I J C T A, 9(15), 2016, pp. 7445-7451

Performance Analysis of a Variant of CP-ABE For Internet of Things

Dhirshya H.* and Jayashree P.*

ABSTRACT

Securing the data is of major concern in the Internet of Things, where one has to consider the light weight nature of the devices that are involved in the communication. Attribute Based Encryption finds its way in such applications due to its capability of providing considerable amount of security for broadcast encryption. Light weight Attribute Based Encryption Technique incurs additional computation costs, which drags itself from being used in IoT related applications. The proposed light weight hierarchical Attribute Based Encryption scheme splits the encryption phase into two sub phases so that when the message needs to be encrypted for much larger groups, it need not consider the entire number of attributes and can be encrypted just for the role. This is because as the number of attributes increases, the size of the ciphertext also increases considerably. It is found to be secure and at the same time less complex compared to the existing algorithms.

Keywords: Attribute Based Encryption, Internet of Things, CP-ABE, KP-ABE

1. INTRODUCTION

In recent days the Internet of Things (IoT) has achieved importance in almost every field [6], and is trending out to be a good area in research. IoT in broader terms could be defined as an interconnection of devices. But when we consider a stricter sense of definition it is nothing but the interconnection of smart devices such as laptops, mobile phones that interact with each other continuously and generate huge amount of data of application importance. The main concept here is that these devices communicate without any human intervention. This interconnection generates a huge amount of data that should be collected, aggregated, stored and shared in an efficient manner. These devices could be connected to the internet using Internet Protocol (IP) and are uniquely identified. This unique identification makes the devices to be easily controlled any time. When there is large interconnection of these devices, there is a greater need to secure the data being generated by these devices. The security mechanisms for the IoT devices are standardized by the Internet Engineering Task Force (IETF), and they specify the security protocols for the same. More over the IoT consist of large number of resource constrained devices that are being interconnected. Hence there is a limitation in applying the existing cryptographic algorithms on to these devices, since not only data security speaks, but also the resource availability is of a major concern. Hence it is important to define an algorithm that uses fewer amounts of resources and time for computation is reduced. Recently a large number of encryption techniques are widely used to provide authentication and at the same time to encrypt the data to provide confidentiality and integrity to the same. But there comes a problem when these traditional encryption techniques are used in light weight devices. There might be a situation in which a user wants to encrypt the data to a large number of users. In such cases we could not use the traditional public key encryption algorithms as the complexity under such case for encrypting the data for all the users tend to be maximum. Under such case Attribute Based Encryption (ABE) has proven to be the best encryption technique where the data could be encrypted to a large number of independent users at once. ABE is a technique where the

* Department of Computer Technology, Madras Institute of Technology, Anna University, Chennai, *Email: {drishakash, pjshree12}@gmail.com*

owner of the data could explicitly specify who has the permission to decrypt the same. Due to this nature of ABE it has gained much importance in the field of security.

2. RELATED WORKS

Attribute Based Encryption is nothing but an enhancement over the existing Identity Based Encryption (IBE) with some enhancements over the technique. It was first proposed by Sahai and Waters in 2004 and they named it to be Fuzzy identity-based encryption [1]. ABE is one in which the users could explicitly specify who has the access to the ciphertext in the access policy itself. There are two variants of ABE. They are Ciphertext policy Attribute Based Encryption (CP-ABE) and Key- policy Attribute Based Encryption (KP-ABE). CP-ABE is one in which the private key of the user is defined over a set of attributes and the cipher text specifies an access policy defined over the set of attributes. A user can decrypt the cipher text only if his attributes satisfy the policy of the respective cipher text. KP-ABE scheme is dual to cipher text-policy ABE. In this an access policy is encoded onto the user's secret key.

Hai Jiang et al. [4] proposed a method for secure and scalable storage for the data collected in IoT. Due to a large amount of data being generated seamlessly in IoT devices there is a need for securely storing the data on to the cloud storage. They used Shamir's secret sharing method and modified the same to securely store the data on to the internet. The reason for using this technique is that it does not have any complex key management involved in it. In this method the data being collected is first split into number of shares and these shares are given to a number of users. The size of each block of the share is predetermined according to a fixed threshold. The main advantage here is that the size of the shares being generated is small and moreover the original data could be reconstructed easily using these shares. Another advantage is that when a data needs to be modified, it is not necessary to regenerate all the shares, but it is enough to generate shares for only those data that need to be modified.

Lihong jiang et al. [5] proposed a method for storing the data in cloud environment for IoT platform. They addressed the issue in storing the rapid amount of data that is being generated in IoT. The method that they used for data storage is Hadoop to store and distribute the large amount of data. They proposed an efficient way in combining the structured and unstructured data being generated by the devices. This is because due to the interconnection of a variety of devices, there is a large variety of data being generated, and the structure of the data varies from structured to unstructured. They clearly segregated the data being generated from IoT devices as sensor data, storage of the data and the technology applied in such storage. They stored the data in two dimension namely time and space for efficient retrieval. They have also addressed the challenges in storing large volume of data.

Zavattoni et al. [3] proposed a scheme for easy software implementation of Attribute Based Encryption. In this method the user's attributes is specified by means of a linear secret sharing scheme. In CP-ABE bilinear pairing is generally used to provide access control mechanism. The problem with bilinear pairing operation is that they incur high cost for computation. This paper provides detailed information about the software cryptographic library for elliptic curve systems. The cost of bilinear operation is high due to the use of high amount of scalar multiplications. As the size of the scalar increases the cost also increases considerably. There are two kinds of grouping operation, one is single pairing and the other is multi-pairing. In spite of providing 126-bit level security they have also reduced considerably the computational cost associated with this algorithm. This is done by defining a software library for pairing operation and hence the time required in calculating the pairing is reduced. In an ABE protocol, a participant encrypts data that can only be decrypted by users able to satisfy an access policy previously agreed. The access policy is specified as a Boolean formula over a set of attributes. The main disadvantage here that was not being addressed is that the cost of single pairing is still expensive and could not be reduced.

Zhou et al. [10] proposed an efficient privacy-preserving CP-ABE scheme for bringing about flexibility in the data storage. Generally the size of the ciphertext increases linearly with the increase in the number of attributes in the access policy. They have proposed a method for reducing the size of the ciphertext to a constant size irrespective of the number of attributes. The message was encrypted using a hidden access policy, and hence a considerable amount in improving the security can be done. The communication and the storage cost were considerably reduced using this technique. This technique could hence be used in resource constrained environment.

Qin et al. [2] proposed an ABE scheme for verifying the decryption. In this technique the plain text is transformed into a ciphertext using a third party and moreover the ciphertext is verified for its correctness. They have used a method called Key Encapsulated Mechanism (KEM). Using this mechanism the message is first compressed using a hash function, and the output hence produced is used to check the correctness of the session key used. All the ciphertext generated are being concatenated and the hash value used above is combined and a second hash function is applied to verify the ciphertext. This method works only if the hash function is collision resistant. Using this method they could considerably reduce the size of the ciphertext to half the size as that of the previous techniques.

Yao at al. [8] proposed a light weight authentication mechanism for small scale IoT related applications. They used a modified Nyberg's fast one way accumulator to make it suitable for light weight authentication mechanism. They have tested it for IoT devices. They used a hash function to map the bit strings of arbitrary length to a fixed length string for efficient processing. They tested it on an application with number of nodes and each of these nodes used this modified algorithm to accumulate the data being generated. There was a unique shared key established with each neighbor and the group key is known by all the members of the group. The main advantage of the work is that it is resistant against node compromise and also it incurred low computational overhead and communication cost.

Fan et al. [9] proposed an ABE scheme that supports dynamic membership. In this system the users can join and leave the system, and new attributes could be added on to the system. This brings about better flexibility to the users. Three additional phases namely the enrollment, leaving and the updating phases were added in order to support this concept of flexibility. The main drawback of this work is that for maintaining the correctness of all the attributes of the users in the system requires additional cost. This overrules the concept of light weight nature o ABE.

3. LIGHT WEIGHT HIERARCHICAL ATTRIBUTE BASED ENCRYPTION

The traditional ABE scheme is modified so that the encryption is not only based on the attributes involved, but also to the roles that correspond to the attributes. The main concept here is that during the encryption phase, the message could be either encrypted under the attributes or under the roles. This is purely based on the intended receiver and not by the owner of the data. During the setup phase, the public key is being generated not only for each attributes but also for each and every role in the system. In the traditional scheme if we want to encrypt a message for all the users that correspond to a particular role, one should get the public key for each and every attribute that corresponds to the role and append the corresponding public keys of the attribute set under which the message is to be encrypted, to form the ciphertext. But when we have a public key for each and every role, the process becomes simple. It is enough that one gets the public key of that particular role and append the same on to the plain text in order to form the cipher text. This method will reduce the complexity during encryption under certain cases.

3.1. Proposed Algorithm

The ABE algorithm consists of four phases namely setup phase, Encryption phase, Keygeneration and Decryption phases. The following algorithm explains about the four phases of the algorithm.

The first phase in the execution of the CP-ABE algorithm is the setup phase. The following table explains about the setup phase of the algorithm. The input for this phase of algorithm is the set of all attributes and the roles that correspond to the system.

Table 3.1
Setup Phase

3.1.1 Setup Phase:

Input: Set of all Attributes $\{A_1,A_2,....,A_n\}$, Set of all Roles $\{R_1,R_2,....,R_m\}$

Output: Master public key and public key for each role and attribute.

Step 1: For each role, choose a number SR_i at random from Z_q^*

Step 2: For each attribute, choose a number SA_i at random from Z_q^*

Step 3: Compute Public key for each role = $SR_i * G$

Step 4: Compute Public key for each attribute = $SA_i * G$

Step 5: Compute Master public key = S.G

The Table 3.1 shows the setup phase for the algorithm. This phase generates the public key for the attribute set and also a public key for each role that correspond to the system. This phase also outputs a master public key.

The second phase in the execution of the CP-ABE algorithm is the encryption phase. The following table explains about the setup phase of the algorithm. The input for this phase of algorithm is the set of all attributes and the roles that correspond to the system. This phase outputs the ciphertext.

Table 3.2
Encryption Phase

3.1.2 Encryption Phase:

Input: Set of role -> á and Set of attributes -> β

Output: Encrypted cipher text

Encryption key derived at random by ECC which can be reconstructed under α and β

To encrypt M under α

Step 1: Select K from Z_q^*

Step 2: Compute $C^I = K. SR_i * G$

To encrypt M under β

Step 1: Select K from Z_q^*

Step 2: Compute $C^I = K. SA_i * G$

If $C^I = 0$, rechoose K again and compute

The Table 3.2 shows the encryption phase for the algorithm. This phase encrypts the message and produces the cipher text using the public key. If one wants to encrypt the message for a particular role, it is encrypted by getting the public key for that particular role. On the other hand if one wants to encrypt the message under a set of attributes, then the public key for that attribute set is being got from the master key generator and is being encrypted. This phase also outputs a master public key.

The third phase in the execution of the CP-ABE algorithm is the key generation phase. The following table explains about the setup phase of the algorithm. The input for this phase of algorithm is the access structure and this phase outputs a decryption key if and only if the value of the access structure is one.

Table 3.3
Key Generation Phase

3.1.3 KeyGeneration Phase:

Input: Access Structure

Output: Decryption key iff access structure = 1

Step 1: Assign a polynomial $q_u(X)$ of order $(d_u - 1)$ for each node u in access tree in top down manner.

Step 2: Set the value of d_u as the threshold of the node U

Step 3: For root R of access tree, set $q_r(0) = S$

Step 4: Choose $(d_r - 1)$ other points from the polynomial $q_R(X)$ for other nodes $q_u(0) = q_{parent}(U)(Index(0))$

Step 5: Compute the secret key for decryption using the formula $q_u(0) / S_i$

The Table 3.3 shows the keygeneration phase for the algorithm. This phase generates the private key for decryption based on the request by a particular user. The master key generator checks if the value of the access structure is one, which means the user satisfies the policy under which the message is being encrypted and then outputs the private key.

The last phase is the decryption phase and the procedure is shown below.

Table 3.4
KeyGeneration Phase

3.1.2 Decryption Phase:

Input: Cipher text

Output: Plain text

Step 1: Decryption is done using the following formula

Decryptnode(CM,D,U) = $q_u(0)$.K.G, I € w

The Table 3.4 shows the decryption phase of the algorithm. After getting the private key from the master key generator the user decrypts the cipher text. The input for this phase of the algorithm is the cipher text and the output is the original plaintext, if and only if the particular user satisfies the access policy under which the message is being encrypted.

4. SECURITY ANALYSIS

4.1. CPU and Memory Utilization

It is important to consider the amount of memory and CPU resource utilization as a good security algorithm with light weight nature should have very minimal amount of these resource utilization. This is done by inserting measurement statements on the executing algorithm as in [7]. The CPU and memory utilization is being considered for both CP-ABE and KP-ABE and the modified CP-ABE algorithm and is found that KP-ABE algorithm uses less amount of CPU resource for the execution of the four phases of the algorithm.

The Figure 1 shows the graphical comparison for the CPU utilization for CP-ABE, modified CP-ABE and KP-ABE algorithm for all the four phases.

It is evident that the CPU Utilization for KP-ABE algorithm is less compared to that of modified CP-ABE algorithm and also the existing CP-ABE algorithm. Moreover the CPU usage for the execution of the four phases of the algorithm is shown. For the modified CP-ABE algorithm Encryption phase consumes the least amount of resource, whereas for KP-ABE algorithm the decryption phase consumes the minimum amount of resource.

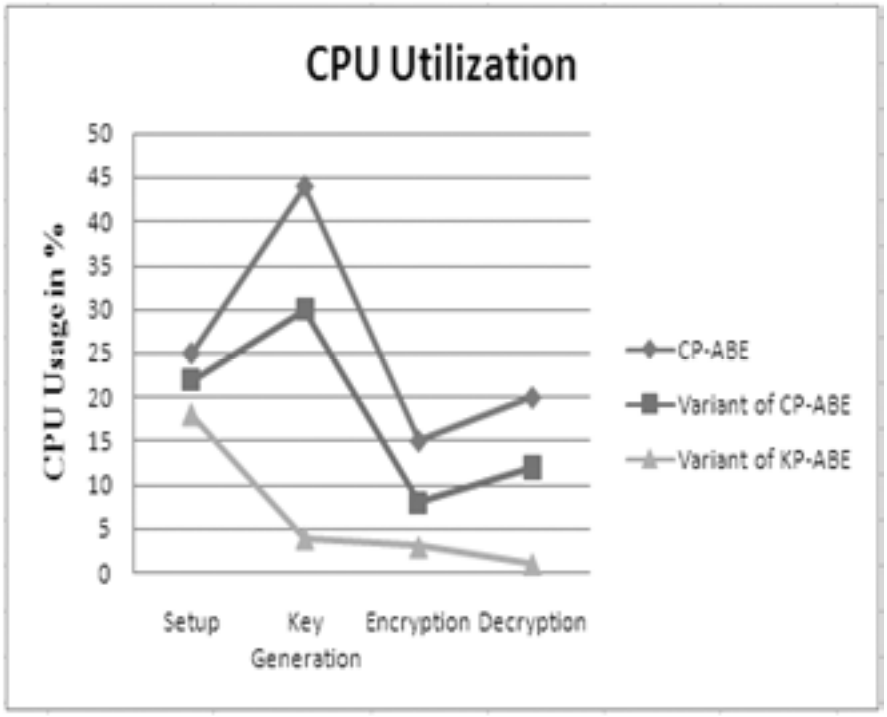

Figure 1: Graph for CPU Utilization

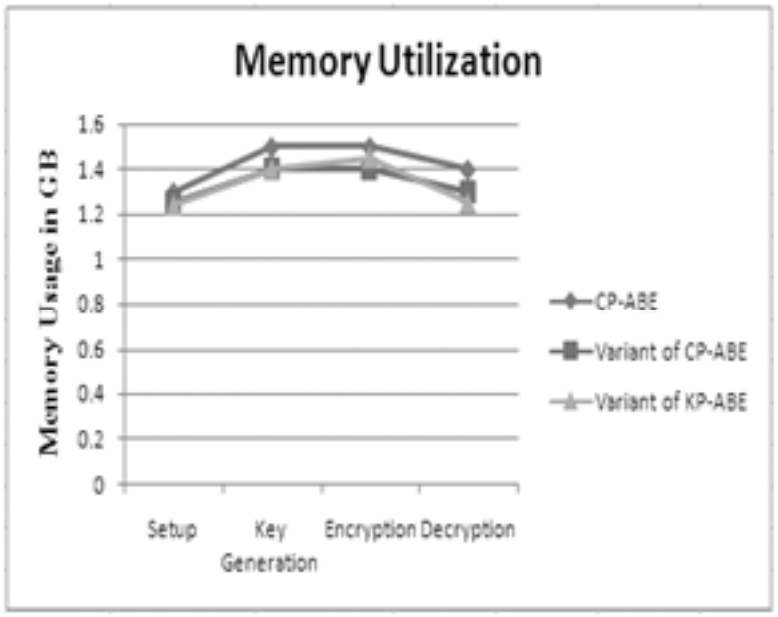

Figure 2: Graph for memory utilization

Figure 2 shows the graphical analysis for the memory utilization of the four phases of CP-ABE, modified CP-ABE and KP-ABE.

It could be seen that the amount of memory used for both the algorithm is almost same for the setup and key generation phase. There is a little amount of variation in the encryption and decryption phase.

5. CONCLUSION

Variants of hierarchical Attribute Based Encryption and comparitive analysis for computing and memory requirements has been carried out. The security analysis shows that the proposed algorithm is resistant against ciphertext only attack and moreover the percentage of CPU being utilized is also minimal. The

amount of memory used for the execution of the algorithm is also less. As a future enhancement reducing the computation power further will be considered.

REFERENCE

[1] A.sahai and B.Waters, "Fuzzy Identity-Based Encryption" Proc.Adv.Cryptology-Eurocrypt, vol.3494, pp.457-473 (2004)

[2] Baodong Qin, Robert H.Deng, Shengli Liu and Siqi Ma, " Attribute-Based Encryption with Efficient Verified Outsourced Decryption", IEEE Transactions on information Forensics and Security, Vol.10, No.7, pp-1384-1393, 2015.

[3] Eric Zavattoni, Luis J.Dominguez Perz and Shigeo Mitsunari, " Software Implementation of an Attribute Based Encryption Scheme", IEEE Transaction on Computers, vol.64, No.5, pp.1429-1441, 2015.

[4] Hai Jiang, Feng Shen, Su Chen, Kuan Ching Li and Young Siu Jeong, "A Secure and Scalable Storage System for Aggregate Data in IoT", ELSEVIER Journal on future Generation Computer Systems, Vol.49, pp. 133-141, 2015.

[5] Liong Jiang and Li Da Xu, "An IoT-Oriented Data Storage Framework in Cloud Computing Platform", IEEE Transaction on Industrial Informatics, Vol.10, No.2, pp. 1443-1451, 2014.

[6] S.Loong Keoh, Sandeep Kumar and Hannes Tschofenig, "Securing the Internet of Things : A Standerdization perspective", IEEE Internet of Things Journal, Vol.1, No.3, pp. 265-275, 2014.

[7] Xinlei Wang, Jianqing Zhang, Eve M. Schooler and Mihaela Ion, "Performance Evaluation of Attribute-Based Encryption: Toward Data Privacy in the IoT", IEEE Symposium on Communication and Information Systems Security, pp. 725-730, 2014.

[8] Xuanxia Yao, Xiaoguang Han and Xiaojiang Du, " A Lightweight Multicast Authentication Mechanism for Small Scale IoT Applicatons, " IEEE Sensors Journal, Vol.13, No.10, pp.3693-3701, 2013.

[9] Xuanxia Yao, Zhi Chena, and Ye Tian. (2014),'A lightweight attribute-based encryption scheme for the Internet of Things',Elsevier Journal On Future Generation Computer Systems, pp. 104–112.

[10] Zhibin Zhou, "Efficient Privacy-Preserving Ciphertext Policy Attribute Based Encryption and Broadcast Encryption", IEEE Transactions on Computers, vol.64, No.1, pp.126-138, 2015.