characteristic is called as the principle of ACAS1. Searchable Encryption techniques hybridization and ABE is presented according to face the property of ACAS. The safe personalized, multi-user access for the outsourced data is presented in the model showing performance of highlighted search. This act with searching some keywords, also is planned for the catalyze search time with using High Performance Computing extensively utilized in the Cloud Computing.

Hierarchical attribute base encryption method implementation based on the Cloudsim tool is described in paper [12]. ABE is performed where the algorithm of rijndael algorithm utilized to encrypt data. Additionally, the article includes decryption scheme utilized for performing, taking advantage of encryption efficiency.

The novel technique namely as match then decrypt is described in paper [13] where the phase of matching is also offered before the phase of decryption. The technique calculates the specific bases in cipher-texts utilized for achieving test whenever the private key of attribute matches hidden access policy in the cipher-texts with no decryption. The specific secret key elements of attribute are produced because of the fast decryption allowing pairings during decryption compression. They present the basic anonymous construction of attribute-based encryption, then receive the safety promulgated deployment based on forcefully existentially unforgeable one-time signatures. The attribute computational cost matching proposed constructions test is less than an operation of decryption that just requires pairings small numbers and constant.

Paper [14] shows the verifiable safe method of CP-ABSC for the PHR sharing system based on cloud which can confidentiality supply, authenticity, fine-grained access control, sign-cryptor privacy, public verifiability at the same time. Demonstrative uniform functions of Boolean are utilized in the framework as the signing, encryption applies and predicates the safety security in a model which is standard. According to the positive consideration, the construction of them looks for requiring the pairing computations less number and the short size of cipher-text in comparison with the existing schemes in record.

B. Cipher-text Policy ABE schemes Improvements

Up to now, in the case of efficiency Cipher-text Policy ABE, the actions is presented. In Reference [15], the novel method is suggested to improve Cipher-text Policy ABE efficiency method. Authors provide the new technique which known as the union of attributes where the attributes specific number are able to be incorporated in one union of attributes. This method core idea is based on the theorem that is arithmetic. Firstly, this will map every attribute in world set of attributes for the single element of prime. Next, we are able to utilize multiply product, show the set of attributes with all prime numbers multiply product relating to every attribute in collection.

Although, reference [15] just addresses problem, develops method efficiency just by the Gate. Although, in the paper, we popularize method to the methods of tree access, containing gates AND, OR that more to be observed in true universe. In approach of tree access, additionally for considering gate OR, moreover to gate AND should consider tree hierarchy, as well as every node share division, and every parent node share generation decoding time with nodes of child. Then propose solution for the cases.

Authors in paper [16] focus on developing the attribute0based encryption efficiency with leveraging the already rejected truth; e.g., the hierarchy relationships which are frequently-found between the attributes natural in a lot of scenarios of access control. The first study attempt of them was inventing HABE notion that is able to be observed as the traditional attribute-based encryption generalization as two descriptions area able to be taken into account to be similar if whole the attributes are not dependent. Further they provide the real construction of HABE addressing the hierarchy of tree between the attributes that is guiding safety. More critically, the construction of them indicates the significant raises in comparison with the traditional attribute-based encryption if the hierarchies of attribute exist.

In article [17], they present the HASBE2 with expanding ASBE3 by the users hierarchical structure for having scalable, fine-grained, flexible access control of data that is outsourced data in the computing of cloud. The method which is proposed by them not only can obtain the scalability through the hierarchical structure of it, but also receive the flexibility, fine-grained access control in the ASBE backing compound attributes.

In paper [18], authors give the attribute-based encryption details, then offer the new cryptosystem which is versatile referred to as the CPHABE4. In the method of CPHABE, the attributes are listed in matrix, so clients with the attributes in higher-level are able to deputize the access rights of themselves to the clients at lower- level. The attributes let the system of CP-HABE for hosting users large number coming from various organizations with deputizing the keys includes enabling effective sharing of data between the hierarchically organized great sets. They construct the scheme of CPHABE with the short cipher-texts. This method has verified to be safe in the model which is standard under the assumptions that are non-interactive. This paper authors build the method of CPHABE utilizing the access structures of LSSS5.

In paper [19], the system focuses on the novel technique for HASBE; this is controlled with CP-ABE to have cloud users hierarchical structure. The method not only achieves the scalability but also this obtains two data fine grained access control in the flexibility, cloud. The action is the information privacy protection usual way to store data in the encrypted form.

[1] Access Control Aware Search

[2] Hierarchical attribute-set-based encryption

[3] Cipher-text-policy attribute-set-based encryption

[4] Cipher-text-policy hierarchical ABE

[5] linear secret sharing scheme

Paper [20] utilize the hierarchical attributes for optimizing the basic scheme decreasing two time of decryption/encryption, also the size of cipher-text when keeping the security of CPA. Finally, they plan the deployment where access policies are arbitrary taken into consideration threshold trees, they reduce by the Cipher-text-Policy ABE practical applications discussion.

III. CIPHER-TEXT-POLICY ABE

Cipher-text-Policy ABE is the tool which is cryptographic and is promising to the future cryptography where owner of data is able to detect the structure of an access for using this; therefore, data which is sensitive will be safeguarded. Now we encrypt the data of cloud utilizing the Cipher-text-Policy ABE based on the attributes/ certificates of data owner; thus, an individual who require the decrypt data, should own an attributes set which is required to the right proper access.

Here are 4 algorithms which the common system of Cipher-text-Policy ABE includes:

- Setup algorithm
- Key-generation
- Encrypt
- Decrypt

These algorithms definition is defined as follow`1:

Setup: but the implied parameter of security, we need no input for the setup. The algorithm is only for initializing system to create the PK (public key), as well as MSK.

Encrypt: an algorithm input include: M, PK, the policy of access A which the M (massage) must to be encrypted including this. Output is the CT which is an encrypted message when policy of access A is mounted in this.

Key-generation: an algorithm inputs include: MSK, S (attribute set). Output is the private-key SK which is in relation to set of attributes.

Decrypt: Inputs include: PK, SK, CT. SK is private key of clients which trying for decrypting the message with this. If client is able to successfully decrypt message (i.e. attributes in SK are able to please the policy of access A) the output of it will be first M.

Later, we will define an ABE system low-level inner working which is based on the elliptic curve cryptography.

By comparing the schemes of KP-ABE, Cipher-Text-Policy ABE, this should to be tell which we described before, In Cipher-Text-Policy ABE users in system are defined with the attributes, so the encrypted message is under attribute-based policies, although in KP-ABE the massage which is encrypted is defined by the attributes when personal keys contain attribute-based policies.

In the sense, Cipher-Text-Policy ABE is more sensory, at least from the view of encryption, than KP-ABE. Owner of data finds attributes which receivers have with defining users who might access content of data, against KP-ABE that this is negligible for specifying who is able for accessing data, so responsibility of access control lies more by an authority who export keys than owner of data [21].

In every items, this should to be considered that the Cipher-Text-Policy ABE is more expensive than KP-ABE computationally, in the cases RAM usage as well as encryption time [21]. In every items, it might be uneven to the users for performing encryption when control of access is transmitted more heavily to authorities. In the test, we try to overcome the Cipher-Text-Policy ABE schemas disadvantage. In following, we test the mathematical background, as well as the Cipher-Text-Policy ABE [22] schema detail.

IV. PROPOSED SCHEMA

The paper stimulates, also shows the effective cloud-based network data access usage based on the encryption method of Cipher-Text-Policy ABE. Present schemes of Cipher-Text-Policy ABE have long decryption keys drawback that key size depends on, also linear to attributes number. The problem is able to avoid using this practically for the limited devices. Now for having the effective, safe access of data, we offer the effective approach of Cipher-Text-Policy ABE, that attempt for decreasing the cipher text length dependency to attributes number. In following, the approach is defined in detail.

A. Proposed effective Cipher-Text-Policy ABE

Basic technical novation for having the more effective construction of Cipher-Text-Policy ABE based on finding the integrating specific attributes number way in one. Let a, b $\in$ Z be two integers, if this is positive, so a is able to divisor b, next we can say that a is the b divisor. The basic theorem in arithmetic says that we are able to say uniquely each integer M > 1 (up to order) by prime divisors product. Utilizing arithmetic theorem, this will be feasible for integrating the attributes number in one. If N = {$attr_1$, ..., $attr_n$} is attributes universal set, after that we will define the way we are able to unionize the attributes set in one where S $\subseteq$ N. Firstly, we select *n* primes randomly {p_1, ..., p_n}, also set exponent e_i = 1; next we calculate the compound like: $M_s = \prod_{i,attri\in S} p_i$ when we are able to utilize p_i to represent attribute $attr_i$, therefore set of attributes is able to be shown as M_S. We are able to decide as bellow for checking whether the attribute $attr_i$ is in set. When MS mod p_i == 0, $attr_i$ $\in$ S; otherwise, $attr_i$ $\notin$ S. whereas MS compound is able to be shown uniquely as:

$M_s = \prod_{i,attri\in S} p_i$, M_S is able to be utilized for representing set accurately.

Let T = $attr_{t1}$ $\wedge$ $attr_{t2}$ $\wedge \cdot \cdot \cdot \wedge attr_{tm}$ where t_i $\in$ [1, n]. For displaying the AND gate T, we ae able to calculate bellow compound: $M_T == \prod_{ti,1\le i\le m} p_{ti}$. Here we have the theorem:

Theorem 1: the structure of access T is able to be pleased with the set of attributes if, only if M_S mod M_T == 0.

In the OR gates case, this is not essential for satisfying whole the OR gate attributes. Satisfying only one of the attributes is adequate, therefore there should be at least the typical number among MT, MS, that we utilize for determining GCD[6] function number existence.

[6] Greatest common divisor

Theorem 2: the set of attribute pleases the OR gate if GCD (M_S, M_T)>1. It means that when values M_S, M_T have the typical divisor, thus M_S pleases M_T that is the OR gate.

- **Setup**

The algorithm of standard setup will first select the bilinear prime set G_0 order p, generator g. Next this choose 2 exponents $\alpha, \beta \in Z_p$ randomly. This will shared PK as [22]:
PK = G_0, g, h = g^{β}, f = $g^{1/\beta}$, e(g, g)$^{\alpha}$, MK = (β, g^{α}).

- **Key generation**

Key generation algorithm input is the M_K, also the attribute set. Firstly, an algorithm chooses one number randomly $r \in Z_p$, after that calculates compound $M_s = \prod_{i,attri \in S} p_i$. Then secret key is taken into consideration as:
$\{ d = g^{\frac{\alpha+1}{\beta}}, dp = g^{\alpha-\beta.r}, dz = g^{r+MS} \}$.

- **Encryption**

Algorithm inputs are PK, M, the structure of access T, firstly the algorithm of encrypt selects an exponent randomly $s \in Z_p$ and compute:

$$\check{c} = M.e(g,g)^{\alpha.s}, c = h^s$$

Firstly, to every x node (includes leaves) includes T, an algorithm choose the polynomial q_x, beginning from root node R. After that, this polynomial q_x degree of set d_x for being one less than value kx of node threshold, to every node x in tree, which is, $d_x = k_x - 1$.
An algorithm begins from root node R, selects the $s \in Zp$ randomly, next sets $q_R(0) = s$ then selects polynomial qR other points in random for completely defining this. To other tree x node, this sets $q_x(0) = q_{parent(x)}(index(x))$, selects the other points d_x in random for completely defining q_x.
Next to every node in gate nodes last level that whole the children are leaves, we choose s_i, then we compute MT_i as:
$M_T == \prod_{ti, 1 \le i \le m} p_{ti}$
After that we calculate values:

$$c_i = e(g,g)^{\alpha.s_i + q_i(0)}, cp_i = g^{s_i}, cz_i = g^{\frac{\beta.s_i}{MT_i}}$$

To every gate in tree last level.
This must be considered that for leaves that are like the last level gate nodes sibling nodes, we do as above, also we compute the 3 values.
Based on our assumption, cipher-text includes access policy T implicitly, which means the user who has set of attributes is able to decrypt M if and only if the set of attributes are able to pleases tree T.

- **Decryption**

Now, as the standard decryption we explain the function of a Decrypt node that works as bellow: For nodes of gate at tree last level, we compute $e(g,g)^{q_i(0)}$ as bellow:
For AND threshold gates:

$$e(g,g)^{qi(0)} = \frac{c_i}{e(cp_i, dp).e(cz_i, (dz^{\%MT})^{MT})} = \frac{e(g,g)^{\alpha.s_i + q_i(0)}}{e(g^{s_i}, g^{\alpha-\beta.r}).e(g^{\frac{\beta.s_i}{MT_i}}, (g^{(r+MS)\%MT})^{MT})}$$

For OR threshold gates:

$$e(g,g)^{qi(0)} = \frac{c_i}{e(cp_i, dp).e(cz_i, (dz^{\%GCD(MS,MT)})^{MT})} = \frac{e(g,g)^{\alpha.s_i + q_i(0)}}{e(g^{s_i}, g^{\alpha-\beta.r}).e(g^{\frac{\beta.s_i}{MT_i}}, (g^{(r+MS)\%GCD(MS,MT)})^{MT})}$$

Proposed scheme basic operation code is given in Appendix 2.
For internal nodes, we do as standard algorithm:
Here, we describe recessive item, if x is the non-leaf node, algorithm Decrypt Node (CT, SK, x) continues as bellow: this is known as Decrypt Node (CT, SK, z) for the whole nodes of children z of x, function output is stored as F_z. If S_x is the child nodes z arbitrary set with size k_x so that $F_z \neq \perp$. If the same set does not exist, next we can say that node is not pleased, therefore function returns $\perp$.
Otherwise, F_x is calculated as follows [22]:

$$F_x = \prod_{z \in S_x} F_z^{\Delta_{i,S'_x}(0)}, where \ {}^{i=index(z)}_{S'_x=\{index(z):z \in S_x\}}$$
$$= \prod_{z \in S_x} (e(g,g)^{r.q_z(0)})^{\Delta_{i,S'_x}(0)}$$
$$= \prod_{z \in S_{x(}} e(g,g)^{r.q_{parent(z)}(index(z))})^{\Delta_{i,S'_x}(0)} \text{ (by construction)}$$
$$= \prod_{z \in S_x} e(g,g)^{r.q_z(i).\Delta_{i,S'_x}(0)}$$

= e (g,g)$^{r.qx(0)}$ (with applying the polynomial interpolation) return conclusion.
We have determined the function of out Decrypt Node, so now we can determine an algorithm of decryption. Firstly, an algorithm begins with calling function for access tree T root node R. When the set please tree, we set A = Decrypt Node (CT,SK,r) = e(g,g)$^{rqR(0)}$ = e(g, g)^{rs}. So now an algorithm is able to decrypt with calculating as bellow:
$\tilde{C}$/ (e(C, D)/A) = $\tilde{C}$/ (e (h^s, g $^{(\alpha+r)/\beta}$)/e (g, g) ^{rs}) = M.

V. SECURITY ANALYSIS

As the last schemes of ABE, the most important problem in designing attribute-based encryption scheme is preventing against attacks that are from colloguing clients. A solution which we propose is to generate PK of user in random, in the way which it is not feasible for combining them; although, in a solution which we propose, we should mount secret dividing in cipher-text instead of PK. The attacker should recover e (g, g)αs clearly to decrypt. For doing it, an attacker should pair component of D from several PK of user with C from cipher-text. In this way, this is able to cause in required value e (g, g)αs however several some value e (g, g)αs are still hidden to him. For hiding the value, only way is if and only if client has the adequate right components of key to satisfy scheme of secret sharing which has mounted in cipher-text. When value of blinding is randomized to randomness from the particular PK of user, the Collusion attacks cannot impact.
Now we focus on reducing cipher-text length for having the effective cipher-text-policy ABE. Therefore, with reducing required elements number in cipher-text, that is dependent to attributes number, to last level nodes number, we obtain our purpose.

VI. EVALUATION

An approach for having the effective Cipher-Text-Policy ABE, analyzing attributes, reduce length of cipher-text. We've taken into consideration both security and efficiency, just authenticated clients can decrypt M. Here are standard libraries that support the elementary that is cryptographic for implementing techniques of ABE [23]. Now to implement an approach which is proposed, we use the toolkit of Cipher-Text-Policy ABE. The toolkit of Cipher-Text-Policy ABE supplies the routines set that are implementing the scheme of CP-ABE. This uses library of PBC for doing the operations of algebraic. A toolkit includes 2 packages which are libbswabe (a which implements operations of core cryptography), as well as cpabe (the library with interface of user, functions at higher level). By Utilizing the toolkit, we will be presented by 4 tools of command line which are used for doing various approach operations.

- **cpabe-setup** – makes PK, MSK
- **cpabe-keygen** – makes PK by the given attributes set
- **cpabe-enc** – encrypts the file for policy, that is the expression in the cases of attributes
- **cpabe-dec** – decrypts the file by using PK

We adjust function of key generation, decryption, encryption for working as we defined in the last section.

A. *Proposed approach efficiency*

For measuring proposed approach efficiency for having the effective Cipher-Text-Policy ABE, we assess this in the parameters,

Length of Cipher-text, time of encryption/decryption under the various random policy trees randomly depth, by various OR, And gates under the various attributes number. Table 1 displays conclusions.

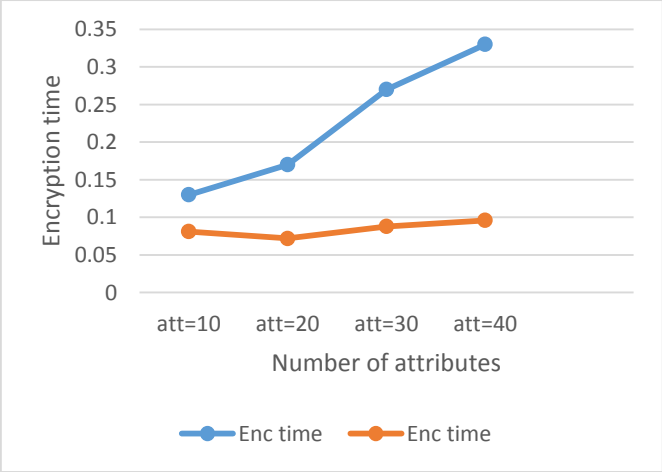

Figure 1. Effect of varying the number of attributes in the access structure on encryption time

As presented in figure.1, by comparing in the time of encryption, the approach achieves better conclusions. In encryption, tree of policy is made, so whole share of nodes is allocated to them. In the proposed approach, we follow tree in less depth due to in nodes with leaves as this children, we allocate the value to this, so this does not require visiting the children of it, therefore this reduce encryption time.

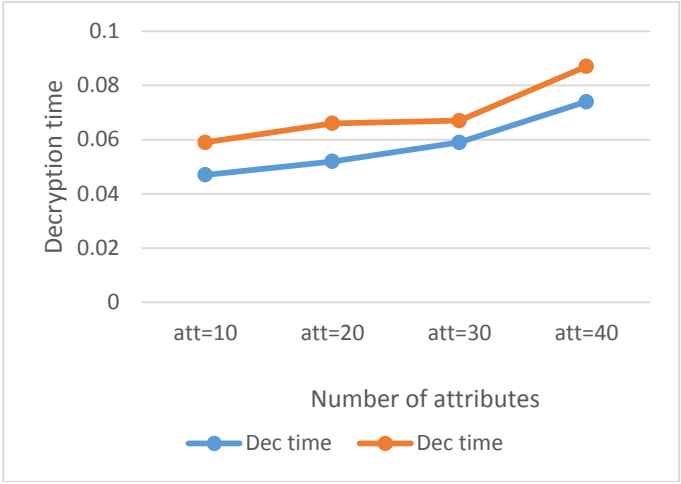

Figure 2. Effect of varying the number of attributes in the access structure on decryption time

As we can see in figure.2, time of decryption, the approach does not have positive impact on the time of decryption, so it causes to raise time of decryption, however this is ignorable due to it does not drastically raise the time of decryption.

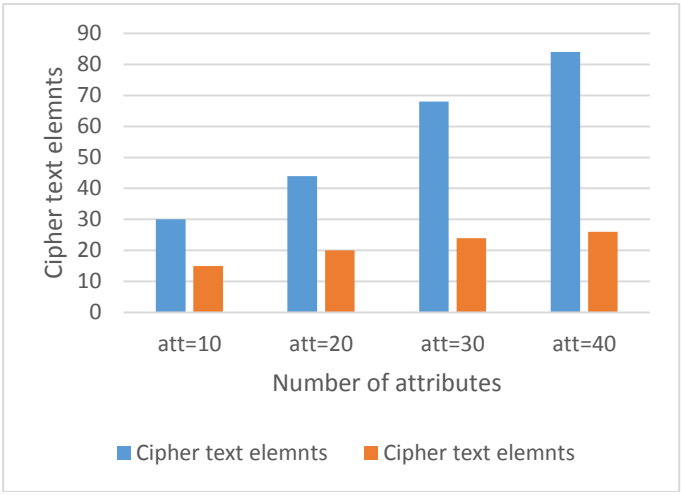

Figure 3. Comparison of cipher text elements

TABLE 1. THE PROPOSED METHOD RESULTS

		att=10	att=20	att=30	att=40
Enc time	**Original**	0.13	0.17	0.27	0.33
	Proposed	0.081	0.072	0.088	0.096
Dec time	**Original**	0.047	0.052	0.059	0.074
	Proposed	0,059	0,066	0.067	0.087
Cipher text elements	**Original**	30	44	68	84
	Proposed	15	20	24	26

ISBN: 978-1-941968-59-8 ©2019 SDIWC

As we can see above, we obtain our purpose for reducing length of cipher-text. As defined in last section, utilizing approaches of union for the OR, AND gates that whole children are leaves, we order elements of cipher-text set. Therefore, as we can see in fig.3, we have less elements of cipher-text in the approach while comparing with original Cipher-Text-Policy ABE.

Also as we can see on data, this stores original attitude. Operations number that are done in decryption are shown in Table 2. As we can see in following table, operation number that the original approach performed to the decryption is more than the proposed approach of us. However, as defined in last approach in procedure of decryption, we should follow all the tree for finding nodes in tree that just have leaves children for decrypting them in various approach while comparing with internal nodes. Thus, following the tree raises decryption time.

TABLE 2. OPERATIONS PERFORMED IN DECRYPTION

			att=10	att=20	att=30	att=40
Decryption operations	**pairings**	**Original**	21	25	27	35
		Proposed	15	9	9	11
	exponentiations	**Original**	10	12	13	17
		Proposed	21	12	12	15
	multiplications	**Original**	136	136	156	221
		Proposed	38	24	25	30

- **The Changes in time of decryption and encryption for the various sizes files**

Additionally, to the above conclusions, for measuring proposed Cipher-Text-Policy ABE scheme efficiency, we examine proposed scheme under the various scenarios by the various files for encrypting in various sizes (10MB-50MB) were encrypted by the random created tree of access under 15, 20 attributes, then the files decrypted by the authenticated client. Conclusions are presented in Tables 3 and 4.

TABLE 3. TEST OF FIRST THE PROPOSED SCHEME WITH T=15

		Size of files				
		10	**20**	**30**	**40**	**50**
Enc time	**Original cpabe**	0.22	0.33	0.45	0.55	0.66
	Proposed cpabe	0.18	0.29	0.39	0.51	0.61
Dec time	**Original cpabe**	0.18	0.32	0.47	0.61	0.76
	Proposed cpabe	0.21	0.34	0.48	0.64	0.78

TABLE 4. TEST OF SECOND THE PROPOSED SCHEME WITH T=20

		Size of files				
		10	**20**	**30**	**40**	**50**
Enc time	**Original cpabe**	0.27	0.37	0.49	0.59	0.70
	Proposed cpabe	0.18	0.29	0.39	0.51	0.62
Dec time	**Original cpabe**	0.18	0.32	0.47	0.61	0.76
	Proposed cpabe	0.18	0.32	0.47	0.63	0.78

As we can see above, due to trees of access which are require to be made; time of Encryption in original Cipher-Text-Policy ABE raises by attributes number in structures of access, in the almost linear procedure. In proposed approach, due to the less dependency to attributes number, the time is less than the main approach.

But, time of decryption wants to be alike due to this is not depend on attribute number, also we are utilizing such key for decrypting an encrypted file.

VII. CONCLUSION

In the article, we use CP-ABE to encrypt the data of cloud based on the credentials/attributes of data owner; thus, an individual who requires to decrypt data should have attributes set which is required for the right access. The article motivates, displays cloud-based network data access usage based on the encryption method of Cipher-Text-Policy ABE

with the primary experiments, analyses for investigating, as well as proving proposed approach efficiency.

As we described before, present Cipher-Text-Policy ABE schemes problem is having the long keys of decryption which size is depend on, linear to attributes number. It is the drawback which avoids practically using limited devices. Now for having the practical and effective access of data, which analyze attributes' dependency and relationship, then remove several extra attributes like leaving attributes are not dependent. In fact, with reducing attributes, we reduce not only length of cipher-text but also time of decryption and encryption.

In the future work, we attempt to do bellow works:

1- Expand approach of union for the other modes of access tree like LSSS/other threshold gates.

2- Expand work to the various kinds of comparative, numerical attributes.

3- Check capability by attributes of negative value.

4-considering proposed schema capability to consider not gate.

REFERENCES

[1] Kumar, N. Saravana, GV Rajya Lakshmi, and B. Balamurugan. "Enhanced attribute based encryption for cloud computing." Procedia Computer Science 46 (2015): 689-696.

[2] Tamizharasi, G. S., B. Balamurugan, and H. Abdul Gaffar. "Privacy preserving ciphertext policy attribute based encryption scheme with efficient and constant ciphertextsize." Inventive Computation Technologies (ICICT), International Conference on. Vol. 3. IEEE, 2016.

[3] Jung, Taeho, et al. "Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption." IEEE Transactions on Information Forensics and Security 10.1 (2015): 190-199.

[4] Zhang, Yinghui, et al. "Computationally efficient ciphertext-policy attribute-based encryption with constant-size ciphertexts." International Conference on Provable Security. Springer International Publishing, 2014.

[5] Chen, Cheng, Zhenfeng Zhang, and Dengguo Feng. "Efficient Ciphertext Policy Attribute-Based Encryption with Constant-Size Ciphertext and Constant Computation-Cost." ProvSec 11 (2011): 84-101.

[6] Rafath, N., Ghouri, W., & Raziuddin, S. "Security in Cloud using Ciphertext Policy Attribute-Based Encryption with Checkability". 3(5). International Journal of Innovative Research in Computer and Communication Engineering. (2015).

[7] Shi, Yanfeng, et al. "Directly revocable key-policy attribute-based encryption with verifiable ciphertext delegation." Information Sciences 295 (2015): 221-231.

[8] Liang, Kaitai, et al. "A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing." Future Generation Computer Systems 52 (2015): 95-108.

[9] Jiang, Yinhao, et al. "Ciphertext-policy attribute-based encryption with key-delegation abuse resistance." Australasian Conference on Information Security and Privacy. Springer International Publishing, 2016.

[10] Zhang, Yinghui, et al. "Efficient attribute-based data sharing in mobile clouds." Pervasive and Mobile Computing 28 (2016): 135-149.

[11] Bouabana-Tebibel, Thouraya, and Abdellah Kaci. "Parallel search over encrypted data under attribute based encryption on the Cloud Computing." Computers & security 54 (2015): 77-91.

[12] Chaudhari, Swapnil H., and B. R. Mandre. "Secure Data Retrieval based on Attribute-based Encryption in Cloud." International Journal of Computer Applications 134.13 (2016).

[13] Zhang, Yinghui, et al. "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing." Information Sciences 379 (2017): 42-61.

[14] Rao, Y. Sreenivasa. "A secure and efficient Ciphertext-Policy Attribute-Based Signcryption for Personal Health Records sharing in cloud computing." Future Generation Computer Systems 67 (2017): 133-151.

[15] Cheng, Yong, Jiangchun Ren, Zhiying Wang, Songzhu Mei, and Jie Zhou. "Attributes union in CP-ABE algorithm for large universe cryptographic access control." In Cloud and Green Computing (CGC), 2012 Second International Conference on, pp. 180-186. IEEE, 2012.

[16] Li, Jin, Qian Wang, Cong Wang, and Kui Ren. "Enhancing attribute-based encryption with attribute hierarchy." Mobile networks and applications 16, no. 5 (2011): 553-561.

[17] Wan, Zhiguo, Jun'E. Liu, and Robert H. Deng. "HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing." IEEE transactions on information forensics and security 7, no. 2 (2012): 743-754.

[18] Denga, Hua, Qianhong Wu, Bo Qinc, Josep Domingo-Ferrerd, Lei Zhange, Jianwei Liub, and Wenchang Shic. "Ciphertext-Policy Hierarchical Attribute-Based Encryption with Short Ciphertexts: Efficiently Sharing Data among Large Organizations."

[19] Gokuldev, S., and S. Leelavathi. "HASBE: a hierarchical attribute-based solution for flexible and scalable access control by separate encryption/decryption in cloud computing." International Journal of Engineering Science and Innovative Technology (IJESIT) 2, no. 3 (2013).

[20] Cheung, Ling, and Calvin Newport. "Provably secure ciphertext policy ABE." In Proceedings of the 14th ACM conference on Computer and communications security, pp. 456-465. ACM, 2007.

[21] Borgh, Joakim. "Attribute-Based Encryption in Systems with Resource Constrained Devices in an Information Centric Networking Context." (2016).

[22] Bethencourt, John, Amit Sahai, and Brent Waters. "Ciphertext-policy attribute-based encryption." Security and Privacy, 2007. SP'07. IEEE Symposium on. IEEE, 2007.

[23] http://acsc.cs.utexas.edu/cpabe/