International Journal of Computer Science and Software Engineering
Volume 1, Number 1 (2015), pp. 1-8

http://www.irphouse.com

Attribute Based Encryption for Multiples Authorities Using Shamir's Secret Sharing Algorithm

Konuru Chandrasekhar[1], A.V.S. Sairam Prasad[2], Jataprolu Anirudh[3] and Eadara Ramakrishna[4]

Department of Computer Science and Engineering, K L University, Andhra Pradesh, India
E-mail: konuru chandrasekhar1, sai.adusumilli@gmail.com2, janirudh93@gmail.com3, ramkychowdary@yahoo.com4

Abstract

We identify users using single attribute in Identity based encryption scheme but here we go for multiple number of attributes in multiple authorities system to determine user identity. Our scheme determines a method to send the message to a particular group of users, registered to a specific set of attributes. An encryptor can choose a particular number of specific attributes and encrypt the message such that only the users with equal number of attributes and having those specific attributes can decrypt the message.

1. Introduction

Identity based encryption may not be entirely realistic since we don't necessarily have a unique string identifier for each person. Instead, we often identify people by their attributes. We might want to send a message to a Particular department in a particular university. We might want to send a message to specific research group in a lab. We might want to send a message to specific group of employees, who works in testing department in a company.

Let us consider an example where we have three authorities say KL-University, IBM-Certification and MIC-Certification. A sender desires to send a message only to the users who are the members of KL-University and certified by IBM. A sender could encrypt a message to be decryptable by anyone who has KL-University ID and IBM-Certification ID. Thus, their scheme allows the sender to encrypt a message for more

than one recipient, and to specify who should be able to decrypt, using his attributes alone.

Sahai and Waters gave a fuzzy solution which could be used for attribute based encryption. In this model, a recipient is defined not by a single string, but by a set of attributes [SW05]. Sahai and Waters presented the following challenge: Is it possible to construct an attribute based encryption scheme in which many different authorities operate simultaneously, each handing out secret keys for a different set of attributes?

2. Our Approach

We approach this method using Shamir secret sharing algorithm. Initially the sender selects the attributes that are to be involved in the encryption and specify their authority. The encryption takes place using Shamir secret sharing algorithm and random number generation functions. The user login and get generated key. He applies LAGRANGE INTERPOLATION method to generate the secret message.

2.1 Challenges and Techniques

Case1

Now consider a case for previous example where X and Y are two students and X has KL University ID and Y has IBM Certification ID .These two together cannot combine their keys to generate keys . The below diagram(1.a) illustrate this case.

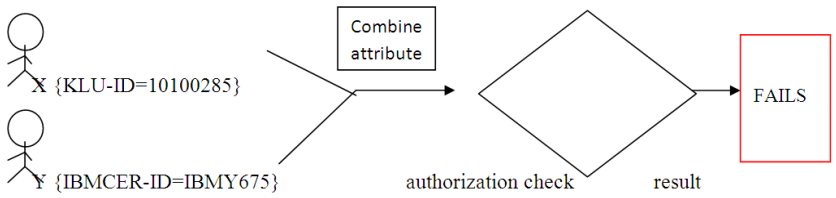


Fig. 1.a

Here the authorization check fails in our scheme because we generate different polynomial equations to distinct users. This process is explained briefly later in upcoming topics.

Case2:

Now consider the case where X and Y are two students . X has two attributes (KLU-ID and IBM-ID) and Y has only one ID(KLU ID).We know that attributes are always publicly known (ex: email, mobile-number). In this case Y try to decrypt the message using X's attributes and fails to get authorized. This can be achieved using a Global Identifier (GID) for each user.GID is distinct for all users and known only to him. The

below diagram explains this case. Let attributes of X ={ IBMCER-ID=IBMY675,KLU-ID=100285} and y={ KLU-ID=10100675}.

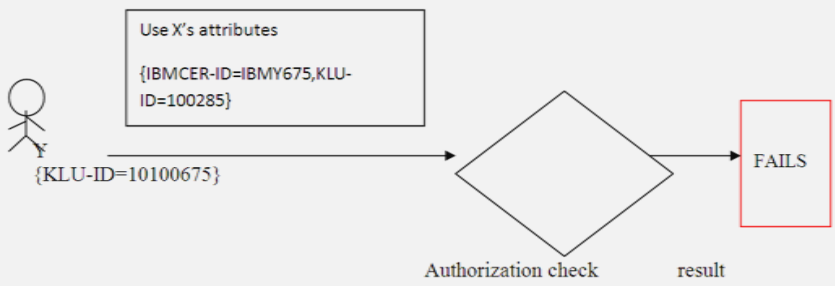


Fig. 1.b

This can be achieved when the user is supposed to submit his GID along with his attributes. Verification is performed at each authority whether the attribute is matched to the respective GID or not.

Case3:
Communication between distinct authorities is not allowed directly. Only communication between central authority and those distinct authorities take place. We make each authority independent to one another. For example, in the above example communication between KLUniversity authority and IBM authority is not allowed. Only central authority communicates with these authorities independently. Thus the direct communication between these authorities is not allowed as they are independent of one another.

2.2 Preliminaries
In this scheme each authority monitors its own attributes. These authorities are independent of one another. Along with these authorities a trusted central authority is also used to generate secret keys. In order to understand this scheme it is essential to understand the below rudiment concepts.

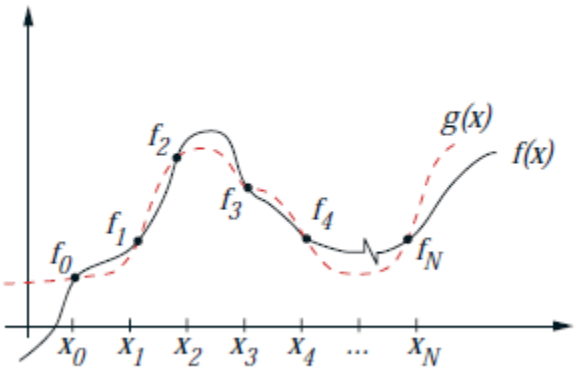
2.3 Shamir's Secret Sharing Algorithm
This can be explained with this example. Let us say that there are 5 authorities and each authority monitors its own attributes. The encryptor uses three attributes from three different authorities and encrypts them. As number of authorities included is three, for each user two distinct random numbers are generated to form polynomial equation .Random numbers generated are distinct from one user to another. Each user has his own randomly generated polynomial equation. Say, X and Y are two authorized candidates. Let the random numbers generated for x and y are (a,b) and (c,d) respectively. Then the equations formed for x and y ax^2+bx+M=0 and

cx^2+dx+M=0 respectively. Here x is the variable and M is the message to be encrypted. Let (i,j,k) are be the random number generated by the three selected authorities that are involved in the equation. We calculate the secret key for each user at that specific authority by substituting these values in the polynomial equation. Let the keys generated for a user X at three authorities be (m,n,r) respectively. It indicates when substitute 'i' in the equation we get m , for 'j' we get n and for 'k' we get r. Now we have three ponts on curve (i,m) , (j,n), (k,r). Consider the above polynomial equations if substitute zero in place of x , we get M. But the decryptor is kept obscure about these equation. He knows only thre points on the curve i.e he knew (i,m),(j,n),(k,r) and he requires (0,?) .To obtain this he goes for LAGRANGE INTERPOLATION.

To explain in a generalized way, If there are n attributes then n-1 degree polynomial equation is formed. Key are generated at each authority, and the obtained keys are used in interpolation method to generated message.

3. Lagrange Interpolation

-> Here we fit N+1 points with nth degree polynomial.



->f(x) = exact function of which only discrete values are known and used to establish an interpolating or approximating function

->g(x)= approximating or interpolating function. This function will pass through all specified ***interpolation points*** (also referred to as ***data points*** or ***nodes***).

Definition:

Given a set of $k + 1$ data points

$$(x_0, y_0), \ldots, (x_j, y_j), \ldots, (x_k, y_k)$$

where no two x_j are the same, the **interpolation polynomial in the Lagrange form** is a linear combination

$$L(x) := \sum_{j=0}^{k} y_j \ell_j(x)$$

of Lagrange basis polynomials

$$\ell_j(x) := \prod_{\substack{0 \le m \le k \\ m \ne j}} \frac{x - x_m}{x_j - x_m} = \frac{(x - x_0)}{(x_j - x_0)} \cdots \frac{(x - x_{j-1})}{(x_j - x_{j-1})} \frac{(x - x_{j+1})}{(x_j - x_{j+1})} \cdots \frac{(x - x_k)}{(x_j - x_k)},$$

where

where $0 \le j \le k$. Note how, given the initial assumption that no two x_i are the same, $x_j - x_m \ne 0$, so this expression is always well-defined. The reason pairs $x_i = x_j$with $y_i \ne y_j$ are not allowed is that no interpolation function L such that $y_i = L(x_i)$ would exist; a function can only get one value for each argument x_i. On the other hand, if also $y_i = y_j$, then those two points would actually be one single point.

For all $j \ne i$, $\ell_j(x)$ includes the term $(x - x_i)$ in the numerator, so the whole product will be zero at $x = x_i$:

$$\ell_{j \ne i}(x_i) = \prod_{m \ne j} \frac{x_i - x_m}{x_j - x_m} = \frac{(x_i - x_0)}{(x_j - x_0)} \cdots \frac{(x_i - x_i)}{(x_j - x_i)} \cdots \frac{(x_i - x_k)}{(x_j - x_k)} = 0.$$

On the other hand,

$$\ell_i(x_i) := \prod_{m \ne i} \frac{x_i - x_m}{x_i - x_m} = 1$$

In other words, all basis polynomials are zero at $x = x_i$, except $\ell_i(x)$, for which it holds that $\ell_i(x_i) = 1$, because it lacks the $(x - x_i)$ term.

It follows that $y_i \ell_i(x_i) = y_i$, so at each point x_i, $L(x_i) = y_i + 0 + 0 + \ldots + 0 = y_i$, showing that L interpolates the function exactly.

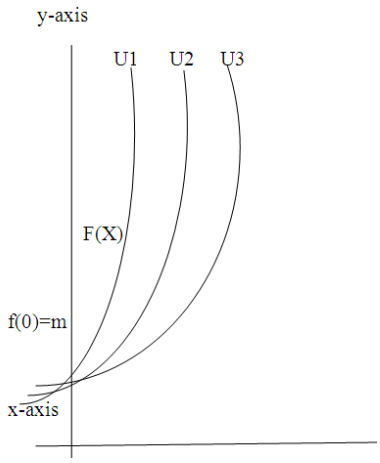
4. Multi Authority Attribute Based Encryption

In multi authority attribute based encryption each user is identified by a set of attributes. These attributes are present in different authorities. Communication between two different authority is not allowed. Along with these authorities there will be a trusted server Central Authority which consists of GID and primary keys of other authorities.

Encryption: During encryption process the user selects the attributes and encrypts the message using the polynomial equations .

Keys Generation : A random prime number is generated based on the message length initially to create a field. If the number of attributes selected is n+1 , then nth degree polynomial curve is generated for each user. The curves generated for each user are different from one another.

The above diagram indicates different curves for different users but all of them meet at the point of secret message i.e., f(0) for every user. This diagrams proves that , any two users cannot combine their attributes when they individually doesn't have the required attributes to decrypt the message.



Each included authority randomly generates a secret number which is to be substituted in the equations of the polynomial to generates keys. If there are m number of attributes included then m secret keys are generated.Each user get different key as each user has his own curve.

Decryption

User enters his GID and attributes to check his authorization. The central authority checks for the matching of the GID for that user central authority checks for the matching of the users GID and his primary attribute key for all authority. The verification for included attributes is performed at the respective authority. If the verification for attributes and GID is successful then he is provided with key. Using these keys he goes for interpolation to get the secret message.

This scheme ensures that only the exact users with his own attributes can decrypt the message. Two users cannot combine their keys to gain the secret keys. A user cannot mask himself as other users i.e., he cannot login using others attributes.

Algorithm

Step1: The Person who encrypts chooses the attributes and encrypts the message M using the Shamir algorithm. A random prime number is selected which defines the field. If n authorities are included in the encryption then n-1 degree distinct polynomial curve is generated for each user.

Step2: Each authority generate a random number. These random numbers are substituted in curve equation to generate secret keys. The user is supplied with the pairs of each authority random number and secret key.

Step3: User goes for interpolation to decrypt the message.

5. Improvements

Query: If an encrypting person encrypts a message using this scheme any user with required number of attributes either included or not in encryption may decrypt the message.

Solution: We generate another set of random number at included authorities. If 'S' be the secret key generated by a authority and t is the random number at the authority. We multiply t with S and supply result to the user. Now he can generate the secret key only if he knows t value.

Improved Algorithm

Step1: The Person who encrypts chooses the attributes and encrypts the message M using the Shamir algorithm. A random prime number is selected which defines the field. If n authorities are included in the encryption then n-1 degree distinct polynomial curve is generated for each user.

Step2: Each authority generate a random number. These random numbers are substituted in curve equation to generate secret keys. The user is supplied with the pairs of each authority random number and secret key.

Step3: The authorities involved in the encryption multiply the secret key with another random number generated by the specific included authority. The user is supplied with the pairs of each authority random number and final secret key along with second set of random numbers for included authorities.

Step4: User goes for interpolation to decrypt the message.

6. Practical Example

Let us assume 5 authorities and we use three of the authorities to include in the encryption. We have these five authorities say KL University(KLU), IBM Certification(IBMC), MIC certification(MICC), City Library , MIC-research centre . We are supposed to send a message to the student of KLU , got certified in IBMC and MICC.

Let X be a user who has the following Ids (KLUID=KLU285, IBMID =IBM604, MICID=MIC249) and the GID= x111. He submits his GID and attributes to the central authority. Central authority checks for match. Each authority checks for matching of the attributes with GID.

As number of authorities included here is 3 for each user u a separate polynomial equation is generated. Let the message to be encrypted be 666. The polynomial equation generated for X be 897x^2+300x+666=y. Let the prime number key be 1321.

Let the random numbers generated by the selected authorities be 1,2 and 3 respectively. Then the keys generated are 542,1018, and 773 respectively. Now the user is kept with the three (x,y) values for his equation. X has the following three (1,542),(2,1018)and (3,773) pairs . He requires (0,?). He uses interpolation to calculate the required message.

References

[1] [BB04] Dan Boneh and Xavier Boyen. Efficient selective-id secure identity based encryption without random oracles. In Proc. of EUROCRYPT 2004, volume 3027, LNCS, 54–73. Springer.
[2] [Wat05] Brent Waters. Efficent identity based encryption without random oracles. In Proc. of EUROCRYPT 2005, volume 3494, LNCS, 114–127. Springer.
[3] https://www.ecri.org/documents/secure/risk_quality_patient_safety.pdf
[4] http://en.wikipedia.org/wiki/Shamir's_Secret_Sharing
[5] http://en.wikipedia.org/wiki/Lagrange_polynomial