes.
- The authority can interact with the user to generate correct secret keys for him even if he does not know the user's identifier and attributes.
- The secret keys derived from multiple authorities can be used together to decrypt a ciphertext.

III. OUR CONTRIBUTIONS

PPDCP-ABE: Privacy-Preserving Decentralized Cipher-Policy Attribute-Based Encryption

Cipher-policy attribute-based encryption (CP-ABE) is a more efficient and flexible encryption system as the encryptor can control the access structure when encrypting a message. We propose a privacy-preserving decentralized CP-ABE (PPDCP-ABE) scheme where the central authority is not required, namely each authority can work independently without the cooperation to initialize the system.

Meanwhile, a user can obtain secret keys from multiple authorities without releasing his global identifier (GID) and attributes to them. This is contrasted to the previous privacy-preserving multi-authority ABE (PPMA-ABE) schemes where a user can obtain secret keys from multiple authorities with them knowing his attributes and a central authority is required.

Some sensitive attributes can also release the user's identity information. Hence, contemporary PPMA-ABE schemes cannot fully protect users' privacy as multiple authorities can cooperate to identifier a user by collecting and analysing his attributes. Therefore, it remains a challenging and important work to construct a PPMA-ABE scheme where the central authority is not required and both the identifiers and the attributes are considered. We propose a privacy-preserving decentralized CP-ABE (PPDCPABE) scheme. In our scheme, any authority can dynamically join or leave the system, and there is no any requirement for the central authority or interactions among multiple authorities.

As a notable feature, each authority can work independently, while other authorities do not need to change their secret keys and reinitialize the system when an authority joins or leaves the system. Each PPDCP-ABE 3authority monitors a set of attributes and distributes secret keys to users accordingly. To resist the collusion attacks, user's secret keys are tied to his GID. Especially, a user can obtain secret keys for his attributes from multiple authorities without revealing any information about his GID and attributes to the authorities.

Therefore, it provides stronger privacy compared to the previous PPMA-ABE schemes where only the identifier is protected. To encrypt a message, the encryptor selects an access structure for each authority and encrypts the message under them so that only the users whose attributes satisfy all the access structures can decrypt the ciphertext and obtain the plaintext.

Compared to the existing decentralized ABE scheme which was constructed in the random oracle model, our scheme is designed in the standard model. To the best of our knowledge, it is the first PPDCP-ABE scheme where both the identifiers and attributes are considered.

Now, we define the security of a PPDCP-ABE scheme, which informally is any IND-sAS-CPA-secure DCP-ABE scheme with a privacy-preserving key extract algorithm PPKeyGen that satisfies two properties: leak-freeness and selective-failure blindness. Leak-freeness requires that by executing the algorithm PPKey-Gen with honest authorities, the malicious user cannot know anything which it cannot know by executing the algorithm Key Gen with the authorities.

Selective failure blindness requires that malicious authorities cannot know anything about the user's identifier and his attributes, and cause the PPKey Gen algorithm to selectively fail depending on the user's identifier and his attributes.

Considering to reduce trust on the authorities, some privacy preserving MA-ABE schemes have been proposed. However, in these schemes, only the privacy of the GID was considered. Hence, existing schemes cannot provide a full solution to protect users' privacy in MA-ABE schemes as some sensitive attributes can also reveal the user's identity. Therefore, our scheme provides a perfect solution for the privacy issues in MA-ABE schemes.

Cipher text -Policy Attribute-Based Encryption

In several distributed systems a user should only be able to access data if a user possesses a certain set of credentials or attributes. Currently, the only method for enforcing such policies is to employ a trusted server to store the data and mediate access control. However, if any server storing the data is compromised, then the confidentiality of the data will be compromised. In this paper we present a system for realizing complex access control on encrypted data that we call Cipher text-Policy Attribute-Based Encryption.

By using our techniques encrypted data can be kept confidential even if the storage server is untrusted; moreover, our methods are secure against collusion attacks. Previous Attribute-Based Encryption systems used attributes to describe the encrypted data and built policies into user's keys; while in our system attributes are used to describe a user's credentials, and a party encrypting data determines a policy for who can decrypt. Thus, our methods are conceptually closer to traditional access control methods such as Role-Based Access Control (RBAC).

In many situations, when a user encrypts sensitive data, it is imperative that she establish a specific access control policy on who can decrypt this data. For example, suppose that the FBI public corruption offices in Knoxville and San Francisco are investigating an allegation of bribery involving a San Francisco lobbyist and a Tennessee congressman.

The head FBI agent may want to encrypt a sensitive memo so that only personnel that have certain credentials or attributes can access it. For instance, the head agent may specify the following access structure for accessing this information: ((“Public Corruption Office” AND (“Knoxville” OR “San Francisco”)) OR (management-level > 5) OR “Name: Charlie Epees”).Traditionally, this type of expressive access control is enforced by employing a trusted server to store data locally. The server is entrusted as a reference monitor that checks that a user presents proper certification before allowing him to access records or files. However, services are increasingly storing data in a distributed fashion across many servers.

Replicating data across several locations has advantages in both performance and reliability. The drawback of this trend is that it is increasingly difficult to guarantee the security of data using traditional methods; when data is stored at several locations, the chances that one of them has been compromised increases dramatically. For these reasons we would like to require that sensitive data is stored in an encrypted form so that it will remain private even if a server is compromised.

Security intuition: As in previous attribute-based encryption schemes the main challenge in designing our scheme was to prevent against attacks from colluding users. Like the scheme of Sahai and Waters our solution randomizes users private keys such that they cannot be combined; however, in our solution the secret sharing must be embedded into the cipher-text instead to the private keys. In order to decrypt an attacker clearly must recover $e(g, g)\alpha s$.

In order to do this the attacker must pair C from the ciphertext with the D component from some user's private key. This will result in the desired value $e (g, g) \alpha s$, but blinded by some valuee $(g, g)rs$. This value can be blinded out if and only if enough the user has the correct key components to satisfy the secret sharing scheme embedded in the ciphertext.

Collusion attacks won't help since the blinding value is randomized to the randomness from a particular user's private key. While we described our scheme to be secure against chosen plaintext attacks, the security of our scheme can efficiently be extended to chosen ciphertext attacks by applying a random oracle technique such as that of the Fujisaki-Okamoto transformation. Alternatively, we can leverage the delegation mechanism of our scheme and apply the Cannetti, Halevi, and Katz method for achieving CCA-security.

Optimizing the decryption strategy: The recursive algorithm given in Section 4 results in two pairings for each leaf node that is matched by a private key attribute, and up to one exponentiation for every node occurring along the path from such a node to the root (not including the root). The final step after the recursive portion adds an additional pairing. Of course, at each internal node with threshold k, the results from all but k of its children are thrown away.

By considering ahead of time which leaf nodes are satisfied and picking a subset of them which results in the satisfaction of the entire access tree, we may avoid evaluating Decrypt Node where the result will not ultimately be used. More precisely, let M be a subset of the nodes in an access tree T . We define restrict (T ,M) to be the access tree formed by removing the following nodes from T (while leaving the thresholds unmodified). First, we remove all nodes not in M. Next we remove any node not connected to the original root of T along with any internal node x that now has fewer children than its threshold kx. This is repeated until no further nodes are removed, and the result is restrict(T ,M). So given an access tree T and a set of attributes γ that satisfies it, the natural problem is to pick a set M such that γ satisfies restrict(T ,M) and the number of leaves in M is minimized (considering pairing to be the most expensive operation).

This is easily accomplished with a straightforward recursive algorithm that makes a single traversal of the tree. We may then use Decrypt Node on restrict(T ,M) with the same result. Our system allows for a new type of encrypted access control where user's private keys are specified by a set of attributes and a party encrypting data can specify a policy over these attributes specifying which users are able to decrypt.

Our system allows policies to be expressed as any monotonic tree access structure and is resistant to collusion attacks in which an attacker might obtain multiple private keys. Finally, we provided an implementation of our system, which included several optimization techniques.

Provably Secure Ciphertext Policy ABE

In ciphertext policy attribute-based encryption (CP-ABE), every secret key is associated with a set of attributes, and every ciphertext is associated with an access structure on attributes. Decryption is enabled if and only if the user's attribute set satisfies the ciphertext access structure. This provides fine-grained access control on shared data in many practical settings, including secure databases and secure multicast.

We study CP-ABE schemes in which access structures are AND gates on positive and negative attributes. Our basic scheme is proven to be chosen plaintext(CPA) secure under the decisional bilinear Diffie-Hellman(DBDH) assumption. We then apply the Canetti-Halevi-Katz technique to obtain a chosen ciphertext (CCA) secure extension using one-time signatures.

The security proof is a reduction to the DBDH assumption and the strong existential unforgeability of the signature primitive. In addition, we introduce hierarchical attributes to optimize our basic scheme reducing both ciphertext size and encryption/decryption time while maintaining CPA security.

Finally, we propose an extension in which access policies are arbitrary threshold trees, and we conclude with a discussion of practical applications of CP-ABE.All existing ABE schemes involve some form of threshold secret sharing construction. Shares of a system master secret are embedded into user secret keys, while in shares of the randomness in an encryption are embedded into ciphertext components. We break from this tradition and consider AND-gates on positive and negative attributes as our access structures.

We show that, by separating threshold secret sharing from the CPABE primitive, we obtain simple and efficient schemes that are provably secure under standard complexity assumptions. Furthermore, threshold access policies can be re-introduced in an independent mechanism; namely, one can construct shares of a message using a standard secret sharing scheme and encrypt each share independently using CP-ABE. We present a CP-ABE scheme that is chosen plaintext (CPA) secure under the Decisional Bilinear Diffie-Hellman (DBDH) assumption. Access structures in this scheme are AND-gates on positive and negative attributes.

We then apply the Canetti-Halevi-Katz technique to obtain a chosen ciphertext (CCA) secure extension, using one-time signatures. Since strongly existentially unforgeable signatures can be constructed under the standard computational Diffie-Hellman (CDH) assumption, the security of our CCA scheme reduces to DBDH and CDH.

To our best knowledge, this is the first formal CCA security proof for CP-ABE. We observe that attributes can be arranged into logical hierarchies, which in turn can be used to improve the efficiency of our basic scheme. Essentially, a hierarchy allows us to use fewer group elements to represent all attributes in the system, thereby reducing the ciphertext size, the number of exponentiations in encryption and the number of pairings in decryption. This optimized scheme is proven to be CPA secure.

Finally, we note that threshold access policies can be enforced by first performing secret sharing on the message and then encrypting the shares independently using our CPABEscheme. As a special case, one can encrypt to any disjunctive normal form (DNF) formula on attributes by encrypting the same message to every AND gate in the formula. We discuss some subtleties in the security of this proposal and leave the formal proof as important future work.

Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data

Fine-grained access control systems facilitate granting differential access rights to a set of users and allow flexibility in specifying the access rights of individual users. We introduce new techniques to implement fine grained access control. In our techniques, the data is stored on the server in an encrypted form while different users are still allowed to decrypt different pieces of data per the security policy. This effectively eliminates the need to rely on the storage server for preventing unauthorized data access.

In this paper, more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encrypting data is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). We develop a new cryptosystem for fine-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE).

In our cryptosystem, ciphertexts are labeled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. We demonstrate the applicability of our construction to sharing of audit-log information and broadcast encryption. Our construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE).

We develop a much richer type of attribute-based encryption cryptosystem and demonstrate its applications. In our system each ciphertext is labeled by the encryptor with a set of descriptive attributes. Each private key is associated with an access structure that specifies which type of ciphertexts the key can decrypt. We call such a scheme a Key-Policy Attribute-Based Encryption (KP-ABE), since the access structure is specified in the private key, while the ciphertexts are simply labeled with a set of descriptive attributes. We note that this setting is reminiscent of secret sharing schemes.

Using known techniques one can build a secret-sharing scheme that specifies that a set of parties must cooperate in order to reconstruct a secret. For example, one can specify a tree access structure where the interior nodes consist of AND and OR gates and the leaves consist of different parties. Any set of parties that satisfy the tree can reconstruct the secret.

In our construction each user's key is associated with a tree-access structure where the leaves are associated with attributes. A user is able to decrypt a ciphertext if the attributes associated with a ciphertext satisfy the key's access structure. The primary difference between our setting and secret-sharing schemes is that while secret-sharing schemes allow for cooperation between different parties, in our setting, this is expressly forbidden.

For instance, if Alice has the key associated with the access structure “X AND Y", and Bob has the key associated with the access structure “Y AND Z", we would not want them to be able to decrypt a ciphertext whose only attribute is Y by colluding. To do this, we adapt and generalize the techniques introduced to deal with more complex settings. We will show that this cryptosystem gives us a powerful tool for encryption with fine-grained access control for applications such as sharing audit log information. In addition, we provide a delegation mechanism for our construction. Roughly, this allows any user that has a key for access structure X to derive a key for access structure Y, if and only if Y is more restrictive than X.

Somewhat surprisingly, we observe that our construction with the delegation property subsumes Hierarchical Identity-Based Encryption and its derivatives. Our current constructions do not hide the set of attributes under which the data is encrypted. However, if it were possible to hide the attributes, then viewing attributes as keywords in such a system would lead to the first general keyword-based search on encrypted data. A search query could potentially be any monotone Boolean formula of any number of keywords. We leave the problem of hiding the set of attributes as open.

Decentralizing Attribute-Based Encryption

We propose a Multi-Authority Attribute-Based Encryption (ABE) system. In our system, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters. A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that reflect their attributes. A user can encrypt data in terms of any Boolean formula over attributes issued from any chosen set of authorities. Finally, our system does not require any central authority.

In constructing our system, our largest technical hurdle is to make it collusion resistant. Prior Attribute-Based Encryption systems achieved collusion resistance when the ABE system authority \tied" together different components (representing different attributes) of a user's private key by randomizing the key. However, in our system each component will come from a potentially different authority, where we assume no coordination between such authorities. We create new techniques to tie key components together and prevent collusion attacks between users with different global identifiers. We prove our system secure using the recent dual system encryption methodology where the security proof works by first converting the challenge ciphertext and private keys to a semi-functional form and then arguing security.

We follow a recent variant of the dual system proof technique due to Lewko and Waters and build our system using bilinear groups of Composite order. We prove security under similar static assumptions to the LW paper in the random oracle model.

This simple system enjoys multiple benefits. Since encryption simply uses a prior ABE system, we can achieve the same level of expressiveness and write a policy in terms of any Boolean formula. The system also requires minimum coordination between separate authorities.

Any party can choose to be an authority by creating and publishing a verification key coupled with a list of attributes it will manage. Different authorities will not need to coordinate or even be aware of each other. There are several issues that will need to be dealt with in any larger system, such as the choice of an appropriate global identifier 1 or a party's decision as to which authority it trusts to issue private keys related to certain attributes. For instance, one might encrypt a policy using Experian's verification key to attest for the attribute of a good FICO(credit) score.

The major drawback of this simple engineered approach is that it requires a designated central authority. This authority must be globally trustworthy, since its failure will compromise the entire system. If we aim to build a large or even global scale system, this authority will become a common bottleneck. Spreading a central authority's keys over several machines to alleviate performance pressures might simultaneously increase the risk of key exposure.

A few works have attempted to create new cryptographic solutions to the multi-authority ABE problem. Chase proposed an interesting solution that introduced the concept of using a global identifier as a \linchpin" for tying users' keys together. Her system relied on a central authority and was limited to expressing a strict \AND" policy over a pre-determined set of authorities.

Therefore a party encrypting would be much more limited than in the simple engineering approach outlined above. Muller, Katzenbeisser, and Eckert give a different system with a centralized authority that realizes any LSSS access structure. Their construction builds on the Waters system; their proof is limited to non-adaptive queries.

The system achieves roughly the same functionality as the engineering approach above, except one can still acquire attributes from additional authorities without revisiting the central authority. Chase and Chow showed how to remove the central authority using a distributed PRF; however, the same limitations of an AND policy of a determined set of authorities remained. Lin et. al.

Our Contribution We proposes a new multi-authority Attribute-Based Encryption system. In our system, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters. (These will be created during a trusted setup.) A party can simply act as an authority by creating a public key and issuing private keys to different users that reflect their attributes. Different authorities need not even be aware of each other.

A user can encrypt data in terms of any Boolean formula over attributes issued from any chosen set of authorities. Finally, our system does not require any central authority. We thus avoid the performance bottleneck incurred by relying on a central authority, which makes our system more scalable. We also avoid placing absolute trust in a single designated entity which must remain active and uncorrupted throughout the lifetime of the system.

This is a crucial improvement for efficiency as well as security, since even a central authority that remains uncorrupted may occasionally fail for benign reasons, and a system that constantly relies on its participation will be forced to remain stagnant until it can be restored. In our system, authorities can function entirely independently, and the failure or corruption of some authorities will not affect the operation of functioning, uncorrupted authorities. This makes our system more robust then the other approaches outlined above.

Improving Privacy and Security in Multi-Authority Attribute-Based Encryption

Attribute based encryption (ABE) determines decryption ability based on a user’s attributes. In a multi-authority ABE scheme, multiple attribute-authorities monitor different sets of attributes and issue corresponding decryption keys to users and encryptor can require that a user obtain keys for appropriate attributes from each authority before decrypting a message. Chase gave a multi-authority ABE scheme using the concepts of a trusted central authority (CA) and global identifiers (GID). However, the CA in that construction has the power to decrypt every ciphertext, which seems somehow contradictory to the original goal of distributing control over many potentially untrusted authorities.

Moreover, in that construction, the use of a consistent GID allowed the authorities to combine their information to build a full profile with all of a user’s attributes, which unnecessarily compromises the privacy of the user. In this paper, we propose a solution which removes the trusted central authority, and protects the users’ privacy by preventing the authorities from pooling their information on particular users, thus making ABE more usable in practice.

Since each authority is responsible for different attributes, we want to allow them to issue decryption keys independently, without having to communicate with one another.As argued in, in order to prevent collusion in such a setting, we need some consistent notion of identity. (Otherwise, a user could easily obtain keys from one authority and then give them all to a friend.)

The solution in that work is to require that each user have a unique global identifier (GID),which they must present to each authority (and to require that the user prove in some way that he is the owner of the GID he presents). Unfortunately, the mere existence of GID makes it very hard for the users to guarantee any kind of privacy. Because a user must present the same GID to each authority, it is very easy for colluding authorities to pool their data and build a “complete profile” of all of the attributes corresponding to each GID.

However, this might be undesirable, particularly if the user uses the ABE system in many different settings, and wishes to keep information about some of those settings private. This situation seems to be unavoidable if all one's attributes are determined by some kind of public identity like a name or SSN – in that case users will need to identify themselves in any case in order to get the decryption keys for a certain set of attributes, so privacy is unavoidably com for further discussion. However, there are many attributes which do not belong to this category. The ability to drive is a good example.

One should be able to prove the ability to do something in an examination and then get the corresponding credential, without presenting any identifying information. Alternatively, one might interact with a service via a pseudonym (e.g. a login name) and wish to obtain attributes relating to this interaction without revealing one's full identity.

Regardless, as the attribute-authorities (AAs) are responsible for managing each user's attributes, it seems inevitable that they will learn which subsets of its attributes are held by different users.

However, we could imagine applications where some of the authorities are different online service providers giving attributes related to online activities like blog/wiki contributions, access to online news sites, participation in social networking sites, or purchases at an online store.

In this case, it would make sense for the user to be able to maintain different, unlinkable attribute sets with each authority. At the same time, it also makes sense for each AA to gather the statistics of their system usage (e.g. the number of users subscribed a particular service as indicated by the number of users who requested a decryption key for a certain attribute) without compromising individual's privacy.All of the prior work described above considers the scenario where all of the attributes are monitored by a single authority. However, as we mentioned in Section 1, it seems natural that one might want to divide control of the various attributes over many different authorities.

The main challenge here is to guarantee that two colluding users cannot each obtain keys from a different authority, and then pool their keys to decrypt a message that they are not entitled to. Furthermore, in the multi-authority case, we may wish to allow for some of the authorities to be untrusted. The techniques for single authority ABE cannot be easily generalized in this case they rely on the fact that the single authority can generate all of a user's keys at once, to ensure that they can only be used together, and cannot be combined with any other user's keys.

The only multi-authority ABE schemes we are aware of are Chase's original proposal (which has already been discussed and the very recent Lin et al. extension. Both schemes are KP-ABE and operate in a setting where multiple authorities are responsible for disjoint sets of attributes. The disadvantages of Chase's scheme have already been discussed. The scheme of, like the scheme we will present here, has the advantage that it does not rely on a central authority. However, their scheme only achieves m-resilience, in that security is only guaranteed against a maximum of m colluding users. And this is not merely an issue of formal security: Lin et al. demonstrated a collusion attack of m+1 user.

In their scheme m is the number of secret keys that each authority obtains from a distributed key generation protocol. (This also means m must be determined when the system is initialized.) Clearly, for a large scale system, m should set reasonably high in order to guarantee security (a very loose desirable lower bound should be N2, where N is the number of authorities). This imposes burdens on the interactive distributed key generation protocol among all the authorities, and on their secure storage. Finally, O (m) online modular operations are required by each authority to issue secret keys to a user.

We further note that this weaker notion of security seems undesirable. It may be of commercial interest to have as many users as possible, yet it simultaneously increases the risk of being compromised. Thus, we argue that it is still a very important open problem to design an efficient and secure multi-authority ABE scheme without a trusted CA, and this is one of the problems we will attempt to solve here. Here we present a multi-authority ABE with user privacy and without the trusted authority.

In this paper, we propose a privacy-preserving DCP-ABE (PPDCP-ABE) scheme where the central authority is not required and each authority can work independently without any cooperation. As a notable feature, each authority can dynamically join or leave the system, namely other authorities do not need to change their secret keys and reinitialize the system when an authority joins or leaves the system.

Each authority monitors a set of attributes and issues secret keys to users accordingly. To resist the collusion attacks, a user’s secret keys are tied to his GID. Especially, a user can obtain secret keys for his attributes from multiple authorities without them knowing any information about his GID and attributes. Therefore, the proposed PPDCP-ABE scheme can provide stronger privacy protection compared to the previous PPMA-ABE schemes where only the GID is protected.

When encrypting a message, the encryptor can select an access structure for each authority and encrypt the message under the selected access structures so that a user can decrypt the ciphertext if his attributes satisfy all the access structures. Comparatively, our scheme is constructed in the standard model, while the existing DCP-ABE scheme was designed in the random oracle model. To the best of our knowledge, it is the first PPDCP-ABE scheme where the privacy of both the identifiers and attributes are considered.

CHALLENGES

When constructing a PPDCP-ABE scheme, the following technical hurdles must be overcome.

- First, the collusion attacks must be resisted. Since the DCP-ABE scheme was constructed in the random oracle model, the collusion attacks can be easily resisted by tieing the user’s secret keys to his GID. However, it is challenging to resist the collusion attacks in the DCP-ABE scheme which is designed in the standard model.
- Second, the user must convince each authority that the attributes for which he is obtaining secret keys are monitored by the authority as the authority cannot know his attributes.
- Third, the authority can interact with the user to generate correct secret keys for him even if he does not know the user’s identifier and attributes.
- Finally, the secret keys derived from multiple authorities can be used together to decrypt a ciphertext.

IV. SYSTEM ARCHITECTURE

INPUT

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- ➢ What data should be given as input?
- ➢ How the data should be arranged or coded?
- ➢ The dialog to guide the operating personnel in providing input.
- ➢ Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3.When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user

will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

OUTPUT

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
2.Select methods for presenting information.
3.Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

V. CONCLUSION

Some PPMA-ABE schemes have been proposed to protect users' privacy and reduce the trust on the central authority. Nevertheless, only the privacy of the GID was considered in the existing scheme. Since sensitive attributes can also reveal the users' identities, existing schemes cannot provide a full solution to protect users' privacy in MA-ABE schemes. In this paper, we proposed a PPDCP-ABE scheme where both the privacy of the GID and the attributes are concerned. In our scheme, a central authority is not required and multiple authorities can work independently without any cooperation.

A user can convince the authorities that the attributes for which he is obtaining secret keys are monitored by them without showing the attributes to them. Therefore, our scheme provides a perfect solution for the privacy issues in MA-ABE schemes.

VI. FUTURE WORK

As for future research direction regarding PPDCP-ABE, it would be interesting to construct a fully secure PPDCP-ABE scheme since the scheme proposed in this paper is selectively secure

VII. REFERENCES

[1]. Au.M.H, Han.J, Mu.J, and Susilo.W, ZhoU.J, (2014) "PPDCPABE: Privacy-preserving decentralized ciphertext-policy attribute-based encryption," in Computer Security (Lecture Notes in Computer Science), vol. 8713. Cham, Switzerland: Springer-Verlag, pp. 73-90.

[2]. Bethencourt.J, Sahai.A, and Waters.B, (2007) "Ciphertext-policy attribute based encryption," in Proc. IEEE Symp. SP, pp. 321-334.

[3]. Bichsel.P, Camenisch.J, GroB.T, and Shoup.V, (2009) "Anonymous credentials on a standard Java Card," in Proc. ACM Conf. CCS, pp. 600-610.

[4]. Chase.M and ChowS.S, (2006) "Improving privacy and security in multi authority attribute-based encryption," in Proc. 16th ACM Conf. CCS, 2009, pp. 121-130.

[5]. Cheung.L and Newport.C , (2007) "Provably secure ciphertext policy ABE," in Proc. 14th ACM Conf. CCS, pp. 456-465.

[6]. Goyal.V, Pandey.O, Sahai.A, and Waters.B, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th ACM Conf. CCS, pp. 89-98.

[7]. Lewko.A and Waters.B, (2011) "Decentralizing attribute-based encryption," in Advances in Cryptology (Lecture Notes in Computer Science), vol. 6632. Heidelberg, Germany: Springer-Verlag, pp. 568-588.

[8]. Ostrovsky.R, Sahai.A, and Waters.B, (2007) "Attribute-based encryption with non-monotonic access structures," in Proc. 14th ACM Conf. CCS, pp. 195-203.

[9]. Sahai.A and Waters.B, (2005) "Fuzzy identity-based encryption," in Advances in Cryptology (Lecture Notes in Computer Science), vol. 3494. Heidelberg, Germany: Springer-Verlag, , pp. 457-473.