# An Efficient Hierarchical Multi-Authority Attribute Based Encryption Scheme for Profile Matching using a Fast Ate Pairing in Cloud Environment

Balaji Chandrasekaran, Student member, ACM, Yasuyuki Nogami, Member, IEEE and Ramadoss Balakrishnan, Member, IEEE

Original scientific paper

***Abstract*—In cloud environment, profile matching is a key technique in applications such as health care and social networks. Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is a suitable technique for data sharing in such environments. In this paper, we propose an asymmetric pairing based Hierarchical Multi-Authority CP-ABE (HM-CP-ABE) construction for profile matching. We utilize the fast Ate pairing to make the proposed HM-CP-ABE scheme efficient. The performance analysis of the proposed scheme shows improved efficiency in terms of computational costs for initialization, key generation and encryption using ELiPS library when compared with existing works.**

***Index Terms*—Asymmetric Pairing, Attribute based encryption, Cloud computing, CP-ABE, Multi-Authority.**

## I. INTRODUCTION

Attribute based encryption (ABE) supports both confidentiality and fine-grained access control with a single encryption for data sharing in cloud environment. Profile matching process based on ABE is one of the typical applications of cloud environments such as health care [1], social networks [2], etc. Profile matching is a technique to find people with similar characteristics. In spite of the advantages, it has some severe problems like risk of privacy leakage. ABE scheme is the solution many researchers have put forwarded for these problems. The ABE scheme is categorized into two: Single-Authority ABE (SA-ABE) [3] and Multi-Authority ABE (MA-ABE) [4], [5], [6], [7].

In MA-ABE scheme, there are two sub-categories: with a Central Authority (CA) and without a Central Authority (CA). The first Single-Authority ABE (SA-ABE) scheme [3] was proposed by Sahai and Water. The major drawbacks of single trusted authority scheme are single point failure, key management and performance bottleneck problem. To overcome the above mentioned problems, Luo et al. [4] proposed a Hierarchical Multi-Authority Ciphertext Policy ABE (HM-CP-ABE) for key management based on Shamir secret sharing scheme to avoid key leakage because of multiple user's key sharing in addition to ruling out of single point failure and performance bottleneck problem. They use symmetric pairing for HM-CP-ABE construction. Yang et al. [7] proposed a scheme to improve privacy and security in decentralizing multi-authority attribute-based encryption in cloud computing. Most often existing ABE schemes are constructed from bilinear pairings which makes an algorithm inefficient due to the high computational complexity of bilinear pairing. Therefore, the main focus of this paper is reducing the computation cost of bilinear pairing operations to improve the efficiency of the HM-CP-ABE scheme.

### A. Our Contributions

The contributions of this paper are highlighted as follows:

1. We propose a more efficient HM-CP-ABE which is based on an asymmetric pairing construction that eliminates the key leakage issue, single point failure and performance bottleneck problem.
2. We apply fast Ate pairing [8] in this asymmetric pairing based HM-CP-ABE construction to reduce its computational costs for initialization, key generation and encryption and then compared with HM-CP-ABE [4], [5], [6], [7].

### B. Paper Organization

The rest of this paper is organized as follows. Section II deals with the preliminaries. Section III explains with fast Ate pairing. Section IV explains the system model. Section V presents the proposed HM-CP-ABE scheme based on asymmetric pairing using Section III. Section VI presents with the experimental results and discussions. Section VII concludes the paper.

## II. PRELIMINARIES

### A. Shamir's secret sharing scheme

Shamir's secret sharing scheme, for $t$-out-of-$n$ secret sharing [9] consists of the following steps.

Manuscript received February 10, 2018; revised March 29, 2018. Date of publication June 7, 2018. Prof. Franco Chiaraluce has been coordinating the review of this manuscript and approved it for publication.

B. Chandrasekaran (corresponding author) and R. Balakrishnan are with the Department of Computer Applications, National Institute of Technology, Tiruchirappalli, TamilNadu, India (e-mails: cbalaji1988@gmail.com, brama@nitt.edu). Y. Nogami is with the Graduate School of Natural Science and Technology, Okayama University, Okayama, Japan (e-mail: yasuyuki.nogami@okayama-u.ac.jp).

Digital Object Identifier (DOI): 10.24138/jcomss.v14i2.461

1845-6421/06/461 © 2018 CCIS

First, choose a large prime $p$, and let field $\mathbb{F} = \mathbb{Z}/p\mathbb{Z}$ . To share secret $x$ as

$$x \Longrightarrow (x_1, x_2, \dots, x_n)$$

do the following:

1. Choose coefficients $f_1, \dots, f_{t-1} \in \mathbb{Z}/p\mathbb{Z}$, which are to be the coefficients of degree $t-1$ polynomial $f$.
2. Let $f(z) = f_0 + f_1 \cdot z + \dots + f_{t-1} \cdot z^{t-1}$ , where $f_0 = x$.
3. Give $f(i)$ to party $i$, $i = 1,2, \dots, n$.

Second, when $t$ parties have a secret, then we have $t$ points on the curve $\lneq (t-1)$ degree polynomial, so by the fact above, we get unique coeffecients to a degree $\lneq (t-1)$ polynomial. The secret is coefficient $f_0$ .

Formalizing, we use Lagrange Interpolation over a finite field. Given $(i, x_i)$ for $i \in \mathcal{G}, \mathcal{G} \subseteq \{1,2, \dots, n\}$,

$$f(z) = \sum_{i \in \mathcal{G}} x_i \prod_{j \in \mathcal{G}, j \neq i} \frac{z-j}{i-j} \quad (1)$$

This is a linear system in $t$ unknowns, the coefficients $f_k$, with $t$ equations. The existence of a unique solution is guaranteed by the fact stated above. So Gaussian elimination can be used to solve. Then,

$$x = f_0 = f(0) = \sum_{i \in \mathcal{G}} x_i \prod_{j \in \mathcal{G}, j \neq i} \frac{-j}{i-j} \quad (2)$$

Letting $\prod_{j \in \mathcal{G}, j \neq i} \frac{-j}{i-j} = c_i$, we have that $x = \sum_{i \in \mathcal{G}} c_i x_i$. Note that $c_i$ is a constant independent of $x_i$'s. Thus we can compute $c_i$, $i \in \mathcal{G}$ ahead of time without knowing the $x_i$'s, and then in linear time can find $x$, the secret, once we have the $x_i$'s. So recoverability is quite efficient in this case.

Based on the Shamir's secret sharing properties we can ensure that recovering of $n$ shares to get the secret, no $n-1$ shares provide any information about the secret. Therefore schemes using Shamir's secret sharing properties have computational security.

### B. *Decisional Bilinear Diffie-Hellman (DBDH) assumption*

Let $\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{z} \in \mathbb{Z}_p$ be chosen at random, $G$ be the group of prime order $q$ and $g$ is the generator of the group $G$. DBDH problem [10] is a problem that no polynomial time adversary is able to distinguish the tuple $(g^{\mathrm{a}}, g^{\mathrm{b}}, g^{\mathrm{c}}, \mathrm{e}(g,g)^{\mathrm{abc}})$ from the tuple $(g^{\mathrm{a}}, g^{\mathrm{b}}, g^{\mathrm{c}}, \mathrm{e}(g,g)^{\mathrm{z}})$ with a non-negligible advantage. This can be formalized as follows:

$$\Big(\big|Pr\big[\mathcal{A}\big(g, g^{\mathrm{a}}, g^{\mathrm{b}}, g^{\mathrm{c}}, \mathrm{e}(g,g)^{\mathrm{abc}}\big) = 0\big] - Pr\big[\mathcal{A}\big(g, g^{\mathrm{a}}, g^{\mathrm{b}}, g^{\mathrm{c}}, \mathrm{e}(g,g)^{\mathrm{z}}\big) = 0\big]\big| \geq \varepsilon\Big)$$

## III. Fast Ate Pairing

Nogami et al., [8] proposed a faster Ate pairing $e$ of embedding degree 12 as follows:

$$e: G_2 \times G_1 \rightarrow F^*_{q^{12}} / (F^*_{q^{12}})^r \quad (3)$$

where $G_1 = E[r] \cap Ker(\emptyset_q - [1])$ and $G_2 = E[r] \cap Ker(\emptyset_q - [q])$ are the two source groups and $r$ satisfies $r \mid (q^{12} - 1)$ and $r \nmid (q^i - 1), i < 12$. $E[r]$ be the set of $r$-torsion points of $E$. Let $[1]$ and $[q]$ denote the 1 and $q$ times of scalar multiplications for a rational point respectively. The Frobenius endomorphism $\emptyset_q$ is defined as $\emptyset_q: E \longrightarrow E, (x,y) \longmapsto (x^q, y^q)$. Let $G_1 \subset E(F_q)$ and $G_2 \subset E(F_{q^{12}})[r]$ be the two source groups, $s = t-1$, where $t$ denotes the Frobenius trace of $E(F_q)$. It satisfies the bilinearity and Non-degeneracy properties. The Ate pairing $e$ is defined as follows:

$$e(Q,P) = f_{s,Q}(P)^{(q^{12}-1)/r}, \forall P \in G_1, \forall Q \in G_2 \quad (4)$$

### A. *Sextic Twist*

The sextic twist is based on Barreto-Naehrig (BN) curve which is a class of non-supersingular pairing friendly curves. The BN curve is defined as $E/F_p: y^2 = x^3 + b, b \in E(F_p)$ with embedding degree 12. The characteristic and Frobenius trace are given by $p(\chi) = 36\chi^4 - 36\chi^3 + 24\chi^2 - 6\chi + 1$ and $t(\chi) = 6\chi^2 + 1$ respectively. Let $\chi$ be an integer. The sextic twisted curve over $F_{p^2}$ is given by $E'/F_{p^2}: y^2 = x^3 + b/v$, where $v$ is a quadratic and cubic non-residue in $F_{p^2}$.

The isomorphism $\psi$ is defined as

$$\psi: E'(F_{p^2})[r] \longrightarrow G_2 \subset E(F_{q^{12}})[r], \\ (x,y) \longmapsto (xv^{1/3}, yv^{1/2}). \quad (5)$$

Nogami et al., [8] used an isomorphic substitution from $G_2 \subset E(F_{q^{12}})$ into $G_2'$ in subfield–twisted elliptic curve $E'(F_{p^2})$. It speeds up scalar multiplications over $G_2$ and eliminates denominator calculations in Miller’s algorithm. Fast Ate pairing proposed by Nogami et al., [8] improves Miller’s algorithm about 30% than BKLS algorithm [11] using proper subfield arithmetic operations. Nogami et al., [8] computes the total cost of Ate pairing in 5.16 ms and 13.3 ms

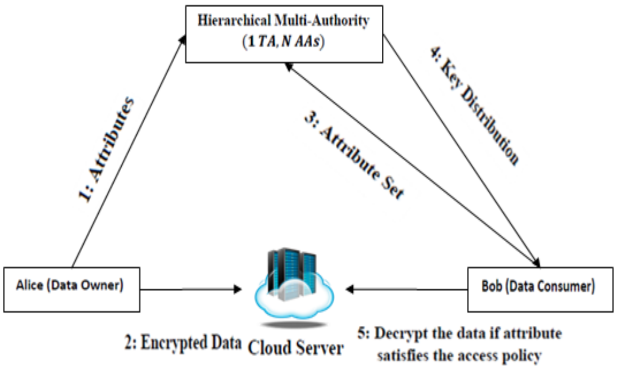

Fig. 1. System model for profile matching process.

## IV. System Model

The system model of profile matching process is shown in Fig. 1. The system model consists of Cloud Server (*CS*), Trusted Authority (*TA*), Attribute Authority (*AA*), data owner (Alice) and data consumer (Bob). This scheme consists of one fully trusted $TA$ and $N\ AAs$ honest-but-curious. The *CS* stores the encrypted data about user's information which includes name, gender, photo, video, contact, hobbies and so on. *TA* is authoritative for the initialization phase and the key generation phase, access policy distribution for each *AA*, approval and administration of subordinate *AA*. Every AA is authoritative for the administration of part attribute set and subordinate AA and authoritative for the key generation and distribution. The system model is used as hierarchical structure of multi-authority [4] scheme. The *TA* is the highest hierarchy and other *AAs* are lower hierarchy. Data consumer can decrypt the data or message only if his/her attributes satisfies the access policy.

## V. Proposed HM-CP-ABE

Fast Ate pairing algorithm [8] is applied in the proposed asymmetric pairing based HM-CP-ABE constructions. Let bilinear map $e: G_1 \times G_2 \rightarrow G_T$, $G_1\ and\ G_2$ be the bilinear groups of prime order $p$. Let $g_1$ and $g_2$ be the generators of groups $G_1\ and\ G_2$ respectively. The $TA$ will allot a unique Global Identifier ($GID$) and Assign an Identifier ($AID$) for all users and each $AA$ respectively. Usually $GID$ is a digital signature string and verified by all $AA$s and cannot be obtained illegally.

$Initialization$: This algorithm is executed by the $TA$ and randomly chooses the parameter $\alpha$ and $\{\beta_1, \beta_2, \ldots, \beta_{dep}\} \in \mathbb{Z}_p$, where $dep$ is number of recursive times in key structure. It produces public key ($PK$) and master secret key ($MSK$) as the output which is calculated in (6) and (7) as follows:

$$PK_0 = \left\{\begin{matrix} G_1, G_2, h_1 = g_1^{\beta_1}, h_2 = g_1^{\beta_2}, h_3 = g_2^{\beta_1}, h_4 = g_2^{\beta_2}, \\ f_1 = g_1^{\frac{1}{\beta_1}}, f_2 = g_1^{\frac{1}{\beta_2}}, f_3 = g_2^{\frac{1}{\beta_1}}, f_4 = g_2^{\frac{1}{\beta_2}}, e(g_1, g_2)^{\alpha} \end{matrix}\right\} \tag{6}$$

$$MK_0 = \{\beta_1, \beta_2, g_1^{\alpha}, g_2^{\alpha}\} \tag{7}$$

After $Initialization$, suppose $AA$ manages the attribute set $\wedge = \{A_0, A_1, A_2, \ldots, A_n\}$, where $A_0$ is the first layer attribute, $A_i$ is the second layer attribute (i.e., $dep = 2$), $a_{i,j}$ is the $j^{th}$ attribute in the attribute set $A_i$, $m$ is the number of attributes in the attribute set $A_i$. The master key of the level 1of $AA$ is calculated in (8) as follows:

$$MK_1 = \left\{\begin{matrix} \wedge, D = g_2^{\frac{\alpha+r}{\beta_1}}, D_{i,j} = g_1^{r_i}.H(s_{i,j})^{r_{i,j}}, \\ D'_{i,j} = g_1^{r_{i,j}}, E_i = g_2^{\frac{r+r_i}{\beta_2}} \end{matrix}\right\} \tag{8}$$

where $r \in_{randomly} \mathbb{Z}_p$ is used to represent $\wedge$ and $r_{i,j} \in_{randomly} \mathbb{Z}_p$ is used to represent $a_{i,j} \in A_i$ $(0 \leq \mathrm{i} \leq \mathrm{n}, 1 \leq \mathrm{j} \leq \mathrm{m})$ and $E_i$ is used to convert node and decrypt cross set matching for attributes. The sub $AA$ is authorized by level 1 $AA$. The master key of $AA_k + 1$ is calculated in (9) as follows:

$$MK_2 = \left\{\begin{matrix} \widehat{\wedge}, \widetilde{D} = D \cdot f_3^{\tilde{r}}, \widetilde{D}_{i,j} = D_{i,j} \cdot g_1^{\tilde{r}_i} \cdot H(s_{i,j})^{\tilde{r}_{i,j}}, \\ \widetilde{D}'_{i,j} = D'_{i,j} \cdot g_1^{\tilde{r}_{i,j}}, \tilde{E}_i = E_i \cdot f_4^{r+r_i} \end{matrix}\right\} \tag{9}$$

where the attribute set of Sub $AA$ is $\widehat{\wedge} \subset \wedge$, $\tilde{r} \in_{randomly} \mathbb{Z}_p$ is used to represent $\widehat{\wedge}$ and $\tilde{r}_{i,j} \in_{randomly} \mathbb{Z}_p$ is used to represent $\hat{a}_{i,j} \in A_i$ $(0 \leq i \leq n, 1 \leq j \leq m)$ and $E_i$ is used to convert node and decrypt cross set matching for attributes.

$Key\ Generation$: The $AA_k$ of Alice attribute set is represented as

$$Au_{Alice}^{(k)} = \{Au_{Alice\_0}{}^{(k)}, Au_{Alice\_1}{}^{(k)}, \ldots, Au_{Alice\_n}{}^{(k)}\}$$

where $Au_{Alice\_0}{}^{(k)}$ is the set of individual attributes and $Au_{Alice\_1}{}^{(k)}$ to $Au_{Alice\_n}{}^{(k)}$ is the subset of attributes with depth 2.

$$Au_{Alice\_i}{}^{(k)} = \{a_{i,0}^{(k)}, a_{i,1}^{(k)}, \ldots, a_{i,j}^{(k)}, \ldots, a_{i,m}^{(k)}\}$$

where $a_{i,j}^{(k)}$ is the $j^{\mathrm{th}}$ attribute in $Au_{Alice\_i}{}^{(k)}$ and $m$ is the number of attributes in $Au_{Alice\_i}{}^{(k)}$. $AA_k$ computes Alice private key component (i.e., $P_{SK}(\,u) = \alpha_{k,u}$) with the help of pseudo random function $P_{SK}(\,)$ according to $GID$ and $AID_k$. It chooses $ru^{(k)}$ and $ru_i^{(k)} \in_{randomly} \mathbb{Z}_p (i = 1,2, \ldots, n)$ for users and each attribute subset respectively. For set $Au_{Alice\_0}{}^{(k)}$, set $ru_0^{(k)} = ru^{(k)}$ and $ru_{i,j}^{(k)} \in_{randomly} \mathbb{Z}_p (0 \leq \mathrm{i} \leq \mathrm{n}, 1 \leq \mathrm{j} \leq \mathrm{r})$ for each attribute $a_{i,j}^{(k)}$ in $Au_{Alice\_i}{}^{(k)}$. The $k^{\mathrm{th}}$ $AA$ of Alice sub key is computed as follows:

$$SK_{Alice}^{(k)} = \left\{\begin{matrix} Au_{Alice}^{(k)}, D_{Alice}^{(k)} = g_2^{\frac{\alpha_{k,u}+ru^{(k)}}{\beta_{k,1}}}, \\ D_{i,j}^{(k)} = g_1^{ru_i^{(k)}}.H(a_{i,j}^{(k)})^{ru_{i,j}^{(k)}} \end{matrix}\right\} \tag{10}$$

$$u'^{(k)}_{i,j} = \left\{\begin{matrix} g_1^{ru_{i,j}^{(k)}} (0 \leq i \leq n, 1 \leq j \leq r), \\ Eu'^{(k)}_i = g_2^{\frac{ru^{(k)}+ru_i^{(k)}}{\beta_{k,2}}} \end{matrix}\right\} \tag{11}$$

The two local master keys of $AA_k$ are $\beta_{k,1} = \frac{\beta_1}{\alpha+r+\hat{r}}$ and $\beta_{k,2} = \frac{\beta_2}{r+r_i+\hat{r}+\hat{r}_i}$. The Alice master key is computed as follows:

$$SK_{Alice} = \left\{\begin{matrix} \{SK_{Alice}^{(k)}\}_{k=1}^{K}, \\ D_{Alice} = g_2^{(\alpha+\sum_{k=1}^{K} \alpha u^{(k)})/\sum_{k=1}^{K} \beta_{k,1}} \end{matrix}\right\} \tag{12}$$

where $D_{Alice}$ is used to meet the Alice access structure to decrypt the file. $D_{Alice}$ is issued by the $TA$.

$Encryption$: $AA_k$ generates the access tree $T^{(k)}$ which is based on the characteristics of Alice attributes to $AA_k$. From the root R in $AA_k$, every node $x^{(k)}$ in $T^{(k)}$ has a corresponding polynomial $q_x$. The order of $q_x$ is calculated as follows:

$$Order\ of\ q_x = \begin{cases} k_x - 1, & x^{(k)} \text{ is a non} - \text{leaf node} \\ 0, & x^{(k)} \text{ is a leaf node} \end{cases}$$

$$For\ every\ node\ x^{(k)},$$

$$x^{(k)} = \begin{cases} q_R(0) = \theta, \theta \in Z_p, & x^{(k)} \text{ is a root node} \\ q_{parent(x)}\big(index(x)\big), & otherwise \end{cases}$$

where $parent(x)$ be the parent of node $x^{(k)}$ and $index(x)$ returns the unique value associated with node $x^{(k)}$.

The $AA_k$ ciphertext is calculated as follows:

$$CT_{Alice}{}^{k} = \left\{\begin{array}{c} T^{(k)}, C^{(k)} = \{h_1^{\theta}\}, \bar{C}^{(k)} = \{h_2^{\theta}\}, \\ \forall y^{(k)} (\text{leaf node}) \in Y^{(k)}: C_y^{(k)} = g_2^{q_y(0)}, \\ C'^{(k)}_y = H\left(att\big(y^{(k)}\big)\right)^{q_y(0)}, \\ \forall x^{(k)} (\text{non} - \text{leaf node}) \in X^{(k)}: \hat{C}_x^k = h_2^{q_x(0)} \end{array}\right\} \tag{13}$$

where $att\big(y^{(k)}\big)$ be the attributes linked with the leaf node in $T^{(k)}$. Alice finally gets the ciphertext of message $M$ and access policy after calculating the ciphertext of $k-1$ $AA$ which is similar to the $AA_k$ ciphertext:

$$CT = \left\{\begin{array}{c} \{T^{(k)}\}_{k=1}^{K}, \widetilde{CT}_{Alice} = M.e(g_1, g_2)^{\alpha\theta}, \\ \{CT_{Alice}{}^{k}\}_{k=1}^{K} \end{array}\right\} \tag{14}$$

Alice sends the ciphertext $CT$ to the server.

$Decryption$: A user Bob decrypts the $AA_k$ ciphertext of Alice $CT_{Alice}{}^{k}$ which is based on $T_{Alice}^{(k)}$ with the help of Bob secret key $SK_{Bob}^{(k)}$ which is based on Bob attributes set, if Bob attributes satisfy the access structure of the ciphertext $CT_{Alice}{}^{k}$.

If $x \in Y^{(k)}$ is a leaf node, then $DecryptNode\big(CT_{Alice}{}^{k}, SK_{Bob}^{(k)}, x, i\big)$ should be computed as follows.

$$DecryptNode\Big(CT^{(k)}, SK_{u'}^{(k)}, x, i\Big) = \frac{e(D_{i,j}^{(k)}, C_x^{(k)})}{e(D'^{(k)}_{i,j}, C'^{(k)}_x)}$$
$$= e(g_1, g_2)^{ru_i^{(k)}.q_x(0)} \tag{15}$$

If $x \notin Y^{(k)}$ is a non-leaf node, then $DecryptNode\big(CT_{Alice}{}^{k}, SK_{Bob}^{(k)}, x, i\big)$ should be computed as follows:

We can calculate $F_x$ of node $x$ by using Lagrange interpolation method, $F_x^k = \prod F_z^{\Delta iz, S_z'}$, where $iz = index(z), S_z' = \{index(z): z \in B_x\}$ and Lagrange coefficient is $\Delta iz, S_z' = \prod \frac{0-jz}{iz-jz}$.

Through upward recursion, we can obtain the value of $DecryptNode\big(CT_{Alice}{}^{k}, SK_{Bob}^{(k)}, x, i\big)$ of root $R$ as follows:

$$F_R^{(k)} = \begin{cases} e(g_1, g_2)^{ru_i^{(k)}.q_x(0)} = e(g_1, g_2)^{ru_i^{(k)}}.\theta, & i \neq 0 \\ e(g_1, g_2)^{ru^{(k)}.q_x(0)} = e(g_1, g_2)^{ru^{(k)}}.\theta, & i = 0 \end{cases}$$

where $i \neq 0$, the transformation of $F_R^{(k)}$ is:

$$F^{(k)} = \frac{e\big(\hat{C}_r^k, E_i^{(k)}\big)}{F_R^{(k)}}$$
$$= e(g_1, g_2)^{ru^{(k)}.\theta} \tag{16}$$

If the user Bob satisfies the access policy of $K$ $AA$, which means $\{F^{(k)}\}_{k=1}^{K}$ has no null values, then the computation is as follows:

$$Q = \prod_{k=1}^{K} \frac{e\big(C^{(k)}, D_u^{(k)}\big)}{F^{(k)}}$$
$$= e(g_1, g_2)^{\theta.\sum_{k=1}^{K} \alpha^{(k)}} \tag{17}$$

$$e(g_1, g_2)^{\alpha\theta} = \frac{e(\prod_{k=1}^{K} C^{(k)}, D_{user})}{Q}$$

$$= \frac{e({g_1}^{\theta.\sum_{k=1}^{K} \beta_{k,1}}, {g_2}^{(\alpha + \sum_{k=1}^{K} \alpha^{(k)})/\sum_{k=1}^{K} \beta_{k,1}})}{e(g_1, g_2)^{\theta.\sum_{k=1}^{K} \alpha^{(k)}}} \tag{18}$$

The plaintext after decryption is as follows:

$$M = \frac{\widetilde{CT}}{e(g_1, g_2)^{\alpha\theta}} = \frac{M.e(g_1, g_2)^{\alpha\theta}}{e(g_1, g_2)^{\alpha\theta}} = M \tag{19}$$

### A. Proposed Scheme: Proof of Security

A proof of security of the proposed scheme has been recently reported in [10] and is here repeated for the sake of clarity.

**Theorem 1:** *If an adversary can break our construction with non-negligible advantage in the security model under Decisional Bilinear Diffie-Hellman (DBDH) assumption, then a simulator can be constructed to solve the DBDH problem.*

**Proof.** The security game is based on the hardness of the DBDH assumption. Suppose attacker $atk$ can win the FH-CP-ABE game with advantage $\varepsilon$. We construct a simulator $sim$ that can distinguish a DBDH tuple from a random tuple with advantage $\frac{\varepsilon}{2}$. Let $G_1$ and $G_2$ are a source group and $G_T$ is a target group. Let $g_1$ and $g_2$ are generators of $G_1$ and $G_2$ respectively. The challenger chooses the fair binary coin ɦ $\in \{0,1\}$, $g_1 \in G_1, g_2 \in G_2, R \in G_T$ and $a, b, c \in \mathbb{Z}_p$. If ɦ $= 0$, then the challenger defines $T$ to be $e(g_1, g_2)^{abc}$. Otherwise, he sets $T = e(g_1, g_2)^{z}\ or\ R$. The challenger then gives the simulator the DBDH details and then simulator $sim$ now plays the role of challenger in the security game.

**Init:** During the init phase, $sim$ receives the challenge access structure $T^*$ from attacker $atk$.

**Setup:** To provide a public key $PK$ to $atk$, $sim$ randomly chooses $\alpha' \in \mathbb{Z}_p$ and note $\alpha = \alpha' + ab$. It calculates

$e(g_1, g_2)^{\alpha}$ like this: $e(g_1, g_2)^{\alpha} = e(g_1, g_2)^{\alpha'}.e(g_1, g_2)^{ab}$. Finally, $sim$ sends public key $PK$ to $atk$.

**Phase 1:** During this phase, $atk$ submits an attribute set $\mathcal{W}_j \in T$ $(\mathcal{W}_j \notin T^*)$ to $sim$. Simulator $sim$ chooses $ru'^{(k)}, ru_i'^{(k)} \in_{randomly} \mathbb{Z}_p$ and set $ru^{(k)} = ru'^{(k)} - a, ru_i^{(k)} = ru_i'^{(k)} - a$. It can obtain $D^{(k)} = g_2^{\frac{\alpha_{k,u} + ru^{(k)}}{\beta_{k,1}}} = g_2^{\frac{\alpha_{k,u}}{\beta_{k,1}}}.g_2^{\frac{(ru'^{(k)} - a)}{\beta_{k,1}}}$. For each attribute in $\mathcal{W}_j$, $sim$ wants to choose $r_j \in_{randomly} \mathbb{Z}_p$. It computes the rest of the secret key of $k^{th}$ $AA$ as follows: $D_{i,j}^{(k)} = g_1^{ru_i'^{(k)} - a}.H(a_{i,j}^{(k)})^{ru_{i,j}^{(k)}}$. Finally, $sim$ sends the secret key to $atk$.

**Challenge:** The attacker $atk$ submits two equal length messages $\mathrm{m}_1$ and $\mathrm{m}_2$ along with a challenge access structure $\mathcal{A}^*$. $sim$ randomly generates a bit $\hat{\mathrm{h}} \in \{0,1\}$ and computes $CT^*$ as $C_y^{(k)} = g_2^{q_y(0)} = g_2^{\theta} = g_2^{c}, \widetilde{CT} = m_{\hat{\mathrm{h}}}.e(g_1, g_2)^{\alpha\theta} = m_{\hat{\mathrm{h}}}.T.e(g_1, g_2)^{\alpha' c}$. Finally, $sim$ sends the $CT^*$ to $atk$.

**Phase 2:** Same as the Phase 1.

**Guess:** The attacker $atk$ outputs a guess $\hat{\mathrm{h}}'$ of $\hat{\mathrm{h}}$. If $\hat{\mathrm{h}} = \hat{\mathrm{h}}'$, simulator $sim$ guess that $T = e(g_1\,, g_2)^{abc}$. Otherwise, $T$ is a random target group element in $G_T$.

The advantage of the attacker is $\varepsilon$, when $T = e(g_1\,, g_2)^{abc}$. The advantage of the attacker is $\frac{1}{2}$, when $T$ is a random target group element in $G_T$. Finally, the advantage of the simulator in this security game is $\frac{\varepsilon}{2}$.

## VI. Experimental Results and Discussions

In this section, we show the computation cost of initialization phase, key generation phase and encryption for the proposed HM-CP-ABE scheme in cloud environment. We implemented the proposed HM-CP-ABE scheme based on the ELiPS [8] pairing library and GMP6.0.0 library and using the GCC4.8.2 compiler. The implementation uses a Barreto-Naehrig (BN) curve which is class of a non-supersingular pairing friendly curves $y^2 = x^3 + b\,, b \in E(F_p)$ with embedding degree 12. The experimental results are shown in Fig. 2, Fig. 3 and Fig. 4. The number of attributes used in the simulation are {20,40,60,80,100}.

As illustrated in Fig. 2, the proposed scheme reduces the computational cost for initialization phase and we can also show that the results are slightly increasing and approximately following a linear relationship with the number of attributes. The initialization phase is associated with the hierarchical structure of authority and number of attributes handled by $AA$. The schemes [5], [6] adopt the accountability mechanism that utilizes bilinear mappings multiple times. As we have shown in Fig. 2, when the number of attributes are varied, the initialization time of proposed work is less than [4], [5], [6] and [7].

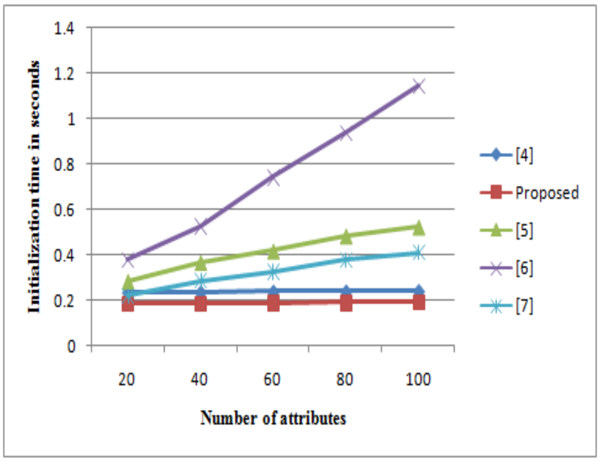

Fig. 2. Comparison of results for initialization time based on Number of attributes.

As illustrated in Fig. 3, the proposed scheme reduces the computational cost for key generation phase and we can also show that the results are gradually increasing and approximately following a linear relationship with the number of attributes. As in [4], each $AA$ handles certain number of attributes. It reduces the complexity of attributes management.

In schemes [5], [6] , all attributes are managed by one authority which is responsible for key generation, thus their computation time is larger. As we have shown in Fig. 3 and Fig. 4, when we varied the number of attributes, the key generation time and encryption time of the proposed work is less than [4] , [5], [6] and [7] respectively.

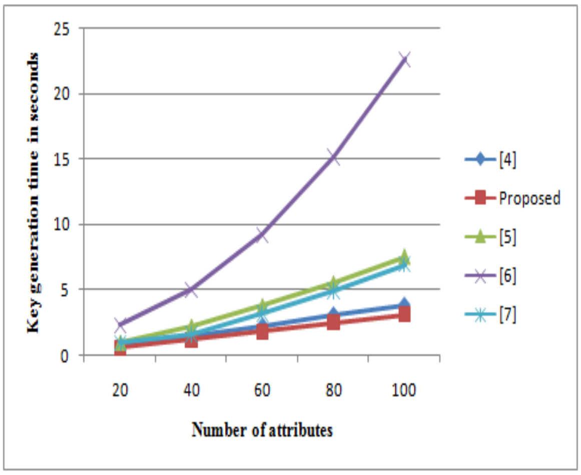

Fig. 3. Comparison of results for key generation time based on Number of attributes.

As in [4], the proposed HM-CP-ABE scheme reduces the workload and communication cost since all pairing operations are computed in server and data owner update their own attributes in $AA$ only, rather than in the entire attribute sets. The results show that the proposed HM-CP-ABE scheme enhances the efficiency and provide fine grained access control in cloud environment.

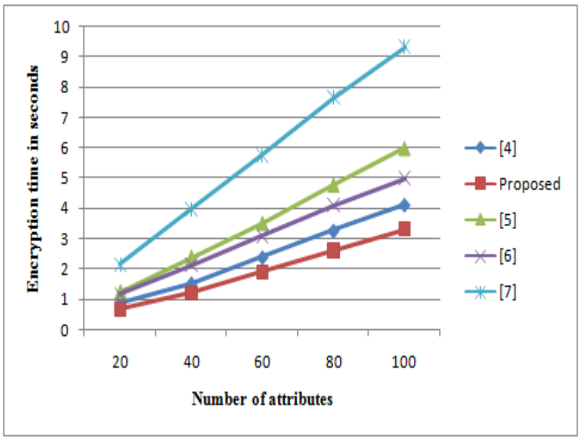

Fig. 4. Comparison of results for encryption time based on Number of attributes.

## VII. Conclusion

In this paper, we proposed a more efficient asymmetric pairing based Hierarchical Multi-Authority CP-ABE (HM-CP-ABE) construction for profile matching. We utilized fast Ate pairing [5] which was based on a Barreto-Naehrig (BN) curve with embedding degree 12. It also speeds up scalar multiplications and eliminates denominator calculations in Miller's algorithm. The proposed HM-CP-ABE scheme is then compared with existing schemes [4], [5], [6], [7]. The experimental results show a mitigation in terms of computational costs for initialization, key generation and encryption.

## Acknowledgment

This work was supported by Ministry of Human Resource Development (MHRD) under the Government of India.

## References

[1] L. Guo, C. Zhang, J. Sun and Y. Fang, "A Privacy-Preserving Attribute-Based Authentication System for Mobile Health Networks," in IEEE Transactions on Mobile Computing, vol. 13, no. 9, pp. 1927-1941, Sept. 2014. doi: 10.1109/TMC.2013.84

[2] W. Dong, V. Dave, L. Qiu and Y. Zhang, "Secure friend discovery in mobile social networks," 2011 Proceedings IEEE INFOCOM, Shanghai, 2011, pp. 1647-1655. doi: 10.1109/INFCOM.2011.5934958

[3] A. Sahai and B. Waters, "Fuzzy identity-based encryption," In Proceedings of the 24th annual international conference on Theory and Applications of Cryptographic Techniques (EUROCRYPT'05), Ronald Cramer (Ed.). Springer-Verlag, Berlin, Heidelberg, 457-473, ACM, 2005. doi: http://dx.doi.org/10.1007/11426639_27

[4] E. Luo, Q. Liu and G. Wang, "Hierarchical Multi-Authority and Attribute-Based Encryption Friend Discovery Scheme in Mobile Social Networks," in IEEE Communications Letters, vol. 20, no. 9, pp. 1772-1775, Sept. 2016. doi: 10.1109/LCOMM.2016.2584614

[5] M. Chase and S.S. Chow, "Improving privacy and security in multi-authority attribute-based encryption." In Proceedings of the 16th ACM conference on Computer and communications security, pp. 121-130. ACM, 2009. doi: http://dx.doi.org/10.1145/1653662.1653678

[6] J. Li, Q. Huang, X. Chen, S.S. Chow, D.S. Wong and D. Xie, "Multi-authority ciphertext-policy attribute-based encryption with accountability." In Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, pp. 386-390. ACM, 2011. doi: https://doi.org/10.1145/1966913.1966964

[7] Y. Yang, X. Chen, H. Chen and X. Du, "Improving Privacy and Security in Decentralizing Multi-Authority Attribute-Based Encryption in Cloud Computing," in IEEE Access, vol. 6, pp. 18009-18021, 2018. doi: 10.1109/ACCESS.2018.2820182

[8] M. Akane, Y. Nogami, and Y. Morikawa, "Fast ate pairing computation of embedding degree 12 using subfield-twisted elliptic curve." IEICE transactions on fundamentals of electronics, communications and computer sciences 92, no. 2 (2009): 508-516. doi: 10.1587/transfun.E92.A.508

[9] X. Huang, Q. Tao, B. Qin and Z. Liu, "Multi-Authority Attribute Based Encryption Scheme with Revocation," 2015 24th International Conference on Computer Communication and Networks (ICCCN), Las Vegas, NV, 2015, pp. 1-5. doi: 10.1109/ICCCN.2015.7288431

[10] B. Chandrasekaran, R. Balakrishnan and Y. Nogami "Secure Data Communication using File Hierarchy Attribute Based Encryption in Wireless Body Area Networks." Journal of Communications Software and Systems [Online], 14.1 (2018): 75-81. doi: http://dx.doi.org/10.24138/jcomss.v14i1.446

[11] A.J. Devegili, M. Scott and R. Dahab, "Implementing cryptographic pairings over Barreto-Naehrig curves." In International Conference on Pairing-Based Cryptography, pp. 197-207. Springer, Berlin, Heidelberg, 2007. doi: https://doi.org/10.1007/978-3-540-73489-5_10

**Balaji Chandrasekaran** received the B.E degree in Computer Science and Engineering from J.J College of Engineering and Technology, Anna University, Chennai in 2009 and the M.E degree in Software Engineering from Anna University, Tiruchirappalli in 2011. Currently, he is a PhD researcher in the Department of Computer Applications at National Institute of Technology, Tiruchirappalli, India. His research interests include: Cryptography, Cloud Computing, and Information Security. He is a student member of ACM.

**Yasuyuki Nogami** graduated from Shinshu University in 1994 and received the PhD degree in 1999 from Shinshu University. He is now an associate professor of Okayama University. His main fields of research are finite field theory and its applications such as recent public key cryptographies. He is now studying about elliptic curve cryptography, pairing-based cryptography, Lattice-based cryptography, pseudo random number generator, Advanced Encryption Standard, and homomorphic encryptions. Recently, he is a member of security research group in Okayama University and particularly focusing on IoT security from the viewpoints of software and hardware implementations. He is a member of IEICE and IEEE .

**Ramadoss Balakrishnan** received the M.Tech degree in Computer science and Engineering in 1995 from the Indian Institute of Technology, Delhi and the PhD degree in Applied Mathematics in 1983 from Indian Institute of Technology, Bombay. Currently he is working as a Professor of Computer Applications at National Institute of Technology, Tiruchirapalli. His research interests include: Software Testing Methodologies, Security and Privacy in Big Data and Cloud, software Metrics, Data Warehouse – EAI, Data Mining, WBL, and XML. He is a recipient of Best Teacher Award at National Institute of Technology, Tiruchirapalli, India during 2006-2007. He is a member of IEEE, Life Member (LM) of ISTE, New Delhi, Life Member (LM), Computer Society of India.