iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** VIII **Month of publication:** August 2015

DOI:

www.ijraset.com

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

Volume 3 Issue VIII, August 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Secure Cloud Storage Using Decentralized Access Control with Unspecified Authentication

M.Sujana[1], M.Swati[2], T.Sujilatha[3]
CSE Dept, GKCE Sullurpeta, Nellore, A.P, India

Abstract-The Secure Cloud Storage Using Decentralized Access Control with Unspecified Authentication for secures the data. The cloud verifies the authenticity of the series without knowing the user's identity before storing data. Our scheme also has the added feature of access control in which only valid users are able to decrypt the stored information. The scheme prevents replay attacks and supports creation, modification, and reading and writing data stored in the cloud. Our authentication and access control scheme is decentralized and robust, unlike other access control schemes designed for clouds which are centralized. The communication, computation, and storage overheads are comparable to centralized approaches.in this paper, we like to hide the attributes and access policy of a user.

Keywords-AC - Access Control, CS-Cloud Storage, Authentication, ABS-Attribute-Based Signatures, ABE-Attribute-Based Encryption.

I. INTRODUCTION

Cloud is a market-oriented distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers. In cloud computing, users can outsource their computation and storage to servers using Internet. Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (e.g., Amazon's EC2, Eucalyptus, Nimbus), mazon'sandplatforS3Windows Azure).Security is needed because data stored in clouds is highly sensitive, for example, medical records and social networks. User privacy is also required so that the cloud or other users do not know the identity of the user. Thus it is a complex system which possesses highly securable processes. So it must need a proper systematic scheme to manage data.

A system which is based on attribute based encryption for Fine-Grained Access Control of Encrypted Data. To keep sensitive user data confidential against unauthenticated servers, existing schemes usually apply cryptographic methods by disclosing data decryption keys only to authorized users. We combine techniques of attribute-based encryption (ABE) and several other techniques. The problem in this latest technique is that Single data owner will be easily be overwhelmed by the key management overhead. So apart from security concerns we have to concentrate on the key distribution also. Security and privacy protection in clouds are being explored by many researchers, Addressed storage security using Reed-Solomon erasure-correcting codes.

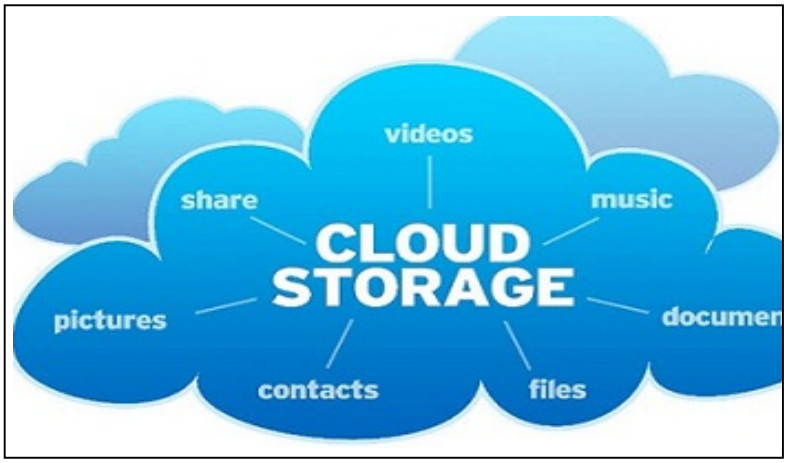


Fig.1 Cloud Storage

Authentication of users using public key crypto-graphic techniques has been studied in homomorphic encryption techniques have been suggested to ensure that the cloud is not able to read the data while performing computations on them. Using homomorphism encryption, the cloud receives cipher text of the data and performs computations on the cipher text and returns the encoded value of the result. The user is able to decode the result, but the cloud does not know what data it has operated on. In such circumstances, it must be possible for the user to verify that the cloud returns correct results. It is just not enough to store the contents securely in the cloud but it might also be necessary to ensure anonymity of the user. For example, a user would like to store some sensitive information but does not want to be recognized. The user might want to post a comment on an article, but does not want identity to be disclosed. However, the user should be able to prove to the other users that are a valid user who stored the information without revealing the identity. There are cryptographic protocols like ring signatures mesh signatures, group signatures which can be used in these situations. Ring signature is not a feasible option for clouds where there are a large number of users.

Group signatures assume the pre-existence of a group which might not be possible in clouds. Mesh signatures do not ensure if the message is from a single user or many users colluding together. For these reasons, a new protocol known as attribute-based signature (ABS) has been applied. In ABS, users have a claim predicate associated with a message. The claim predicate helps to identify the user as an authorized one, without revealing its identity. Other users or the cloud can verify the user and the validity of the message stored. ABS can be combined with ABE to achieve authenticated access control without disclosing the identity of the user to the cloud. Access controls in cloud are centralized in nature. All other schemes use ABE. The scheme in uses a symmetric key approach and does not support authentication. The schemes do not support authentication as well. However, the authors take a centralized approach where a single key distribution center (KDC) distributes secret keys and attributes to all users. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment.

We, therefore, emphasize that clouds should take a decentralized approach while distributing secret keys and attributes to users. It is also quite natural for clouds to have many KDCs in different locations in the world. A decentralized approach, their technique does not authenticate users, who want to remain Unspecified while accessing the cloud. A distributed access control mechanism in clouds. However, the scheme did not provide user authentication. Write access was not permitted to users other than the creator, our previous work with added features that enables to authenticate the validity of the message without revealing the identity of the user who has stored information in the cloud. In this version we also address user revocation, this is an important property because a user, revoked of its attributes, might no longer be able to write to the cloud.

The main contributions of this paper are the following:

The Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them.

Authentication of users who store and modify their data on the cloud.

The identity of the user is protected from the cloud during authentication.

The architecture is decentralized, meaning that there can be several KDCs for key management.

Revoked users cannot access data after they have been revoked.

The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been revoked cannot write back stale information.

The protocol supports multiple read and writes on the data stored in the cloud.

II. RELATED WORK

The ABE, a user has a set of attributes in addition to its unique ID. There are two classes of ABEs. In key-policy ABE or KP-ABE the sender has an access policy to encrypt data. A writer whose attributes and keys have been revoked cannot write back stale information. The receiver receives attributes and secret keys from the attribute authority and is able to decrypt information if it has matching attributes. In Cipher text-policy, CP-ABE the receiver has the access policy in the form of a tree, with attributes as leaves and monotonic access structure with AND, OR and other threshold gates. All the approaches take a centralized approach and allow only one KDC, which is a single point of failure. A multi-authority ABE, in which there are several KDC authorities (coordinated by a trusted authority) which distribute attributes and secret keys to users. Multi-authority ABE protocol was studied which required no trusted authority which requires every user to have attributes from at all the KDCs. Recently proposed a fully decentralized ABE where users could have zero or more attributes from each authority and did not require a trusted server. In all these cases, decryption at user's end is computation intensive. So, this technique might be inefficient when users access using their mobile devices.

However, the presence of one proxy and one KDC makes it less robust than decentralized approaches.

III. BACKGROUND

A. Assumptions

Users can have either read or write or both accesses to a file stored in the cloud.
All communications between users/clouds are secured by the secure shell protocol technique, SSH.

B. Formats of Access Policies

Boolean functions of attributes,
Linear secret sharing scheme (LSSS) matrix of the data [1], or
Monotone span programs.

TABLE 1: NOTATIONS

Symbols	Meanings
U_u	u-th User/Owner
$\mathcal{A}_j$	j-th KDC
$\mathbb{A}$	Set of KDCs
L_j	Set of attributes that KDC $\mathcal{A}_j$ possesses
$l_j = \|L_j\|$	Number of attributes that KDC $\mathcal{A}_j$ possesses
$I[j,u]$	Set of attributes that $\mathcal{A}_j$ gives to user U_u for encryption/decryption
I_u	Set of attributes that user U_u possesses
$J[j,u]$	Set of attributes that $\mathcal{A}_j$ gives to user U_u for claim attributes
J_u	Set of attributes that user U_u possesses as claim attributes
$AT[j]$	KDC which has attribute j
$PK[j]/SK[j]$	Public key/secret key of KDC $\mathcal{A}_j$ for encryption/decryption
$sk_{i,u}$	Secret key given by $\mathcal{A}_j$ corresponding to attribute i given to user U_u
TPK/PSK	Trustee public key/secret key
$APK[j]/ASK[j]$	Public key/secret key of KDC $\mathcal{A}_j$ for verifying claim
$\mathcal{X}$	Boolean access structure
$\mathcal{Y}$	Claim policy
τ	Time instant
R	Access matrix of dimension $m \times h$
M	Matrix of dimension $l \times t$ corresponding to the claim predicate
MSG	Message
$\|MSG\|$	Size of message MSG
C	Ciphertext
$H, \mathcal{H}$	Hash functions, example SHA-1

C. Mathematical Background

1) Properties

a) $e(aP,bQ) = e(P,Q)^{ab}$ for all $P,Q \in G$ and $a,b \in Z_q$,
$Z_q = \{0, 1, 2, \ldots, q-1\}$.
b) Nondegenerate: $e(g, g) \neq$

D. Attribute-Based Signature Scheme

1) System Initialization: Select a prime q, and groups G1 and G2, which are of order q. We define the mapping ˆe : G1 ×G1 → G2. Let g1, g2 be generators of G1 and hj be generators of G2, for j ∈ [tmax], for arbitrary tmax. Let H be a hash function. Let A0 =

ha0 0, where a0 ∈ Z∗ q is chosen at random. (TSig,TV er) mean TSig is the private key with which a message is signed and TV er is the public key used for verification. The secret key for the trustee is TSK = (a0, TSig) and public key is TPK = (G1, G2, H, g1, A0, h0, h1. . . htmax, g2, TV).

2) *User Registration:* For a user with identity Uu the KDC draws at random Kbase ∈ G. Let K0 = K1/a0 base. The following token γ is output γ = (u, Kbase, K0, ρ), where ρ is signature on u||Kbase using the signing key TSig.

3) *KDC Setup:* We emphasize that clouds should take a decentralized approach while distributing secret keys and attributes to users. It is also quite natural for clouds to have many KDCs in different locations in the world. The architecture is decentralized, meaning that there can be several KDCs for key management. Attribute generation.

The token verification algorithm verifies the signature contained in γ using the signature verification key TV er in TPK. This algorithm extracts Kbase from γ using (a, b) from ASK[i] and computes Kx = K1/(a+bx) base , x ∈ J[i, u]. The key Kx can be checked for consistency using algorithm ABS.KeyCheck(TPK,APK[i], γ,Kx), which checks ˆe(Kx,AijBx ij) = ˆe(Kbase, hj), for all x ∈ J[i, u] and j ∈ [tmax].

4) *Attribute Generation:*

a) System Initialization: Select a prime q and groups G_1, G2 .Define the mapping a G * G →G .Let → hash function=hand. (TSig, TVer) mean TSig is the private key with which a message is signed and TVer is the public key used for verification.

b) User Registration: For a user with identity Uu the KDC draws at random K_{base} ∈G. Let $K_0 = K_{base}^{1/a0}$. Token is generated.

5) *Sign:* The access policy decides who can access the data stored in the cloud. The creator decides on a claim policy Y, to prove her authenticity and signs the message under this claim. The ciphertext C with signature is c, and is sent to the cloud. The cloud verifies the signature and stores the ciphertext C. When a reader wants to read, the cloud sends C. If the user has attributes matching with access policy, it can decrypt and get back original message.

6) *Verify:* The verification process to the cloud, it relieves the individual users from time consuming verifications. When a reader wants to read some data stored in the cloud, it tries to decrypt it using the secret keys it receives from the KDCs.

IV. PROPOSED METHOD

We propose our privacy preserving authenticated access control scheme. According to our scheme a user can create a file and store it securely in the cloud. This scheme consists of use of the two protocols ABE and ABS; we will first discuss our scheme in details and then provide a concrete example to demonstrate how it works. There are three users, a creator, a reader, and writer. Creator Alice receives a token-from the trustee, who is assumed to be honest. A trustee can be someone like the federal government who manages social insurance numbers etc. On presenting the id (like health/social insurance number), which can be scattered. For example, these can be servers in different parts of the world. A creator on presenting the token to one or more KDCs receives keys for encryption/decryption and signing. The access policy decides who can access the data stored in the cloud. The creator decides on a claim policy Y, to prove her authenticity and signs the message under this claim. The cipher text C with signature is c, and is sent to the cloud. The cloud verifies the signature and stores in the cloud.

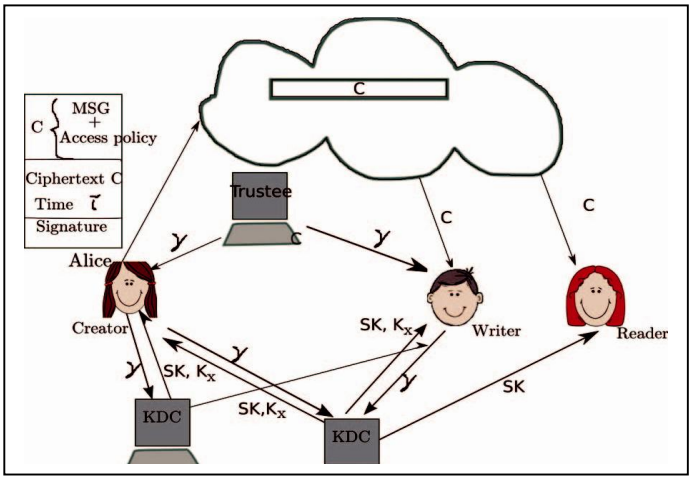


Fig. 2 Cloud Secure Storage Model.

A. Data Storage In Clouds

Users Uu have one or more trustees. This is used to prevent to the replay attacks. In this time data is not sent, then the user can write previous stale message back to the cloud with a valuable signature, even when its claim policy and attributes have been revoked.

B. Reading from the Cloud

The user requests data from the cloud, the cloud sends the cipher text using SSH protocol. Decryption proceeds using algorithm ABE.

C. Writing To The Cloud

The user must send its message with the claim policy as done during file creation. The cloud verifies the claim policy, and only if the user is authentic is allowed to write on the file.

D. User Revocation

It should be ensured that users must not have the ability to access data, even if they possess matching set of attributes.

V. SECURITY OF THE PROTOCOL

We will explain that our scheme authenticates a user who wants to write to the cloud. A user should only write provided the cloud is able to validate it access to the claim. An invalid user cannot receive the attributes from a KDC, if it does not have the credentials are revoked, then it cannot replace data with previous data, thus preventing replay attacks.

Theorem 1: Our access control scheme is secure, collusion resistant and allows access only to authorized users.

Theorem 2: Our authentication data is correct, collusion secure, resistant to the replay of attacks, and protects privacy of the user.

Next we confirm that only a valid user with valid access claim is only able to store the message in the cloud. A user, who wants to create a file and tries to make a wrong access claim, cannot do so, since it will not have attribute keys Kx from the related KDCs. Since the message is encrypted, a user without valid access policy cannot decrypt and change the information.

VI. COMPUTATION COMPLEXITY

To calculate the computations required by users (creator, reader, writer) and that is provided by the cloud. The following Table 2 presents notations used for different operations.

TABLE 2: NOTATIONS

Symbols	Computation
E_{π}	Exponentiation in group G_{π}
τ_H	Time to hash using function H
$\tau_{\mathcal{H}}$	Time to hash using function $\mathcal{H}$
$\tau_{\hat{e}}/\tau_{\hat{e}}$	Time taken to perform 1 pairing operation in $e/\hat{e}$
$\|G\|$	Size of group G
a	Number of KDCs which contribute keys to user

VII. COMPARISON WITH OTHER ACCESS

A. Data Control Schemes In Cloud

The compare our proposed scheme with other control schemes. The comparison is shown in the following table-3:

Volume 3 Issue VIII, August 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

TABLE 3: Comparison of Our Scheme with Existing Access Control Schemes

Schemes	Fine-grained access control	Centralized/ Decentralized	Write/read access	Type of access control	Privacy preserving authentication	User revocation?
[38]	Yes	Centralized	1-W-M-R	Symmetric key cryptography	No authentication	No
[12]	Yes	Centralized	1-W-M-R	ABE	No authentication	No
[13]	Yes	Centralized	1-W-M-R	ABE	No authentication	No
[16]	Yes	Decentralized	1-W-M-R	ABE	No authentication	Yes
[33]	Yes	Centralized	1-W-M-R	ABE	No authentication	No
[34]	Yes	Decentralized	1-W-M-R	ABE	Not privacy preserving	Yes
[15]	Yes	Centralized	M-W-M-R	ABE	Authentication	No
Ours	Yes	Decentralized	M-W-M-R	ABE	Authentication	Yes

TABLE 4
Comparison of Computation during Read and Write by User and Cloud

Schemes	Computation by user while write	Computation by user while read	Computation by cloud while write
[12]	No write access	$m\tau_{P'}$	No write access
[13]	No write access	$m\tau_{P}$	No write access
[16]	No write access	$2m\tau_{P'} + \tau_H + O(mh)$	No write access
[33]	No write access	$E_0 + \tau_H + O(mh)$	No write access
[34]	No write access	$E_0 + \tau_H + O(mh)$	No write access
[15]	$E_1 + (2m+1)E_3 + m\tau_H$ (encrypt) $(2l+2)E_1 + 2tE_2 + \tau_{\mathcal{H}}$ (sign)	$(2m+1)\tau_{P'}$ (decrypt)	$(l+2l)\tau_{\hat{P}} + l(E_1 + E_2) + \tau_{\mathcal{H}}$ (verify)
Our approach	$(3m+1)E_0 + 2mE_T + \tau_P$ (encrypt) $(2l+2)E_1 + 2tE_2 + \tau_{\mathcal{H}}$ (sign)	$2m\tau_P + \tau_H + O(mh)$ (decrypt)	$(l+2t)\tau_{\hat{P}} + l(E_1 + E_2) + \tau_{\mathcal{H}}$ (verify)

1-W-M-R means that only one user can write while many and read. We can see that most schemes do not support others are centralized. Our scheme supports to the privacy preserving authentication of user, but the other schemes are not supported.

VIII. CONCLUSION

The conclusion of the paper is to present a decentralized access control technique with unspecified authentication. Its provides user revocation and prevents to the replay attacks. The cloud does not know identity of the users who store the information but one and only verifies the user's credentials.

IX. ACKNOWLEDGEMENT

I like to thank our HOD, PRINCIPAL and OTHER FACULTIES for their valuable comments and helpful suggestions.

REFERENCES

[1] Cloud Storage," Proc. 14th Int'l Conf. Financial Cryptography and Data Security, pp. 136-149, 2010.
[2] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-Based Authentica-tion for Cloud
[3] Computing," Proc. First Int'l Conf. Cloud Computing (CloudCom), pp. 157-166, 2009.
[4] C. Gentry, "A Fully Homomorphic Encryption Scheme," PhD dissertation, Stanford Univ., http://www.crypto.stanford.edu/ craig, 2009.
[5] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-Based Cloud Computing," Proc. Third Int'l Conf. Trust and Trustworthy Computing (TRUST), pp. 417-429, 2010.
[6] R.K.L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B.S. Lee, "Trustcloud: A Framework for Accountability and Trust in Cloud Computing," HP Technical Report HPL-2011-38, http://www.hpl.hp.com/techreports/ 2011/HPL-2011-38.html, 2013.
[9] D.F. Ferraiolo and D.R. Kuhn, "Role-Based Access Controls," Proc. 15th Nat'l Computer Security Conf., 1992.
[10] D.R. Kuhn, E.J. Coyne, and T.R. Weil, "Adding Attributes to Role-Based Access Control," IEEE Computer, vol. 43, no. 6, pp. 79-81, June 2010.
[11] M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-Owner Settings," Proc. Sixth Int'l ICST Conf. Security and Privacy in Comm. Networks (SecureComm), pp. 89-106, 2010.
[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based Data Sharing with Attribute Revocation," Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS), pp. 261-270, 2010.

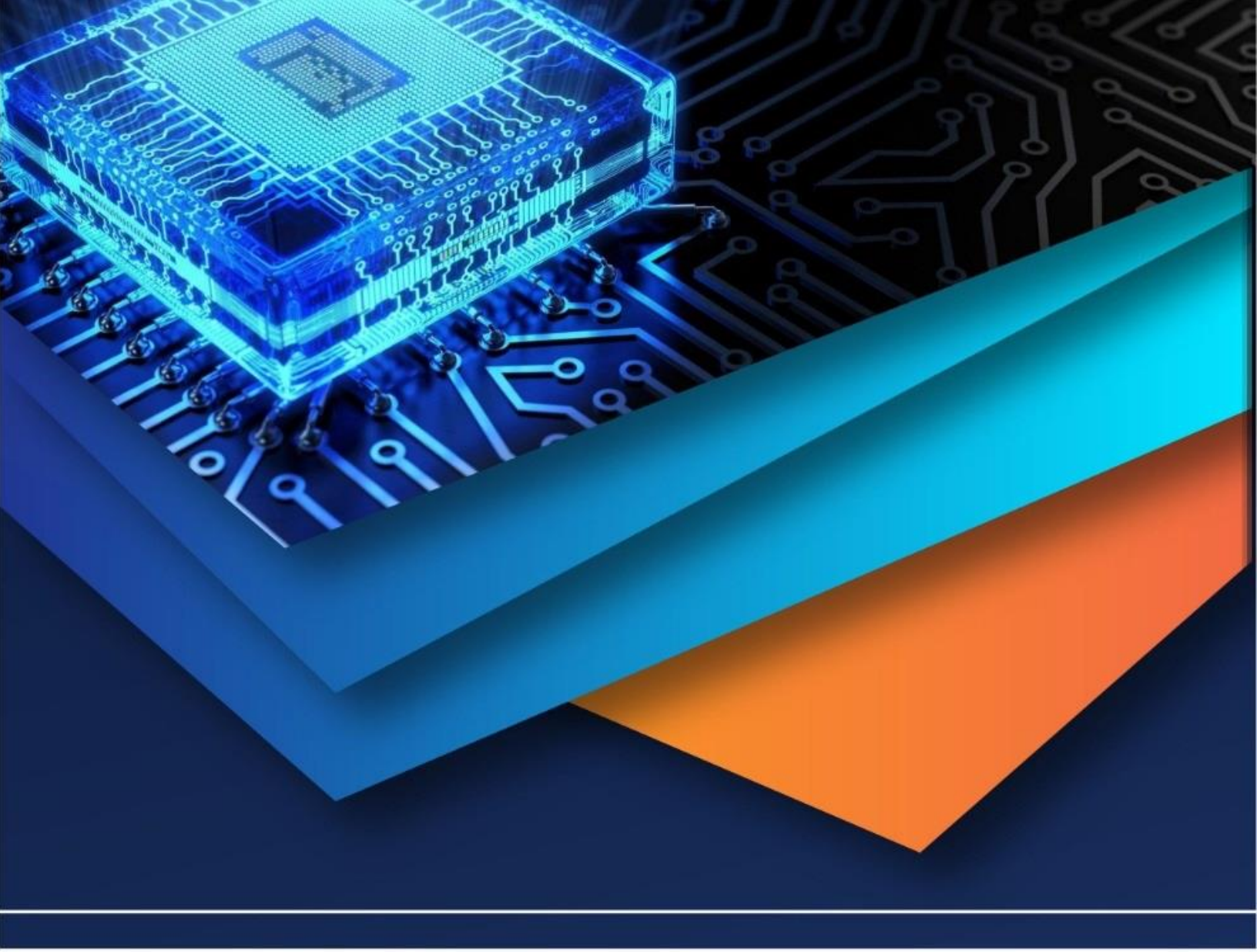
[13] G. Wang, Q. Liu, and J. Wu, "Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services," Proc. 17th ACM Conf. Computer and Comm. Security (CCS), pp. 735-737, 2010.
[14] F. Zhao, T. Nishide, and K. Sakurai, "Realizing Fine-Grained and Flexible Access Control to Outsourced Data with Attribute-Based Cryptosystems," Proc. Seventh Int'l Conf. Information Security Practice and Experience (ISPEC), pp. 83-97, 2011.
[15] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed Access Control in Clouds," Proc. IEEE 10th Int'l Conf. Trust, Security and Privacy in Computing and Communications (TrustCom), 2011.
[16] http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cs-01-en.pdf, 2013.
[17] http://securesoftwaredev.com/2012/08/20/xacml-in-the-cloud, 2013.

AUTHORS

Mrs.M.SUJANA, Pursuing my M.Tech (CSE) in GOKULKRISHNACOLLEGE OF ENGINEERING, Sullurpeta, and my area of interest is networking and cloud computing, E-mail id:kati.kalau@gmail.com

Mrs. M.SWATI., ASST PROFESSOR in the Department of CSE at GOKULA KRISHNA COLLEGE OF ENGINEERING, Sullurpeta, and my area of interest is Cloud computing, OS,&SPM etc, E-mail id:swatim.ghate@gmail.com.

MissT.SUJILATHA.ASST PROFESSOR in the Department of CSE at GOKULA KRISHNA COLLEGE OF ENGINEERING, Sullurpeta, and my area of interest is networking, Cloud computing, WSN and DMDW etc, Email id: illu.suji@gmail.com.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY