iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** I **Month of publication:** January 2017

DOI: http://doi.org/10.22214/ijraset.2017.1005

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue I, January 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Achieving Anonymity with Fully Anonymous Attribute-Based Encryption and Controlling Access Privilege in Cloud

Snehanka K. Patil[1], Prof. Ram Joshi[2]
[1]P.G. Student, Department of Computer Engineering, Indira College of Engineering and Management, Pune, Maharashtra, India
[2]Professor, Department of Computer Engineering, Indira College of Engineering and Management, Pune, Maharashtra, India

Abstract: ***Cloud computing could be a revolutionary computing paradigm, that allows versatile, on-demand, and inexpensive usage of computing resources, however the info is outsourced to some cloud servers, and varied privacy considerations emerge from it. Various schemes Supported the attribute-based coding have been projected to secure the cloud storage. However, most work focuses on the info contents privacy and therefore the access management, while less attention is paid to the privilege management and therefore the identity privacy. During this paper, a semi anonymous privilege management theme AnonyControl is proposed to handle not solely the data privacy, however conjointly the user identity privacy in existing access control schemes. Anony Control decentralizes the central authority to limit the identity run and therefore achieves semi anonymity. Besides, it conjointly generalizes the file access management to the privilege control, by that privileges of all operations on the cloud knowledge can be managed in an exceedingly fine-grained manner. Afterwards, the Anony Control-F is presented, that totally prevents the identity leakage and succeed the complete namelessness. The security analysis shows that each Anony Control and Anony Control-F square measure secure under the decisional linear Diffie–Hellman assumption, and new performance analysis exhibits the practicableness of new schemes.***
Index Terms— Anonimity, Attribute-based encryption, , namelessness, multi-authority.

I. INTRODUCTION

CLOUD computing could be a revolutionary computing technique, by that computing resources square measure provided dynamically via net and therefore the information storage and computation are outsourced to somebody or some party during a 'cloud' It greatly attracts attention and interest from each world and business because of the profit, however it conjointly has a minimum of three challenges that has got to be handled before coming back to our real life to the most effective of our data. Initial of all, data confidentiality ought to be warranted. the information privacy isn't only concerning the information contents. Since the foremost engaging a part of the cloud computing is that the computation outsourcing, it is far beyond enough to simply conduct AN access management. Additional probably, users need to manage the privileges of information manipulation over different users or cloud servers. this is often as a result of once sensitive info or computation is outsourced to the cloud servers or another user, that is out of users' management in most cases, privacy risks would rise dramatically as a result of the servers may lawlessly examine users' information and access sensitive information, or different users can be ready to infer sensitive information from the outsourced computation. Therefore, not only the access however conjointly the operation ought to be controlled. Secondly, personal info is in danger as a result of one's identity is attested based on his info for the aim of access management. As folks have become additional concerned concerning their identity privacy lately, the identity privacy conjointly has to be protected before the cloud enters our life. Preferably, any authority or server alone shouldn't grasp any client's personal info. Last however not least, the cloud computing system ought to be resilient within the case of security breach during which some a part of the system is compromised by attackers. Various techniques are planned to safeguard the data contents privacy via access management. Identity-based encryption (IBE) was initial introduced by Shamir [1], in which the sender of a message will specify AN identity such that solely a receiver with matching identity will decipher it. Few years later, Fuzzy Identity-Based encoding [2] is planned, that is additionally referred to as Attribute-Based Encryption (ABE). In such encoding theme, An identity is viewed as a group of descriptive attributes, and secret writing is possible if a decrypter's identity has some overlaps with the one laid out in the cipher text. Soon after, additional general tree-based ABE schemes, Key-Policy Attribute-Based Encryption (KP-ABE) [3] and Ciphertext-Policy Attribute Based encoding (CP-ABE) [4], square measure given to express additional general condition than easy 'overlap'. They are counterparts to every different within the sense that the choice of encoding policy is created by completely different parties.

Volume 5 Issue I, January 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

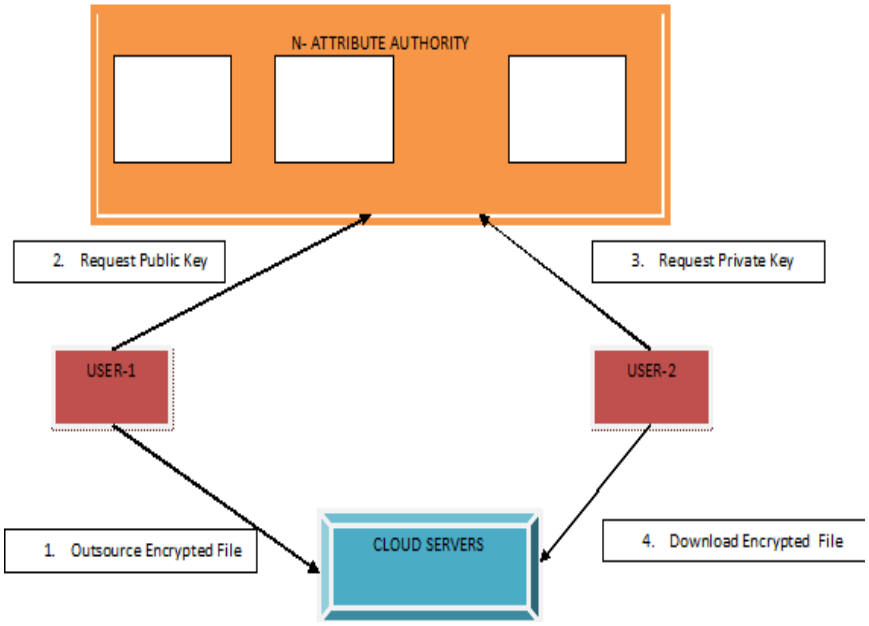


Fig. 1. General flow of scheme

In the KP-ABE [3], a ciphertext is related to a set of attributes, and a personal secret's related to a monotonic access structure sort of a tree, that describes this user's identity (e.g. IIT AND (Ph.D OR Master)). A user can decode the ciphertext if and given that the access tree in his personal secret's happy by the attributes within the ciphertext. However, the cryptography policy is delineated within the keys, so the encrypter doesn't have entire management over the cryptography policy. He needs to trust that the key generators issue keys with correct structures to correct users. what is more, when a re-encryption happens, all of the users within the same system should have their personal keys re-issued therefore on gain access to the re-encrypted files, and this method causes tidy problems in implementation. On the opposite hand, those problems and overhead area unit all solved within the CP-ABE [4]. In the CP-ABE, ciphertexts area unit created with AN access structure, which specifies the cryptography policy, and personal keys area unit generated per users' attributes. A user will decode the ciphertext if and given that his attributes within the personal key satisfy the access tree laid out in the ciphertext. By doing therefore, the encrypter holds the last word authority regarding the cryptography policy. Also, the already issued personal keys can ne'er be modified unless the complete system reboots. Unlike the info confidentiality, less effort is paid to guard users' identity privacy throughout those interactive protocols. Users' identities, that area unit delineated with their attributes, are generally disclosed to key issuers, and also the issuers issue personal keys per their attributes. However it appears natural that users area unit willing to stay their identities secret whereas they till get their personal keys. Therefore, AnonyControl and AnonyControl-F is proposed (Fig. 1) to permit cloud servers to regulate users' access privileges while not knowing their identity info. Their main deserves are: 1) The projected schemes area unit ready to shield user's privacy against every single authority. Partial info is disclosed in AnonyControl and no info is disclosed in AnonyControl-F. 2) The projected schemes area unit tolerant against authority compromise, and compromising of up to (N − 2) authorities doesn't bring the complete system down. 3) offered careful analysis on security and performance to show practicability of the theme AnonyControl and AnonyControl-F

II. RELATED WORK

In [5] and [6], a multi-authority system is given in which every user has AN ID and that they will act with every key generator (authority) exploitation completely different pseudonyms. One user's different pseudonyms square measure tied to his personal key, but key generators ne'er comprehend the personal keys, and so they are not ready to link multiple pseudonyms happiness to the same user. Also, the complete attributes set is split into N disjoint sets and managed by N attributes authorities. In this setting, every authority is aware of solely a region of any user's attributes, that don't seem to be enough to work out the user's identity. However, the theme projected by Chase et al. [6] considered the essential threshold-based KP-ABE, that lacks generality within the encoding policy expression. several attribute based encoding schemes having multiple authorities have been projected later [7]–[10], however they either additionally use a threshold-based ABE [7], or have a semi-honest central authority [8]–[10], or cannot tolerate

haphazardly several users' collusion attack [7]. The work by Lewko et al. [11] and Muller et al. [12] square measure the most similar ones to in this they additionally tried to alter the central authority within the CP-ABE into multiple ones. Lewko et al. use a LSSS matrix as AN access structure, but their scheme solely converts the AND, OR gates to the LSSS matrix, which limits their encoding policy to Boolean formula, while we inherit the flexibleness of the access tree having threshold gates. Muller et al. additionally supports solely mutually exclusive traditional Form (DNF) in their encoding policy. Besides the actual fact that we are able to specific haphazardly general encoding policy, this system additionally tolerates the compromise attack towards attributes authorities, that isn't lined in several existing works. Recently, there additionally appeared traceable multi-authority ABE [13] and [14], that square measure on the other direction of ours. Those schemes introduce responsibility specified malicious users' keys are often derived. On the opposite hand, similar direction as this scheme is often found in [15]–[17], UN agency try and hide encoding policy within the ciphertexts, however their solutions don't stop the attribute revealing within the key generation part.

III. PRELIMINARIES

Let G0 be a increasing cyclic cluster of prime order p and g be its generator. The linear map e ([18], [19]) is outlined as follows: e : G0 × G0 → GT , wherever GT is the codomain of e. The linear map e has the subsequent properties: ∀u,v ∈ G0 and a, b ∈ Zp, e(ua,vb) = e(u,v)ab (bilinearity); for all u,v ∈ G0, e(u,v) = e(v, u) (symmetry); and e(g, g) = one (non-degeneracy). Definition 1: The Decisional linear Diffie-Hellman (DBDH) downside in cluster G0 of prime order p with generator g is outlined as follows: on input g, ga, gb, gc ∈ G0 and e(g, g)z ∈ GT , where a, b, c ∈ Zp, decide whether or not e(g, g)z = e(g, g)abc. The security of the many ABE schemes [4], [20]–[23] and have confidence the belief that no probabilistic polynomial time algorithms will solve the DDH or DBDH downside with non-negligible advantage (DDH assumption and DBDH assumption). This assumption is affordable since separate exponent issues in sizable amount field are wide thought-about to be refractory [24]–[28], and also the teams It is tend to selected are cyclic increasing teams of prime order, within which DBDH problems are believed to be laboriousThe Lagrange constant i,S for i ∈ Zp and a set, S, of components in Zp: i,S(x) := j∈S, j=i x i− −jj , which will be utilized in the polynomial interpolation within the decryption formula. to boot, a unidirectional hash operate H : ∗ → G0 is outlined as a random oracle, which maps any attribute worth to a random component in Zp.

A. *Privilege Trees Tp*

In this work, secret writing policy is delineated with a tree known as access tree. every non-leaf node of the tree could be a threshold element, and each leaf node is delineated by associate degree attribute. One access tree is needed in each record to outline the secret writing policy. In this paper, existing schemes are extended by generalizing the access tree to a privilege tree. The privilege in this scheme is outlined as just like the privileges managed in ordinary operative systems. a knowledge file has many operations executable on itself, and every of them is allowed solely to authorized users with completely different level of qualifications. For example, could be a privileges set of students' grades. Then, reading Alice's grades is allowed to her and her professors, however all alternative privileges should be proved solely to the professors, thus we'd like to grant the "Read_mine" to Alice and every one alternative to the professors. Every operation is related to one privilege p, which is delineated by a privilege tree Tp. If a user's attributes satisfy Tp, he's granted the privilege p. By doing thus, not only the file access management is done however conjointly management alternative workable operations, that makes the file dominant fine-grained and thus appropriate for cloud storage service.

In this theme, many trees are needed in each record to verify users' identity and to grant him a privilege consequently. There are alleged to be r these quite structures, which mean there are completely different privileges outlined for the corresponding record. The privilege zero is unlined because the privilege to scan the file, and alternative privileges could also be outlined every which way (the m-th privilege doesn't essentially have additional powerful privilege than the n-th one once m > n). The tree is analogous to the one outlined in [4]. Given a tree, if numx is that the range of the node x's youngsters node and kx is its threshold price 0 < kx ≤ numx, then node x is assigned a real price if a minimum of kx youngsters nodes are assigned true price. Specially, the node becomes associate degree gate once kx = one associate degreed an gate when kx = numx.

B. *Satisfying the Privilege Tree*

If a user's attributes set S satisfies the privilege tree Tp or the node x, we tend to outline it as Tp(S) = one or x(S) = one severally. Tp (S) is calculated recursively as follows. If x could be a leaf node, x(S) = one if and provided that att(x) ∈ S. If x could be a non-leaf node, x(S) = one only if a minimum of kx kid nodes come back one. For the root node R p of Tp, Tp(S) = one provided that Rp(S) = one.

IV. DOWNSIDE FORMULATION

A. System Model

In this system, there are four varieties of entities: N Attribute Authorities (denoted as A), Cloud Server, information house owners and Data customers. Users are often a knowledge Owner and a knowledge Consumer at the same time. Authorities are assumed to own powerful computation talents, and that they are supervised by government offices as a result of some attributes partly contain users' in person identifiable information. the full attribute set is split into N disjoint sets and controlled by every authority, so every authority is attentive to solely a part of attributes. A Data Owner is that the entity United Nations agency needs to source encrypted record to the Cloud Servers. The Cloud Server, who is assumed to own adequate storage capability, does nothing however store them. Newly joined information customers request non-public keys from all of the authorities, and that they don't apprehend that attributes are controlled by that authorities. once the information customers request their non-public keys from the authorities, authorities jointly produce corresponding non-public key and send it to them. All information customers are able to transfer any of the encrypted information files, however solely those whose non-public keys satisfy the privilege tree Tp will execute the operation related to privilege p. The server is delegated to execute associate degree operation p if and provided that the user's credentials are verified through the privilege tree Tp.

B. Threts Model

It is assumed that the Cloud Servers are semi-honest, who behave properly in most of your time however might conspire with malicious information Consumers or information house owners to reap others' file contents to gain illegitimate profits. however they're conjointly assumed to realize legal benefit once users' requests ar properly processed, which means they'll follow the protocol normally. N authorities ar assumed to be untrusted. That is, they will follow our planned protocol normally, but try to find out the maximum amount data as doable separately. More specifically, we tend to assume they're inquisitive about users' attributes to achieve the identities, however they'll not conspire with users or alternative authorities. This assumption is analogous to several previous researches on security issue in cloud computing (see [20], [29]–[31]), and it's conjointly cheap since these authorities are going to be audited by government offices. Assumption is relaxed and permit the collusion between the authorities.Data shopper's area unit untrusted since they're random users including attackers. They will interact with alternative information shoppers to lawlessly access what they're not allowed to. Besides, don't take into account the identity outpouring from the underlying network since this will be trivially prevented by employing anonymized network protocols (see [32], [33]).

C. Security Model

To formally outline the protection of this AnonyControl, we first provide the subsequent definitions. Setup → PK, MKk: This algorithmic program takes nothing as input except implicit inputs like security parameters. Attributes authorities execute this algorithmic program to collectively cypher a system-wide public parameter PK similarly as associate degree authority-wide public parameter yk, and to separately cypher a master key MKk. Key Generate(PK, MKk, Au) → SKu: This algorithmic program enables a user to move with each attribute authority, and obtains a personal key SKu similar to the input attribute set Au. Encrypt(PK, M, p∈) → (CT, VR): This algorithm takes as input the general public key PK, a message M, and a set of privilege trees p∈, wherever r is set by the encrypter. it'll encipher the message M and returns a ciphertext CT and a verification set VR in order that a user will execute specific operation on the ciphertext if and provided that his attributes satisfy the corresponding privilege tree Tp. As defined, T0 stands for the privilege to browse the file. Decrypt (PK, SKu, CT) → M or verification parameter: This algorithmic program are going to be used at file dominant (e.g. reading, modification, deletion).

It takes as input the general public key PK, a ciphertext CT, and a personal key SKu, that contains a set of attributes Au and corresponds to its holder's GIDu. If the set Au satisfies any tree within the set p∈, the algorithmic program returns a message M or a verification parameter. If the verification parameter is with success verified by Cloud Servers, who use VR to verify it, the operation request are going to be processed. Next, Outline the protection of this AnonyControl with the following game. Init: The someone A declares the set of compromised authorities ⊂ A (where a minimum of 2 authorities in a very are not management led by A) that area unit below his control (remaining authorities A/ area unit controlled by the challenger). Then, he declares T0 that he desires to be challenged, during which some attributes area unit being in charged by the challenger's authorities. Setup∗: The contender and therefore the someone collectively run the Setup algorithmic program to receive the valid outputs. Phase 1: The someone launches Key Generate algorithms to query for as several non-public keys as he desires, that correspond to attribute sets A1,. . ., Aq being disjointly in charged by all authorities , however none of those keys satisfy T0. Besides, he conjointly conducts randomly several computations mistreatment the public and secret keys that he has (belonging to compromised authorities).

Challenge: The someone submits 2 messages M0 and M1 of equal size to the contender. The contender flips a random binary coin b and encrypts Mb with T0. The ciphertext CT is given to the someone. Phase 2: part one is recurrent adaptively, however none of the queried keys satisfy T0. Guess: The someone outputs a guess b of b. The advantage of associate degree someone A during this game is outlined as Pr[b = b] − one two . Definition 2: Our theme is secure and indistinguishable against chosen-attribute attack (IND-CAA) if all probabilistic polynomial-time adversaries (PPTA) have at the most a negligible advantage within the on top of game. Note that the IND-CAA outlined on top of implies IND-CCA since the someone will conduct encryptions and decryptions mistreatment the general public keys and secret keys it owns in Phase one and part two (but he cannot rewrite the target ciphertext since none of its secret keys satisfy T0).

D. Style Goals

Our goal is to realize a multi-authority CP-ABE which: achieves the protection outlined above; guarantees the confidentiality of knowledge Consumers' identity information; and tolerates compromise attacks on the authorities or the collusion attacks by the authorities. For the visual comfort, subsequent notations are regularly used hereafter. American state denotes the k-th attribute authority; Au denotes the attributes set of user u; Au k denotes the set of Au controlled by Ak; and ATP denotes the attributes set included in tree Tp.

V. ANONYCONTROL CONSTRUCTION

A. Set up

At the system formatting part, anyone of the authorities chooses a linear cluster G0 of prime order p with generator g and publishes it. Then, all authorities severally and randomly picks vk ∈ Zp and send Yk = e(g, g)vk to all or any other authorities United Nations agency one by one reason Y := k∈A Yk = e(g, g)k∈A vk .

Then, each authority American state willy-nilly picks N − one integers skj ∈ Zp(j ∈ \) and computes gskj . Each gskj is shared with one another authority A j. Associate in Nursing authority American state, after receiving N − one items of gs jk generated by A j , computes its secret parameter xk ∈ Zp as follows:

$$Xk = (\prod_{j\in\{1,\dots,N\}\{k\}} gskj \ )/(\prod_{j\in\{1,\dots,N\}\{k\}} gskj \ )$$

$$=g(\sum_{j\in\{1,\dots,N\}\{k\}} skj - \sum_{j\in\{1,\dots,N\}\{k\}} skj)$$

It is simple to examine that these haphazardly made integers satisfy k∈A xk = one mod p. this can be a vital property which achieves compromise attack tolerance for our theme, which will be mentioned within the next section. Then, the passkey for the authority American state is MKk = {vk, xk},, and public key of the entire system is printed as PK ={G0, g, Y = e(g, g)vk} Note that the time complexness of the setup computation is O(N2) since each authority computes N − one items of gskj . However, this will be any reduced to O(N) by applying the following straightforward trick. we tend to initial cluster the authorities into C clusters, and exchanges the parameters among the cluster solely. Then, the time complexness is reduced to O(C N) = O(N) since C could be a constant.

B. Keygenerate(PK, MKk, Au)

When a replacement user u with GIDu needs to hitch the system, he requests the non-public key from all of the authorities by following this method that consists of 2 phases.

1) Attribute Key Generation: For any attribute i ∈ Au, every Ak indiscriminately picks Ocean State ∈ Zp to severally cipher the partial private keys H(att(i))ri, Di = gri, that square measure in private sent to the user u. Then, every authority AK indiscriminately picks dk ∈ Zp, computes xk · gvk · gdk and in private shares it with alternative authorities (i.e. unbroken secret to the user u). Then, he in private sends xk ·gdk to the user u (i.e. unbroken secret to alternative authorities). Any one of N authorities computes and sends the subsequent term to the user u: D = πxkgvk gdk = g$\sum$ vk + g$\sum$ dk where gvk acts as a system-wide key wont to generate a valid secret key, however no single authority is in a position to infer its value. a legitimate D with a legitimate gvk are often achieved only if all the authorities properly follow the protocol and conduct a joint computation. Then, the user computes the subsequent term that is that the attribute key for the attribute i (att(i) refers to the part in G0 such as i): Di = H(att(i))ri · π(xk · gdk)

= H(att(i))ri · g(dk) Note that Di is computed firmly while not revealing individual gdk's to the user or revealing gdk to any attribute authority. this can be vital within the tolerance to the compromise attack, which is able to be mentioned later. 2) Key Aggregation: User u, once receiving D, Di 's and D i 's, aggregates the elements as his non-public key: SKu = { D, ∀i ∈ Au : Di = g(dk) · H(att(i))ri, Di = gri }

C. Encrypt (PK, M, {Tp}p∈{0,…,r–1})

The Data Owner encrypts the info with any existing symmetric secret writing theme, and generates the secret writing key Ke. Then, he determines a group of privilege trees p∈ and executes Encrypt(PK, Ke,). Remember that the privilege tree in our theme is based on the brink gates. Here, Shamir's secret sharing technique [34] is directly wont to implement the brink gate. Shamir's t-out of-n secret share theme permits one to divide a secret to n shares, and also the original secret are often recovered with t of them. So, in our tree, the node worth of the gate is recovered if and providing a minimum of kx values of kids nodes are recovered in algorithmic manner. The random range, which is used to mask the secret writing key Ke, is keep at the basis of the privilege tree and is secret-shared to its kids nodes, and the secret shares within the kids nodes square measure secret-shared to their kids nodes, thus so forth till the algorithmic secret sharing reaches the leaf nodes.

This is enforced within the following method. For every Tp, the formula initial chooses a polynomial qx for every node x in it. for every node x, sets the degree dx of the polynomial qx in concert but the brink worth kx. ranging from the root node Rp, the formula indiscriminately picks sp ∈ Zp and sets qR p(0) := sp and indiscriminately chooses alternative coefficients for qR p. Then, for the other node x, the coefficients square measure chosen randomly and also the constant term is ready as qparent(x)(index(x)) such that qx(0) = qparent(x)(index(x)) (index(x) is that the index of the x's kid nodes, and parent(x) is node x's parent node). Finally, he picks a random part h ∈ Zp such h−1 mod p exists, and calculates gh·s p, Dh−1, and also the ciphertext CT is formed as CT = p∈, E0 = Ke · Y s0, C = ghs p, C ˆ = Dh−1 i∈ATp,∀p∈ Note that Dh−1 is introduced to forestall key combination attack, that is comparable to the concept appeared in [4], but in different ways: they introduced such a inverse within the power in key generation formula whereas we tend to will thus within the secret writing in order to attain the de-centralization. Then, VR, that is disclosed solely to the Cloud Server, is created for the aim of privilege verification. VR = {E p = Y s p}p∈{1,...,r-1} Finally, knowledge Owner sends CT, VR and also the encrypted file to the Cloud Server to share them with alternative knowledge shoppers.

D. Decrypt (PK, SKu, CT)

Every user among the system will transfer the ciphertext from the Cloud Server, however he's ready to execute operations on encrypted knowledge solely once he with success decrypts it. Firstly, we outline a algorithmic formula decipher Node(CT, SKu, x), where x stands for a node within the privilege tree Tp. If the node x could be a leaf node, we tend to let i be the attribute of the node x and define as follows. If i ∈ Au,

Decrypt Node

(CT, SKu, x) = e(Di, Cx)/e(Di, Cx)

= e(g $\sum dk$ · H(att(i))ri, gqx(0)) /e(gri, H(att(i))qx(0))

= e(g, g)($\sum dk$)·qx(0)

If not, we tend to outline decipher Node(CT, SKu, x) := ⊥. If x isn't a leaf node, the formula yield as follows: For all nodes z that square measure kids of x, it calls decipher Node(CT, SKu, z) and stores the output as Fz. Let Sx be associate degree discretional kx-sized set of child nodes z such Fz = ∅. If no such set exists then the node wasn't glad and also the formula returns ⊥. Otherwise, compute

Fx = $\prod_{z \in Sxz} Fz\Delta ds'x(0)$ where d =index (z) Sx'= index (z):Z ∈ Sx

$=\prod_{z \in Sx}(e(g,g)\sum(dk).qz(0)\Delta d,s'x(0)$

$=\prod_{z \in Sx}(e(g,g)\sum(dk).qparent(z)(d))\Delta d,s'x(0)$

$=\prod_{z \in Sx}(e(g,g)\sum(dk).qx(d)\Delta d,s'x(0)$

=e(g, g)(dk)·qx(0)

The interpolation higher than recovers the parent node's worth by scheming coefficients of the polynomial and evaluating the p(0). we tend to direct the readers to [34] for complete alculation. A user recursively calls this formula, ranging from the root node Rp of the tree Tp, once downloading the file. If the tree is glad, which implies he's granted the privilege p, then Decrypt Node(CT, SKu,

Rp) = e(g, g)s p$\sum dk$ Finally, if the user is making an attempt to browse the file, the secret writing key Ke are often recovered by:

$$\frac{E_0}{\frac{e(C,\hat{C})}{e(g,g)^{s_0\sum d_k}}} = \frac{K_e \cdot Y^{s_0}}{\frac{e(g,g)^{s_0(\sum d_k+\sum v_k)}}{e(g,g)^{s_0\sum d_k}}} = K_e$$

Then, the info file are often decrypted by exploitation it. Otherwise, if he needs to execute some operation on the info, he should be verified as a licensed user for the execution initial. If the execution needs the j-th privilege, the user recursively calls Decrypt(CT, SKu, x) ranging from the basis node R j of the tree Tj to induce e(g, g)s j dk and any deliver the goods Y s j with the same equation as higher than. The user sends it to the Cloud Server as well because the operation request. The Cloud Server checks whether Y s j = E j , and yield if they are doing equal one another. In fact, Y s j ought to be encrypted to avoid replay attack. This can be merely enforced by introducing any public key encryption protocol.

VI. ACHIEVING ANONYMITY FULLY

The Obscurity A semi-honest authorities is assumed in AnonyControl and also assumed that they're going to not conspire with one another. This is a necessary assumption in AnonyControl as a result of every authority is to blame of a set of the total attributes set, and for the attributes that it's to blame of, it is aware of the precise information of the key requester. If the data from all authorities is gathered altogether, the whole attribute set of the key requester is recovered and so his identity is disclosed to the authorities. during this sense, AnonyControl is semi anonymous since partial identity info (represented as some attributes) is disclosed to every authority, but we can achieve a full-anonymity and additionally enable the collusion of the authorities. The key purpose of the identity info escape we tend to had in our previous theme likewise as each existing attribute based secret writing schemes is that key generator or attribute Algorithm one 1-Out-of-2 Oblivious Transfer

1: Bob indiscriminately picks a secret s and publishes gs to Alice.

2: Alice creates associate degree encryption/decryption key pair:

3: Alice chooses i and calculates E Ki = gr, E Ki−1 = gs/gr and sends E K0 to Bob.

4: Bob calculates E K1 = gs/EK0 and encrypts M0 exploitation E K0 and money supply exploitation E K1 and sends 2 cipher texts EE K0(M0), EE K1(M1) to Alice.

5: Alice will use r to decipher the specified cipher text EE Ki(Mi), however she cannot decipher the opposite one. Meanwhile, Bob doesn't understand that cipher text is decrypted.

Algorithm a pair of 1-Out-of-n Oblivious Transfer

1: Bob indiscriminately picks n secrets s1,. . ., metal and calculates ti as follows:

∀i ∈ : ti = s1 ⊕ · · · ⊕ si−1 ⊕ Mi

2: for every i ∈ , Bob and Alice square measure engaged in a 1-out-of-2 OT wherever Bob's initial message is ti and also the second message is si. Alice picks ti to receive if she needs Mi and si otherwise.

3: once Alice receives n elements, she has ti = s1 ⊕· · ·⊕ si−1 ⊕ Mi for the i she needs and sk for k = i, she can recover the Mi by Mi = ti ⊕ si−1 ⊕ si−2 ⊕ · · · ⊕ s1 authorities in our scheme problems attribute key supported the reported attribute, and also the generator needs to understand the user's attribute to try and do thus. we want to introduce a replacement technique to let key generators issue the right attribute key while not knowing what attributes the users have. A naive answer is to provide all the attribute keys of all the attributes to the key requester and let him choose no matter he needs. During this method, the key generator does not understand that attribute keys the key requester picked, but we've got to totally trust the key requester that he won't pick any attribute key not allowed to him. To unravel this, subsequent Oblivious Transfer (OT) is leveraged.

A. 1-Out-of-n Oblivious Transfer

In associate degree 1-out-of-n OT, the sender Bob has n messages M1,. . ., Mn, and also the receiver Alice needs to select one Mi from those money supply,. . ., Mn. Alice with success achieves Mi without knowing any helpful info regarding alternative messages, and Bob doesn't understand that Mi is picked by Alice.

[35] is used as a building block out of the many implementations [35]–[37], in our totally anonymous multi-authority CP-ABE within the next section.

Then the 1-out-of-2 OT (Algorithm 1) is used , within which Alice picks Mi from Bob's M0, M1, to introduce the 1-out-of-n OT delineated in formula a pair of. In formula a pair of, Alice can do Mi if and providing she picks ti for the i she needs the message and sk for any k = i. If she picks many tk's, some sk's square measure missing and she or he isn't able to recover any message.

B. Totally Anonymous Multi-Authority CP-ABE

In this section, we tend to gift the way to deliver the goods the total obscurity in AnonyControl to styles the totally anonymous privilege control theme AnonyControl-F. The KeyGenerate formula is that the solely half that leaks identity info to every attribute authority. Upon receiving the attribute key request with the attribute worth, the attribute authority can generate H(att(i))ri and sends it to the requester wherever att(i) is that the attribute worth and Ocean State could be a random number for that attribute. The attribute worth is disclosed to the authority during this step. We can introduce the higher than 1-out-of-n OT to forestall this leakage. we tend to let every authority be to blame of all attributes belonging to identical class. for every attribute class c (e.g., University), suppose there square measure k doable attribute values (e.g., IIT, NYU, CMU ...), then one requester has at the most one attribute worth in one class. Upon the key request, the attribute authority will choose a random range atomic number 44 for the requester and generates H(att(i))ru for all i ∈ {1…..k}. After the attribute keys square measure prepared, the attribute authority and the key requester square measure engaged during a 1-out-of-k OT wherever the key requester needs to receive one attribute key among k. By introducing the 1-out-of-k OT in our KeyGenerate algorithm, the key requester achieves the right attribute key that he needs, however the attribute authority doesn't have any useful info regarding what attribute is achieved by the requester. Then, the key requester achieves the total obscurity in our theme and notwithstanding what percentage attribute authorities collude, his identity info is unbroken secret.

VII. DISCUSSION

Trust of Users: Our AnonyControl-F additionally wants to trust the requester that he picks correct attribute keys corresponding to his identity, however the requester will choose solely one attribute key in one class, that is way higher than the naive plan higher than, and it's not this paper's scope to guarantee the truthful news of the attributes. To the simplest of our data, it's assumed that another authentication (e.g., government check) is in situ to verify the rumored attributes in most of ABE-related works. Performance: the additional computation introduced in AnonyControl-F is simply many exponent calculations, which are negligible. However, further communication overhead is a problematic issue in AnonyControl-F. for every attribute category, the user is concerned during a 1-out-of-n OT that wants O(n) rounds of communication. Therefore, the communication overhead grows from O(1) in AnonyControl to O(I) wherever I is the size of the whole attribute set. This can be the most downside of our totally anonymous theme, that ought to be solved in our future work.

VIII. CONCLUSIONS

This paper proposes a semi-anonymous attribute-based privilege management theme AnonyControl and a fully-anonymous attribute-based privilege management theme AnonyControl-F to address the user privacy drawback during a cloud storage server. Using multiple authorities within the cloud ADPS, our planned schemes attain not solely fine-grained privilege control however conjointly identity namelessness whereas conducting privilege control supported users' identity data. additional significantly, our system will tolerate up to N − two authority compromise, that is very desirable particularly in Internet-based cloud computing atmosphere. we have a tendency to conjointly conducted elaborated security and performance analysis that shows that AnonyControl each secure and economical for cloud storage system. The AnonyControl-F directly inherits the protection of the AnonyControl and therefore is equivalently secure because it, but extra communication overhead is incurred throughout the 1-out-of-n oblivious transfer. One of the promising future works is to introduce the economical user revocation mechanism on high of our anonymous ABE. Supporting user revocation is a crucial issue within the real application, and this is often an excellent challenge within the application of ABE schemes. creating our schemes compatible with existing ABE schemes [39]–[40] World Health Organization support economical user revocation is one in all our future works.

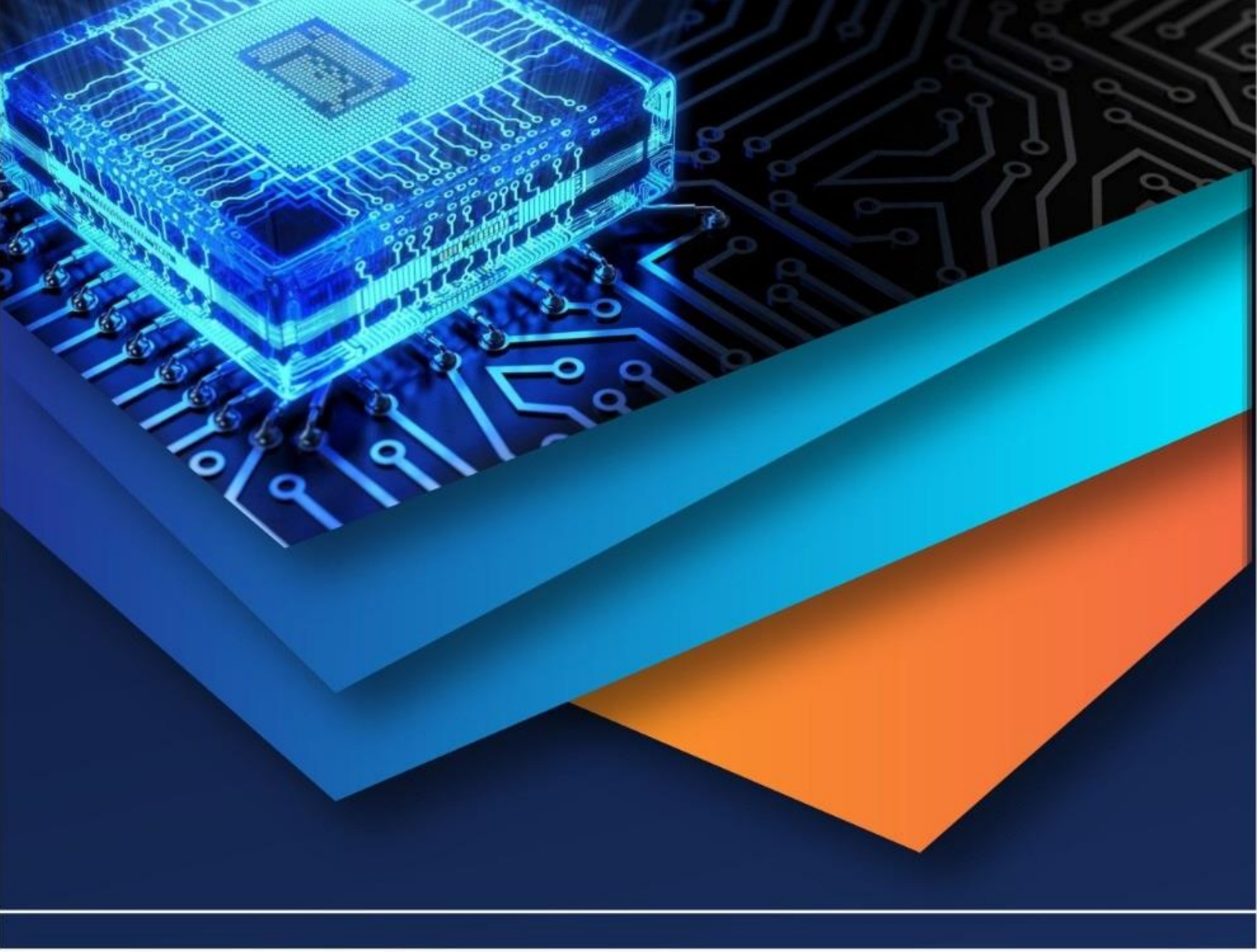
IX. ACKNOWLEDGMENT

The I would like to express my gratitude to Ram Joshi for providing me adequate facilities to complete this paper. I express my gratitude for her support and suggestions regarding dissertation. I also thank Department of Computer Engineering for support and encouragement.

REFERENCES

[1] Shamir, "Identity-based cryptosystems and signature schemes," in Advances in Cryptology.Berlin,Germany:Springer-Verlag,1985,pp.47–53.
[2] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Advances in Cryptology. erlin,Germany:Springer-Verlag,2005,pp.457–473.
[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th CCS, 2006, pp. 89–98.

[4] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attributebased encryption,” in roc. IEEE SP, May 2007, pp. 321–334.
[5] M. Chase, “Multi-authority attribute based encryption,” in Theory of Cryptography. Berlin, Germany: Springer-Verlag, 2007, pp. 515–534.
[6] M. Chase and S. S. M. Chow, “Improving privacy and security in multi-authority attribute based encryption,” in Proc. 16th CCS, 2009, pp. 121–130.
[7] H. Lin, Z. Cao, X. Liang, and J. Shao, “Secure threshold multi authority attribute based encryption without a central authority,” Inf. Sci., vol. 180, no. 13, pp. 2618–2632, 2010.
[8] V. Božovi´ c, D. Socek, R. Steinwandt, and V. I. Villányi, “Multi-authority attribute-based encryption with honest-but-curious central authority,” Int. J. Comput. Math., vol. 89, no. 3, pp. 268–283,2012.
[9] F. Li, Y. Rahulamathavan, M. Rajarajan, and R. C.-W. Phan, “Low complexity multi-authority attribute based encryption scheme for mobile cloud computing,” in Proc. IEEE 7th SOSE,-Mar.2013,pp.573–577.
[10] K. Yang, X. Jia, K. Ren, and B. Zhang, “DAC-MACS: Effective data access control for multi-authority cloud storage systems,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 2895–2903.
IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 10, NO. ,JANUARY-2015 JUNG et al.: CONTROL CLOUD DATA ACCESS PRIVILEGE AND ANONYMITY-199
[11] A. Lewko and B. Waters, “Decentralizing attribute-based encryption,” in Advances in Cryptology.Berlin,Germany:-Springer-Verlag-2011,pp.-568–588.
[12] S. Müller, S. Katzenbeisser, and C. Eckert, “On multi-authority ciphertext-policy attribute-based encryption,” Bull. Korean Math. Soc., vol. 46, no. 4, pp. 803–819, 2009.
[13] J. Li, Q. Huang, X. Chen, S. S. Chow, D. S. Wong, and D. Xie, “Multiauthority ciphertext-policy attribute-based encryption with accountability,” in Proc. 6th ASIACCS, 2011, pp. 386–390.
[14] H. Ma, G. Zeng, Z. Wang, and J. Xu, “Fully secure multi-authority attribute-based traito tracing,” J. Comput. Inf. Syst., vol. 9, no. 7, pp. 2793–2800, 2013.
[15] S. Hohenberger and B. Waters, “Attribute-based encryption with fast decryption,” in Public-Key Cryptography. Berlin, Germany: Springer-Verlag, 2013, pp. 162–179.
[16] J. Hur, “Attribute-based secure data sharing with hidden policies in smart grid,” IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 11, pp. 2171–2180, Nov. 2013. [17]
[17] Y. Zhang, X. Chen, J. Li, D. S. Wong, and H. Li, “Anonymous attributebased encryption supporting efficient decryption test,” in Proc. 8th ASIACCS, 2013, pp. 511–516.
[18] D. Boneh and M. Franklin, “Identity-based encryption from the weil pairing,” in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2001, pp. 213–229.
[19] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2005.
[20] Liu, Z. Wan, and M. Gu, “Hierarchical attribute-set based encryption
for scalable, flexible and fine-grained access control in cloud computing,” in Information Security Practice and Experience. Berlin, Germany: Springer-Verlag, 2011, pp. 98–107.
[21] A. Kapadia, P. P. Tsang, and S. W. Smith, “Attribute-based publishing with hidden credentials and hidden policies,” in Proc. NDSS, 2007, pp. 179–192.
[22] S. Yu, K. Ren, and W. Lou, “Attribute-based content distribution with hidden policy,” in Proc. 4th Workshop Secure Netw. Protocols, Oct. 2008, pp. 39–44.
[23] Z. Wan, J. Liu, and R. H. Deng, “HASBE: A hierarchical attributebased solution for flexible and scalable access control in cloud computing,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 743–754, Apr. 2012.
[24] T. Jung, X. Mao, X.-Y. Li, S.-J. Tang, W. Gong, and L. Zhang, “Privacypreserving data aggregation without secure channel: Multivariate polynomial evaluation,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 2634–2642.
[25] T. Jung and X.-Y. Li, “Collusion-tolerable privacy-preserving sum and product calculation without secure channel,” IEEE Trans. Dependable Secure Comput., to be published.
[26] X.-Y. Li and T. Jung, “Search me if you can: Privacy-preserving location query service,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 2760–2768.
[27] L. Zhang, X.-Y. Li, Y. Liu, and T. Jung, “Verifiable private multiparty computation: Ranging and ranking,” in Proc. IEEE INFOCOM, Apr. 2013, pp. 605–609.
[28] L. Zhang, X.-Y. Li, and Y. Liu, “Message in a sealed bottle: Privacy preserving friending in social networks,” in Proc. IEEE 33rd ICDCS, Jul. 2013, pp. 327–336.
[29] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-preserving public auditing for data storage security in cloud computing,” in Proc. IEEE INFOCOM, Mar. 2010, pp. 1–9.
[30] C. Wang, K. Ren, and J. Wang, “Secure and practical outsourcing of linear programming in cloud computing,” in Proc. IEEE INFOCOM, Apr. 2011, pp. 820–828.
[31] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, “Secure ranked keyword
search over encrypted cloud data,” in Proc. IEEE 30th ICDCS, Jun. 2010, pp. 253–262.
[32] Y. Liu, J. Han, and J. Wang, “Rumor riding: Anonymizing unstructured peer-to-peer systems,” IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 3, pp. 464–475, Mar. 2011.
[33] Tor: Anonymized Network. [Online]. Available: https://www.torproject.org/, accessed 2014.
[34] A. Shamir, “How to share a secret,” Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.
[35] M. Naor and B. Pinkas, “Oblivious transfer and polynomial evaluation,” in Proc. 31st STOC, 1999, pp. 245–254.
[36] S. Even, O. Goldreich, and A. Lempel, “A randomized protocol for signing contracts,” Commun. ACM, vol. 28, no. 6, pp. 637–647, 1985.
[37] W.-G. Tzeng, “Efficient 1-out-of-n oblivious transfer schemes with universally usable parameters,” IEEE Trans. Comput., vol. 53, no. 2, pp. 232–240, Feb. 2004.
[38] Ciphertext-Policy Attribute-Based Encryption Toolkit. [Online]. Available: http://acsc.csl.sri.com/cpabe/, accessed 2014.
[39] W. Ren, K. Ren, W. Lou, and Y. Zhang, “Efficient user revocation for privacy-aware PKI,” in Proc. ICST, 2008, Art. ID 11.
[40] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, “Scalable and secure sharing of personal health records in cloud computing using attributebased encryption,” IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 1, pp. 131–143, Jan. 2013.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY