Efficient Multi-level Threshold Attribute Based Encryption

M.Tech Project Interim Review Report

Subhashini V — CS09M054
Guide: Prof. C Pandu Rangan
Area: Theoretical Computer Science

Abstract

Anonymous access control is a very desirable property in various applications e.g. encrypted storage in distributed environments; and attribute based encryption (ABE) is a cryptographic scheme that is targeted to achieve this property. ABE is an encryption mechanism that is useful in settings where the list of users may not be known apriori, but all users may possess certain credentials which can be used in determining access control and at the same time providing a reasonable degree of anonymity. Ciphertext policy attribute based encryption is a scheme that gives a natural way to separate the credentials from the access policy and cleverly combine them at a later stage to provide secure access to protected data. In most ABE schemes the size of the ciphertext is quite large and is of the order of the number of attributes. In this work we present our approach for a multi-level threshold attribute based encryption which is independent of the number of attributes. We conclude this report with the idea for a possible proof sketch and future work.

Keywords

Attribute based encryption, ABE, CP-ABE, multi-level threshold, constant size ciphertext.

1 Introduction

There are several settings where a user would want to give access to documents based on certain credentials or the position/role of a person. This may be comparable with 'Views' in a database. We would want different kind of users of the database to be able to see only those contents that are relevant to them. Similarly, in a distributed setting where all the data may be stored in a server, the server might allow access to files and documents based on some predefined access control policy, for example, clients may have to provide proper certification to retrieve specific files. In such cases, if the data(storage) in the database or server is compromised, then although it may be in the encrypted form, anyone who has access to the database or server may be able retrieve all information including those documents that may not be relevant to them. To be more specific, any normal user of the database who gets his/her hands on the compromised data may now be able to get those files which were restricted and whose access was determined by some application in the database or server.

ABE Attributed based encryption (ABE), first introduced by Sahai and Waters [1, 2], provides a mechanism by which we can ensure that even if the storage is compromised, the loss of information will only be minimal. What attribute based encryption does is that, it effectively binds the access-control policy to the data and the users(clients) instead of having a server mediating access to files. To understand this better, we will take a closer look into what access control is.

Access Policy. An access control policy would be a policy that defines the kind of users who would have permissions to read the documents. e.g In an academic setting, grade-sheets of a class may be accessible only to a professor handling the course and some teaching assistants (TAs) of that course. We can express such a policy in terms of a predicate:

$$\Big(\text{(Professor} \wedge \text{CS dept.)} \bigvee \text{(M.tech student} \wedge \text{course TA} \wedge \text{CS dept.)} \Big)$$

We will call the various credentials (or variables) of the predicate as attributes and the predicate itself which represents the access policy as the access-structure. In the example here the access structure is quite simple. But in reality, access policies may be quite complex and may involve a large number of attributes.

Properties. There are two major features to attribute based encryption:

1. It has the capacity to address complex access control policies.
2. The exact list of users need not be known apriori. Knowledge of the access policy is sufficient.

Also, an important property that attribute based encryption schemes must satisfy is that of *collusion resistance*. *Collusion resistance* means that, if 2 or more users possessing different keys combine to decrypt the ciphertext, they will be successful *if and only if* any one of the users could have decrypted it individually. In other words, even if multiple parties collude, they should not be able to decrypt the ciphertext unless one of them was able to decrypt it completely by herself. These properties ensure that only users possessing the right keys have access to the information. Moreover, as the encryption is based on the access-structure it implicitly assures anonymous access control.

Types of ABE. ABE can be categorized in to two types depending on whether the attributes are embedded in the ciphertext or whether the access-structure is embedded in the ciphertext. The first is the Key-policy based ABE (KP-ABE) which was infact the initial form of attribute based encryption that was developed. It was originally introduced in [1] and later by Goyal et al. in [2] and by Ostrovsky in [3]. In KP-ABE they encrypt the attributes along with the data and give the access structure to each user as part of their secret key. But attribute based encryption is more applicable in the regular world if the access-structure can be embedded in the ciphertext and the users can have their attributes saved in their secret keys. This second form of ABE is known as ciphertext-policy based (CP-ABE) and was introduced by Bethencourt et al [4]. Both these initial schemes [2, 4] were largely based on the secret sharing scheme developed by Shamir [5]. However, it is ciphertext policy based ABE that has become more popular in later schemes like [6, 7, 8, 9] and others. This might be largely due to the fact that CP-ABE represents a natural and more intuitive way to view attribute based encryption.

Contribution. In most of the previous ABE schemes, the size of the ciphertext is very large, it is usually in the order of the number of attributes under consideration. There have been works on more efficient and constant size CP-ABE schemes. However, the current constant size ciphertext schemes are applicable only to some restricted access structures[10, 11] and even the most efficient scheme with expressive access control has ciphertext size proportional to the number of attributes involved [12]. In this work we propose an approach to get a multi-level threshold CP-ABE where the ciphertext size is independent of the number of attributes. Moreover, the access structure we use is more expressive and can be used to represent complex access control policies.

2 Related Work

The ciphertext policy ABE scheme developed by Bethencourt et al. [4], which was based on secret sharing was actually a scheme that supported multi-level threshold access structures. The scheme was developed in a manner which could very easily be extended to support generic (monotone) access structure by simply replacing any AND by *n-out-of-n* threshold gate and an OR by *1-out-of-n* threshold gate. Later, the work by Ostrovsky in [3] gave a KP-ABE scheme that extended to non-monotone access structures. The authors have mentioned ways to extend their work to get a CP-ABE scheme which supports non-monotone threshold access structures. However, inorder to give CP-ABE schemes other properties, including better security, the simple AND access structure became more popular. The AND access structure was preferred by Cheung and Newport in [6] to give better proof of security and a similar access structure was used by Nishide et al. in [8] to make CP-ABE support recipient anonymity (partially hide the actual access structure from the decryptor). The use of the simple AND based access structure was also used to develop schemes that gave more efficient ciphertext size.

All the initial attribute based encryption schemes, both KP-ABE and CP-ABE had ciphertext size and key size in the order of atleast the number of attributes involved. The first attempt to make the most efficient ciphertext policy attribute based encryption can be credited to Waters [12]. Here he proposes a scheme in which the size of the ciphertext is equal to the number of attributes involved, with a constant additive factor. Then in 2009, the work by Emura et al [11] was one of the initial attempts to obtain a constant size ciphertext. However their scheme which was based on the access structure in [6] supported only the all-AND access structure. Later, Zhou et al in [10] also proposed an efficient constant size ciphertext CP-ABE, however their access structure also supported only the AND operation. Recently, the work by Herranz et al., [13] proposes an elegant scheme for a constant ciphertext size threshold CP-ABE. Their work makes use of a clever aggregate method proposed by Delerablée and Pointcheval in [14]. Although the threshold scheme is more expressive, the aggregate function is useful as long as there is only one level in the access structure and does not seem to be easily extendible to the multi-level threshold case.

In this work we make use of the Aggregate function to obtain a more expressive multi-level threshold CP-ABE which is also efficient with respect to the size of the ciphertext and the number of pairing operations involved.

3 Prelimnaries

3.1 CP-ABE Scheme Outline

- **Setup.** A randomized algorithm **Setup(k)** takes in as input a security parameter and provides a set of public parameters (PK) and the master key values (MK).
- **Encryption.** The algorithm **Enc(M, $\mathcal{T}$, PK)** is a randomized algorithm that takes as input the message to be encrypted (M), the access structure $\mathcal{T}$ which needs to be satisfied and the public parameters (PK) to output the ciphertext CT. We can say, that the encryption algorithm embeds the access structure in the ciphertext such that only those users with attributes satisfying $\mathcal{T}$ will be able to decrypt and retrieve the message M.
- **Key-Generation.** The **KeyGen(MK, PK, $\mathcal{A}$)** algorithm takes as input the master key values (MK), the public parameters (PK) and the attribute set of the user ($\mathcal{A}$), and outputs for the user a set of decryption keys SK which confirms the users possession of all the attributes in $\mathcal{A}$ and no other external attribute.
- **Decryption.** The decryption algorithm **Dec(CT, SK, PK)** takes as input the ciphertext CT, the user secret keys SK and the public parameters PK, and it outputs the encrypted message (M) if and only if the attributes $\mathcal{A}$ embedded in SK satisfy the access structure $\mathcal{T}$ which was used while encrypting the ciphertext CT. i.e If $\mathcal{T}(\mathcal{A}) = 1$ then message M is output else, it outputs $\perp$.

4 Sketch of our Scheme

In this section, we present an outline of our construction. We begin by giving the model of the access tree structure which supports multi-level thresholds and then proceed to give the ideas involved in making the ciphertext size more efficient and independent of the number of attributes.

Model. A party who wishes to encrypt a message will specify the access control predicate through an access tree structure, which we denote by $\mathcal{T}$. Any party who wishes to decrypt the ciphertext must be able to satisfy the access tree inorder to retrieve the message. The tree model we follow is similar to that described in [4]. We treat each individual non-leaf node of the tree to be a threshold gate. Note that this representation of the access policy is very expressive since AND and OR gates can be represented as threshold gates. Let s_x denote the number of children that each node x has. We will use k_x to denote the threshold value that needs to be satisfied at node x. An important point to note is that, all the attributes of the access policy form the leaves of the access tree. We will use the notation $\psi_{\mathcal{T}}$ to denote the set of last level of non-leaf nodes (i.e. those nodes whose children are all attributes/leaves).

Encryption. Enc(PK, $\mathcal{T}$, M)
Our encryption uses the idea of linear secret sharing (LSSS) [5]. For every non-leaf node x of the access tree $\mathcal{T}$, we choose a polynomial q_x. We, proceed in a top down manner in selecting the polynomials, starting from the root R. For a node x we set the degree of the node $d_x = k_x - 1$, one less than the threshold value that needs to be satisfied at the gate at that node.

Now, beginning at the root, we choose a random $s \in_R \mathbb{Z}_p^*$ and set $q_r(0) = s$. Then choose d_r other points of the polynomial q_r to define it completely. For all other non-leaf nodes x, we set $q_x(0) = q_{parent(x)}\big(index(x)\big)$ and choose d_x other points to completely define q_x.

For the last level of non-leaf nodes, $x \in \psi_{\mathcal{T}}$, we use a strategy similar to that proposed by Herranz in [13] to get a constant size ciphertext for each threshold gate. We generate a ciphertext component for each of these nodes in $\psi_{\mathcal{T}}$ and hide the secret share in this. This design makes the ciphertext size independent of the number of attributes.

Decryption. Dec(CT, SK, PK)
We use the Aggregate function as defined in [14] to decrypt the last level of non-leaf nodes, $x \in \psi_{\mathcal{T}}$ in the same manner as in [13]. We then get a component which is raised to the node's secret share if decrypted with the right set of attributes. For the higher non-leaf nodes, we combine k_x secret shares of the child nodes with the help of Lagrange's interpolation formula. The rest of the decryption procedure is similar to [4]. In case the threshold at a node is not satisfied we output $\perp$. If the tree is completely satisfied then we will be able to get the hidden value s and by applying bilinear function on components in the secret key and the obtained value, we will be able to unblind the ciphertext and recover the message.

5 Future work

Although the scheme appears to be reliable and robust we need to formally prove it to be secure under some hard cryptographic assumption. The secret key and public key components used in the scheme point us to the *Augmented Multi-sequence of Exponents Diffie-Hellman Problem* (aMSE-DDH)[13] as a potential hardness assumption. In the coming weeks, our aim would be to reduce the hardness of breaking our scheme to solving the aMSE-DDH problem.

We also believe that the ideas presented in this work may be applicable in attribute based signature schemes. We will work towards applying the same to obtain, if possible, a provably secure constant size attribute based signature scheme.

References

[1] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *EUROCRYPT*, pp. 457–473, 2005.

[2] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM conference on Computer and communications security*, CCS '06, (New York, NY, USA), pp. 89–98, ACM, 2006.

[3] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proceedings of the 14th ACM conference on Computer and communications security*, CCS '07, (New York, NY, USA), pp. 195–203, ACM, 2007.

[4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy*, pp. 321–334, 2007.

[5] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[6] L. Cheung and C. Newport, "Provably secure ciphertext policy abe," in *Proceedings of the 14th ACM conference on Computer and communications security*, CCS '07, (New York, NY, USA), pp. 456–465, ACM, 2007.

[7] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in *Proceedings of the 35th international colloquium on Automata, Languages and Programming, Part II*, ICALP '08, (Berlin, Heidelberg), pp. 579–591, Springer-Verlag, 2008.

[8] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden ciphertext policies," *IEICE Transactions*, vol. 92-A, no. 1, pp. 22–32, 2009.

[9] M. Gagné, S. Narayan, and R. Safavi-Naini, "Threshold attribute-based signcryption," in *SCN*, pp. 154–171, 2010.

[10] Z. Zhou and D. Huang, "On efficient ciphertext-policy attribute based encryption and broadcast encryption: extended abstract," in *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, (New York, NY, USA), pp. 753–755, ACM, 2010.

[11] K. Emura, A. Miyaji, A. Nomura, K. Omote, and M. Soshi, "A ciphertext-policy attribute-based encryption scheme with constant ciphertext length," in *Proceedings of the 5th International Conference on Information Security Practice and Experience*, ISPEC '09, (Berlin, Heidelberg), pp. 13–23, Springer-Verlag, 2009.

[12] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization." Cryptology ePrint Archive, Report 2008/290, 2008. http://eprint.iacr.org/.

[13] J. Herranz, F. Laguillaumie, and C. Ràfols, "Constant size ciphertexts in threshold attribute-based encryption," in *Public Key Cryptography*, pp. 19–34, 2010.

[14] C. Delerablée and D. Pointcheval, "Dynamic threshold public-key encryption," in *Proceedings of the 28th Annual conference on Cryptology: Advances in Cryptology*, CRYPTO 2008, (Berlin, Heidelberg), pp. 317–334, Springer-Verlag, 2008.