Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-6, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

Multi-Authority CP-ABE Scheme with Signature Validation

Pratap Walunj[1] & P. R. Ugale[2]
[1]P.G. Student
[2,] Professor. Dept. Of Computer Engg, SPCOE, Pune, India.

Abstract*: Wireless networks are powerful means of connectivity solution for mobile nodes. But mobile nodes suffer from problem of partitioning or frequent dis-connection of network. To bring robustness and guarantee network connectivity wireless network often uses Disruption-tolerant network (DTN) technologies, but it does not guarantee real time data communication thus external node is used as temporary storage. In real world scenarios there is often a situation where user needs to share data to bunch of people based on their attributes, such that only persons having required authority can view the data. In computer infrastructure this facility is provided by CP-ABE schemes. But, CP-ABE schemes are not fully supported in DTNs because of attribute revocation, key escrow problems that CP-ABE schemes often faces. This paper describes a system which provides support for defining complex access structure over data and also allows user to verify the singer of the document.*

Key Words: *CP-ABE, PKC, IBE.*

1. Introduction

Stored data security is very import for any organization. We have recently familiar with many attacks such as ransomware which attacks on data and put the data security at risk which ultimately results in huge business loss. Data theft is also serious problem in computer / wireless network and there are various encryption techniques which are used to for secure data communication. Wireless networks also suffers from frequent network partitioning and connectivity problems. The problem of frequent network dis-connection and communication problems can be solved using Disruption- tolerant network (DTN) technologies. To enable communication between parties involved DTN uses external node for storing intermediate data incase destination node is not reachable, but providing support for expiration of data and signature verification functionality are vital for the success of the system [4].

In cryptography size of keys plays vital role in effectiveness of scheme and efficiency. There is new branch of cryptography that uses pairing characteristics to design novel cryptosystems. Pairing based cryptographic schemes are found to be more secure and the key sizes required are very minimum compared to non-pairing based cryptographic schemes.

In real world scenarios it is desirable to share data among set of member's those having proper access policies and data owner can decide who have to give permission to view data. This scenario can be implemented using attribute based encryption technique and role based access control [6]. Attribute based encryption uses user attribute for granting access to data, here user attributes can be user designation, access keys, etc. Previously it was difficult to enforce complex access policies, however prior knowledge of total number of users is also not required is advantage of ABE schemes. Every attribute based encryption scheme must be resistant against collision attacks, key update, key escrow. However, using ABE schemes in wireless network poses some issues because of frequent network partitioning and network dis-connectivity. The problem rises from the fact that in wireless network nodes are moving across different network areas sometimes they dis-connect from network and it causes problems when we wanted to update the user keys or we wanted to update data accessibility criteria as data owner might not have connectivity with storage node or user attributes might not have been updated due to dis-connectivity with local authorities [5].

ABE schemes works perfectly when policies and keys are issued by single authority, but when problem is turned into incorporation of multiple authorities then it becomes difficult to define and create access policies and their key management. Therefore, it was become necessary to have a system which is capable of expressing complex access policies [2].

This paper proposes new system that seats on top of ABE schemes and allows data owner to define complex access policies and sign the documents, later any user with adequate access policy can check the data integrity and this system also provides expiration policy for stored data. The rest of the

paper is organized as follows section II covers previous researches on pairing based cryptosystems and section III provides the system architecture of current system, section IV provides security analysis of the proposed system and section V concludes.

2. Literature Survey

Cryptographic algorithms are a means of securely transferring information between involving parties. There are mainly two types of cryptographic technique private key cryptography also known as symmetric key cryptography and other one is public key cryptography also known as asymmetric key cryptography. The difference between them is based on whether the encryption and decryption keys are similar or different? Former is known to be private key cryptography and later known as secret key cryptography. For cryptographic algorithms key sharing was mainly important concern on which whole data security was dependent. Diffie-Hellman key exchange protocol [1] was popular protocol and it is still in use today. Cryptographic algorithms works on assumption that algorithm name, its implementation and all required public parameter are known to everyone, yet it is difficult to recover original message without exact decryption key [07]. However, when data owner needs to share information to number of users then normal cryptographic does not support it. There are another category of cryptographic algorithms which as known as attribute based encryption which enables data owner to share their data among number of user based on defined policies.

Attribute based encryption techniques falls into two categories CP-ABE and KP-ABE [01]. CP-ABE schemes embeds access-structure in the cipher-text, whereas KP-ABE schemes embeds attributes inside user keys. Like other schemes ABE schemes also rely on assumption that the problem does not reduce known hardness assumptions. Waters proposed efficient and secure CP-ABE scheme and their algorithm running time and key size was very low. Many of the threshold based attribute based encryption schemes are based on shamir's secret sharing scheme. It shares secret S by dividing them into n number of parts of different sizes and giving them to participants. The original secret S can be recovered only when t number of parties combines their secret shares. The first CP-ABE scheme was proposed by Bethencourt et al. [] which works by making access structure public and user and uses Shamir's threshold based secret sharing scheme. System that uses access structures with AND gates and support for negated attributes was proposed by Cheung et al. Their schemes provides security against chosen ciphertext attack (CCA) [00]. Wang et al. [00] evaluated the performance of CP-ABE and KP-ABE on laptop and smartphone devices and concluded ABE performance is unacceptable for small devices such as smartphones. Yang et al. [08] proposed an improved scheme in CP-ABE which extended multi-authority with attribute revocation. But, authority needed to enhance efficiency.

An efficient user revocation ABE scheme was proposed by Junos et al [00] which utilized broadcast encryption scheme on ABE scheme. The data owner should take full charge of maintaining all the membership lists for each attribute group

An attribute tree is the structure used to present the verifier's request. Attribute tree is the structure used to implement ABE scheme, it also uses bilinear maps and Lagrange interpolation to build a policy for decryption. Each inside node acts like threshold gate and leaves acts as attributes.

Shamir [6] brought the idea of identity based encryption. Identity based encryption uses identity of user as a public key. It provides alternative approach public key infrastructure (PKI). In such schemes there are three entities; a signer, a verifier and a key generator. The signer obtains a private key that corresponds to his identity from a key generator. He signs a message with that key. The verifier uses the identity of the signer to check validity of the signature.

Pairing based cryptography is built around interesting characteristics of curves and advantage of using them is that it reduces communication overhead and provide fine grained access control [06]. Using interesting properties from bilinear maps various researchers developed different types of pairing based cryptographic systems. All of the pairing based cryptographic solutions exploits mapping between Gap group denoted as G_1 and second group denoted as G_2 [03]. BLS [04] is provides shortest key length and it efficient. It allows user to verify singer is authentic. This scheme consist of key generation, singing and verification functions. BLS is short and simple.

Cyclic Groups and Generators

A group *G* is called cyclic if *G* can be generated by a single element g called a generator. A group *G* can have many generators. A set of generators $g^1,...,g^n$ is a set of group elements such that repeated group operation (addition, multiplication) or inverse of g^n on the generators on themselves is capable of producing all the elements in the group [01].

3. SYSTEM ARCHITECTURE

The architecture of this multi-ABE scheme is shown in figure 3.1. The scheme consist of 4 player's main authority, local authority, intermediate storage node and one or many end user. The role of each user discussed below,

Main Authorities: It's responsible for generating unique keys for end user by collaborating with local authorities. This guarantees collision will not take place between user attribute keys.
Authority: Local authorities are like part of organization that is responsible for managing its employees. It defines access policies and responsible for getting user in main flow.

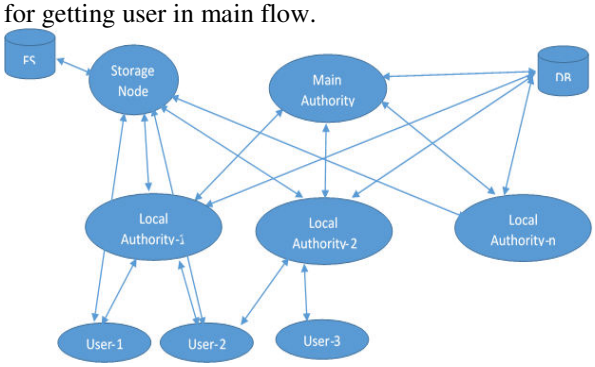

Fig. 3.1 System architecture

Storage node: Node that temporarily stores data until it reaches to receiver in DTN environment. Storage node erases the data based on expiration time set for the data by data owner and it also helps users to verify the integrity of message.
User: Defines access structure tree for data to be shared and shares the data.
There is no cleaver way by which user will know how many authorities are there, also how central authority will know how many local authorities are there. By getting all this initial information then only system will functions correctly.
Assume *T* be the access tree in which each non-leaf node represents threshold gate and n_x be the count of child nodes of node x and t_x be threshold value of node x. Each leaf node denotes attribute and having threshold $t_x = 1$. Each attribute associated with leaf node x is λ_x. If set of attribute ¥ satisfies the tree Tx then it is represented as Tx(¥)=1. For all non-leaf node *y* then we compute $T_y(¥)$. T_y(¥) returns 1 iff at least t_y children's returns 1.
Let G_0 be a bilinear group of prime order p, and *g* is generator of group G_0. A hash function $H:\{0,1\}^* G_0$ is used to associate each attribute with a random group element in G_0.

Public Parameter Setup
During system setup trusted third party chooses a bilinear group G_0 of prime order p with generator g and a hash function H. The public parameter is given by (G_0, g, H).
Main Authority
Main authority chooses a random exponent $\beta \in Z_p$. It sets $h = g^{\beta}$. The public key is *h* and private key *β*.
Local Authority
Each local authority A*i* chooses α_i from Z_p. The public key is set to $e(g, g)^{\alpha}$ and private key is α.

Key Generation
User keys in CP-ABE schemes are composed two components personal key that uniquely identifies user and prevents collision attacks and multiple attribute keys that defines access policies. The personalized key is alone generated by CA.
The intuitive way of sharing the information about user's and number of local authorities is to use single database. It is assumed that the database is managed independently by third party and is secure from un-intended accesses.
Personalized key Generation:
Main authenticates user u_i and from each local authority $A_i \in m$, then main authority chooses σ_i where $i \in m$ and finally it computes $\gamma_k = \sum_{i=1}^{m} \gamma_i$, where γ_i is personalized key of user u_i.

Attribute Key Generation:
To generate attribute keys for user u_i, main authority engages local authorities in two phase communication protocol. It first establishes component $x=(\alpha_i + \gamma_i)\beta$. For creating this shared secret CA shares (γ_i, β) and each local authority A_i shares its σ_i. Local authority A_i then randomly chooses $T \in Z_p$ and computes $g^{x/T}$ and sends it to main authority which then computes $B=T^{1/\beta 2}$ and sends it to local authority A_i. Finally Ai outputs user key component as $D_i = B^T = g^{\frac{(\alpha_i+\gamma_i)\beta}{\tau}}$.
User is expected for combining all the keys to generate all the personalized keys received from given authorities given as $D=\prod_{i=1}^{m} D_i = g^{\frac{(\alpha_1+\cdots+\alpha_m)+r_t}{\beta}}$

Encoding Message
When data owner wants share data with set of user's, then data owner defines access structure using access tree *T* and encrypts message as follows,
For each node *x* in the tree *T*, the algorithm sets the degree of the polynomial q_x to be one less than the threshold value k_x of that node, that is, . For the root node *R*, it chooses a random and sets $s \in Z_p$. Then, it sets other points of the polynomial randomly to define it completely. For any other node *x*, it sets $q_x(0) = q_p(x)$ and chooses other points randomly to completely define q_x. Finally user encrypts the message given as,

$$CT = (\mathcal{T}, \tilde{C} = Me(g,g)^{(\alpha_1+\cdots+\alpha_m)s}, C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C'_y = H(\lambda_y)^{q_y(0)})$$

After the construction of T, the sender stores it to the storage node securely. On receiving any data request query from a user, the storage node responds with to the user.
Message Signing
Waters signature scheme is a three-tuple of algorithms W = (Kg,Sig,Vf). These behave as

follows. Pick random $\alpha \leftarrow Z_p$ and set $A \leftarrow e(g, g)^{\alpha}$. The public key pk is $A \in G_T$. The private key sk is α.

W.Sig(sk, M).

Parse the user's private key sk as $\alpha \in Z_p$ and the message M as a bit string $(m_1, \ldots, m_k) \in \{0,1\}^k$. Pick a random $r \leftarrow Z_p$ and compute

$$S_1 \leftarrow g^{\alpha} \cdot \left(u' \prod_{i=1}^{k} u_i^{m_i}\right)^{r} \quad \text{and} \quad S_2 \leftarrow g^{r}$$

The signature is $\sigma = (S_1, S_2) \in G_2$.

W.Vf (pk, M, σ).Parse the user's public key pk as $A \in GT$, the message M as a bit string $(m1, \ldots, mk) \in \{0,1\}^k$, and the signature σ as $(S1, S2) \in G_2$. Verify that

$$e(S_1, g) \cdot e\left(S_2,\, u' \prod_{i=1}^{k} u_i^{m_i}\right)^{-1} \stackrel{?}{=} A$$

if equation holds so, output valid ; if not, output invalid.

Decoding Message

The message is broadcasted to all users with matching access structure and user can view the data in their dashboard. User decrypts the encoded message with its secret key sk. The algorithm *DEC(CT,sk,x)* performs in a recursive way. We first define a recursive algorithm that takes as inputs a ciphertext *CT*, a private key sk, which is associated with a set of attributes, and a node from the tree T. It outputs a group element from G or G'.

Incase node x is leaf node then,

$$\begin{aligned}
&\mathrm{DecryptNode}(\mathrm{CT}, \mathrm{SK}, x) \\
&= \frac{e(D_x, C_x)}{e(D'_x, C'_x)} = \frac{e(g^{r_t} \cdot H(\lambda_x)^{r_x}, g^{q_x(0)})}{e(g^{r_x}, H(\lambda_x)^{q_x(0)})} \\
&= \frac{e(g^{r_t}, g^{q_x(0)}) \cdot e(H(\lambda_x)^{r_x}, g^{q_x(0)})}{e(g^{r_x}, H(\lambda_x)^{q_x(0)})} \\
&= e(g, g)^{r_t q_x(0)}.
\end{aligned}$$

If node x is not leaf node then, for all nodes that are children of *x*, it calls and stores the output as F_z. Let S_x be an arbitrary -sized set of child nodes such that F_z!=.IΣ . If no such set exists, then the node was not satisfied and the function returns .I. Otherwise compute F_z,

$$\begin{aligned}
F_x &= \prod_{z \in S_x} F_z^{\Delta_{i,S'_x}(0)}, \quad \text{where} \quad \begin{array}{l} i = \mathrm{index}(z), \\ S'_x = \{\mathrm{index}(z) : z \in S_x\} \end{array} \\
&= \prod_{z \in S_x} (e(g, g)^{r_t \cdot q_z(0)})^{\Delta_{i,S'_x}(0)} \\
&= \prod_{z \in S_x} (e(g, g)^{r_t \cdot q_{p(z)}(\mathrm{index}(z))})^{\Delta_{i,S'_x}(0)} \\
&= \prod_{z \in S_x} e(g, g)^{r_t \cdot q_x(i) \cdot \Delta_{i,S'_x}(0)} \\
&= e(g, g)^{r_t \cdot q_x(0)}
\end{aligned}$$

and return the result. The decryption algorithm begins by calling the function on the root node R of the access tree *T*. We observe that if the tree is satisfied by for all. When we set decrypt(CT,sk,R)=$e(g,g)^{r_t s}$, the algorithm decrypts the ciphertext by computing $\tilde{C}/(e(C, D)/A) = M$.

Revocation- As user key are bound to random number, any user key update would require reissuing all user with newly generated keys. To prevent this over-headed operation, this system re-encrypts the ciphertext at storage node and puts a header on it, only the authorized users are given reissued keys. This modified operation drastically improves performance

The operation performed as follows, select for all G_y Ɛ G, select random $K_{\lambda_y} \in \mathbb{Z}_p^*$ and re-encrypt the message as follows,

$$\begin{aligned}
\mathrm{CT}' = \Big(&\mathcal{T}, \tilde{C} = Me(g, g)^{(\alpha_1 + \cdots + \alpha_m)s}, C = h^s, \\
&\forall y \in Y : C_y = g^{q_y(0)}, \\
&C'_y = \left(H(\lambda_y)^{q_y(0)}\right)^{K_{\lambda_y}}\Big).
\end{aligned}$$

Finally it generates message header containing encrypted attribute group and it would only be decrypted by user who having newly arrived keys $K_{\lambda_y} \in \mathbb{Z}_p^*$.

Key Update-

When user joins or leaves particular attribute group he has to forward request to group authority about such event. On receiving such event storage node selects random s' from Z_p. and rencrypt and sends new keys to nonrevoked users the message as follows,

$$\begin{aligned}
\mathrm{CT}' = \Big(&\mathcal{T}, \tilde{C} = Me(g, g)^{(\alpha_1 + \cdots + \alpha_m)(s+s')}, C = h^{s+s'}, \\
&C_i = g^{q_i(0)+s'}, C'_i = \left(H(\lambda_i)^{q_i(0)+s'}\right)^{K'_{\lambda_i}}, \\
&\forall y \in Y \setminus \{i\} : C_y = g^{q_y(0)+s'}, \\
&C'_y = \left(H(\lambda_y)^{q_y(0)+s'}\right)^{K_{\lambda_y}}\Big).
\end{aligned}$$

4. Efficiency and Security Analysis

We define two security games for an adversary *A* and a challenger *C* for the ABE scheme with encryption and decryption outsourcing capabilit.8 In the first game, asks for the secret keys of proxy A, i.e. while in the second game, asks for the secret keys of the user, SKw. These games correspond to the assumption we use in our model that *proxy A and proxy B* are independent of each other and will not collude. The challenger C runs Setup algorithm and gives the adversary *A* the public parameters , while keeping the master secret key to itself. The adversary *A* performs a polynomial

bounded number of queries asking for secret keys Ski I in {1,2, … n} of proxy A. The challenger returns these secret keys to n this phase the adversary submits two equal length plaintexts and from the message space, on which wants to be challenged. The challenger *C* flips a random coin and returns the partial encryption of (i.e. the encryption that does not contain the cryptographic policy) to the adversary . In this phase, outputs a guess b' in {0,11} and wins if b' = b. The advantage of the adversary in attacking the scheme is |Pr[b' = b]-(1/2)|.

The above construction requires that each authority store N -1 seeds and run N - 1 invocations of our anonymous key issuing protocol for each user. The user in turn has to store |Ak| + 1 values for each authority k. The main overhead is on the side of the authority, and even so, it seems a fairly small cost to pay in exchange for guaranteeing security when any N - 1 out of N authorities are corrupted

5. Conclusion

The study of different attribute based encryption techniques given us the limitations when they were implemented in wireless networks where frequent dis-connection are common and nodes are constantly moving. The literature also provided the use of pairing based cryptography to be suitable for implementing attribute based encryption schemes in wireless networks. This system provides support for data owner to share the data based on access policies and data owner can model complex policies. It also allows data owner to setup expiration time that specifies when the message will get expire and also allows data retrieval user to verify the integrity of the message. The security analysis also suggested that it provides secure data sharing schemes. ABE schemes useful as it gives fine grained control on decryptors. This scheme also provides guard against collusion attacks and it also provides solution for key update problem.

6. References

[1]. Atkin and F. Morain, "Elliptic curves and primality proving", Mathematics of Computation, 1993.
[2]. P. Barreto, S. Galbraith, C. and M. Scott, "Efficient pairing computation on super singular abelian varieties", Designs, Codes and Cryptography, 2007.
[3]. P. Barreto, H. Kim, B. Lynn and M. Scott, "Efficient algorithms for pairing-based cryptosystems", Advances in Cryptology – CRYPTO 2002, Lecture Notes in Computer Science, 2002.
[4]. Dan Boneh. The decisional diffie-hellman problem. In Third Algorithmic Number Theory Symposium, pages 48–63. Springer-Verlag, 1998.
[5]. Whitfield Diffie and Martin E. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, IT-22(6):644–654, 1976.
[6]. Adi Shamir. Identity-based cryptosystems and signature schemes. In Crypto '84, LNCS Vol. 196, pages 47–53. Springer, 1985.
[7]. W. Diffie and M. Hellman, "New directions in cryptography", IEEE Transactions on Information Theory, 22 (1976), 644–654.
[8]. S. Galbraith, "Pairings", Ch. IX of I. Blake, G. Seroussi and N. Smart, eds., Advances in Elliptic Curve Cryptography, Cambridge University Press, 2005.
[9]. S. Galbraith, K. Harrison and D. Soldera, "Implementing the Tate pairing", Algorithmic Number Theory: 5th International Symposium, ANTS-V, Lecture Notes in Computer Science, 2369 (2002), 324–337.
[10]. I. Niven, H. Zuckerman and H. Montgomery, An Introduction to the Theory of Numbers, 5th edition, Wiley, 1991.
[11]. J. Silverman, The Arithmetic of Elliptic Curves, Springer, 1986.
[12]. T. Okamoto and D. Pointcheval. The gap-problems: A new class of problems for the security of cryptographic schemes. In Public Key Cryptography, pages 104–118, 2001.
[13]. R. L. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. Lecture Notes in Computer Science, 2248:552+, 2001.
[14]. F. Brezing and A. Weng. Elliptic curves suitable for pairing-based cryptography. http://eprint.iacr.org/2003/143.