IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 12, December 2015.
www.ijiset.com
ISSN 2348 – 7968

Comprehensive Secure Data Sharing in Cloud Storage

Golla Suresh[1] , A Narayanarao[2]

[1] Dept. of CSE, J.N.T.U Anantapur, Tirupathi, Andhra Pradesh, India, Email- resh100.su@gmail.com

[2] Dept. of CSE, J.N.T.U Anantapur, Tirupathi, Andhra Pradesh, India

Abstract

Data sharing is a crucial practicality in cloud storage. During this article, we have a tendency to show a way to firmly, expeditiously, and flexibly share information with others in cloud storage. We have a tendency to describe new public-key cryptosystems that manufacture constant-size cipher texts specified economical delegation of decipherment rights for any set of cipher texts square measure potential. The novelty is that one will combination any set of secret keys and build them as compact as one key, however encompassing the ability of all the keys being aggregative. In alternative words, the key holder will unleash a constant-size combination key for versatile selections of cipher text set in cloud storage; however the opposite encrypted files outside the set stay confidential. This compact combination key may be handily sent to others or be keep during a} revolving credit with very restricted secure storage. We offer formal security analysis of our schemes within the customary model.

Keywords: *Cloud storage, Data Sharing, Cryptography, and symmetric key, Key Generation.*

1. Introduction

Cloud storage is gaining quality recently. In enterprise settings, we tend to see the increase in demand for information outsourcing that assists within the strategic management of company information. It's conjointly used as a core technology behind several on-line services for private applications. Nowadays, it's straightforward to use at no cost accounts for email, photo album, file sharing and/or remote access, with storage size quite 25GB (or some greenbacks for a lot of than 1TB). in conjunction with this wireless technology, users will access most of their files and emails by a mobile phone in any corner of the globe. Considering information privacy, a conventional thanks to guarantee it is to accept the server to enforce the access control once authentication (e.g., [1]), which implies any sudden privilege step-up can expose all information. In a shared-tenancy cloud computing setting, things become even worse. Information from completely different shoppers will be hosted on separate virtual machines (VMs) however reside on one physical machine. Information during a target VM may be taken by instantiating another VM co-resident with the target one [2]. concerning handiness of files, there are a series of science schemes that go as so much as allowing a third-party auditor to ascertain the provision of files on behalf of the info owner while not leaky anything regarding the info [3], or while not compromising the data house owners namelessness. Likewise, cloud users probably won't hold the conviction that the cloud server is doing an honest job in terms of confidentiality. A cryptographic resolution, e.g, with evidenced security relied on number-theoretic assumptions is a lot of fascinating, whenever the user isn't dead proud of trusting the security of the VM or the honesty of the technical staff. These users area unit intended to write in code their information with their own keys before uploading them to the server.

1.1 Existing System

1.1.1 Compact key in symmetric key encryption

- Data owner encrypts his information employing a secret key and is uploaded to the cloud server.
- Whoever desires to share that information has to have same secret key for Decryption.

1.1.2 Cryptographic Keys for a Predefined Hierarchy

- It minimizes the expense of storing and managing secret keys
- It constructs a tree structure.
- When a key is assigned to a branch, the same key is used to derive the keys of its descendent nodes .

1.1.3 Identity based encryption

- It is a sort of public key coding there's a trustworthy party referred to as non-public Key Generator United Nations agency holds a key and generates a secret key to every user.
- This user secret secret is encrypted by master secret key and is issued to the users.
- The users will decode the message by exploitation secret key.

ISSN 2348 – 7968

1.2 Disadvantages of Existing System

- For access revocation rekeying is needed
- Re-Encryption is implemented on the hold on knowledge
- It is costlier than bilaterally symmetric key operations due to high complicated standard arithmetic operations
- More secret keys area unit generated
- Increases the value of storing and sending the cipher text

2. Proposed System

In this paper, we have a tendency to study a way to create a decoding key additional powerful within the sense that it permits decoding of multiple cipher texts, while not increasing its size. Specifically, our drawback statement is "To style Associate in Nursing economical public-key encoding theme that supports versatile delegation within the sense that any set of the cipher texts (produced by the encoding scheme) is reprobate ptable by a constant-size decoding key (generated by the owner of the master-secret key)." we have a tendency to solve this drawback by introducing a special sort of public-key encoding that we have a tendency to decision key-aggregate cryptosystem (KAC). In KAC, users encipher a message not solely underneath a public-key, however conjointly underneath Associate in nursing symbol of cipher text referred to as category. Meaning the cipher texts are additional classified into completely different categories. The key owner holds a master-secret referred to as master-secret key, which may be wont to extract secret keys for various categories. Additional significantly, the extracted key have will be Associate in Nursing combination key that is as compact as a secret key for one category, however aggregates the facility of the many such keys, i.e., the decoding power for any set of cipher text categories.

2.1 Advantages of Proposed System

- ✓ The extracted key have may be associate combination key that is as compact as a secret key for one category.
- ✓ The delegation of secret writing may be with efficiency enforced with the mixture key.

2.2 Contributions

In trendy cryptography, an elementary downside we regularly study is concerning leverage the secrecy of a tiny low piece of information into the flexibility to perform scientific discipline functions (e.g. encryption, authentication) multiple times. During this paper, we have a tendency to study the way to create a cryptography key a lot of powerful within the sense that it permits cryptography of multiple cipher texts, while not increasing its size. Specifically, our downside statement is-

"To style associate economical public-key secret writing theme that supports versatile delegation within the sense that any set of the cipher texts (produced by the secret writing scheme) is decrypt able by a constant-size cryptography key (generated by the owner of the master-secret key)."

We solve this downside by introducing a special style of public-key secret writing that we have a tendency to decision key-aggregate cryptosystem (KAC). In KAC, users encipher a message not solely beneath a public-key, however conjointly beneath associate symbol of cipher text known as category. Which means the cipher texts are more classified into totally different categories? The key owner holds a master-secret known as master-secret key, which might be wont to extract secret keys for various categories. A lot of significantly, the extracted key have will be associate mixture key that is as compact as a secret key for one category; however aggregates the ability of the many such keys, i.e., the cryptography power for any set of cipher text categories.

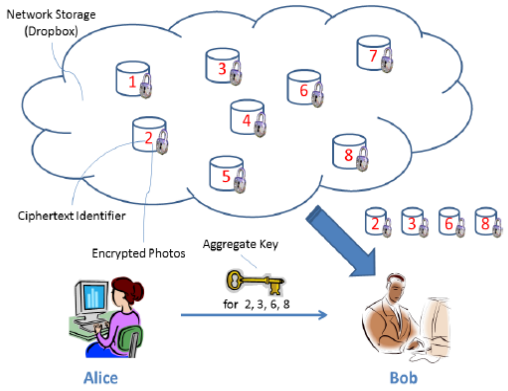

Fig. 1. Alice shares files with identifiers 2, 3, 6 and 8 with Bob by sending him a single aggregate key.

ISSN 2348 – 7968

In this architecture, the solution is Alice will merely send Bob one combination key via a secure e-mail. Bob will transfer the encrypted photos from Alice's Drop box area then use this combination key to decipher these encrypted photos. The situation is represented in Figure one. The sizes of cipher text, public-key, master-secret key and combination key in our KAC schemes area unit all of constant size. the general public system parameter has size linear within the range of cipher text categories, however solely a little a part of its required anytime and it are often fetched on demand from giant (but non-confidential) cloud storage.

2.3 System Architecture

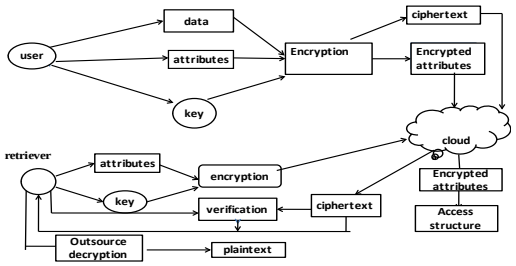

Fig 2.Architecture

In above architecture, the setup formula takes no input apart from the implicit security parameter. It outputs the general public parameters PK and a master MK.

Encrypt (PK, M, A). The encoding formula takes as input the general public parameters PK, a message M, associate degreed an access structure A over the universe of attributes. The formula can inscribe M and turn out a cipher text CT such solely a user that possesses a group of attributes that satisfies the access structure are able to decode the message. We'll assume that the cipher text implicitly contains A.

Key Generation (MK, S). The key generation formula takes as input the master MK and a group of attributes S that describe the key. It outputs a non-public key SK.

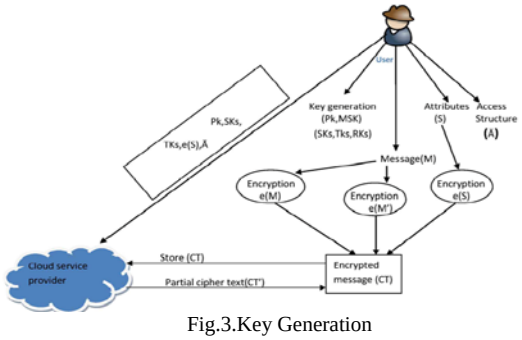

Fig.3.Key Generation

Decrypt (PK, CT, and SK). The coding formula takes as input the general public parameters PK, a cipher text CT, that contains associate degree access policy A, and a private key SK, that could be a non-public key for a group S of attributes. If the set S of attributes satisfies the access structure A then the formula can decode the cipher text and come a message M.

3. Related Work

This section we compare our basic KAC scheme with other possible solutions on sharing in secure cloud storage.
We summarize our comparisons in Table 1.

	Decryption key size	Ciphertext size	Encryption type
Key assignment schemes for a predefined hierarchy (e.g., [7])	most likely non-constant (depends on the hierarchy)	constant	symmetric or public-key
Symmetric-key encryption with Compact Key (e.g., [8])	constant	constant	symmetric-key
IBE with Compact Key (e.g., [9])	constant	non-constant	public-key
Attribute-Based Encryption (e.g., [10])	non-constant	constant	public-key
KAC	constant	constant	public-key

TABLE 1
Comparisons between our basic KAC scheme and other related schemes

3.1 Cryptographic Keys for a Predefined Hierarchy

We start by discussing the most relevant study in the literature of cryptography/security. Cryptographic key assignment schemes aim to minimize the expense in storing and managing secret keys for general cryptographic use. Utilizing a tree structure, a key for a given branch can be used to derive the keys of its descendant nodes (but not the other way round). Just granting the parent key implicitly grants all the keys of its descendant nodes. Sandhu proposed a method to generate a tree hierarchy of symmetric keys by using repeated evaluations of pseudorandom function/block-cipher on a

ISSN 2348 – 7968

fixed secret. The concept can be generalized from a tree to a graph. More advanced cryptographic key assignment schemes support access policy that can be modeled by an acyclic graph or a cyclic graph [7]. Most of these schemes produce keys for symmetric-key cryptosystems, even though the key derivations may require modular arithmetic as used in public-key cryptosystems, which are generally more expensive than "symmetric-key operations" such as pseudorandom function. We take the tree structure as an example. Alice can first classify the cipher text classes according to their subjects like Figure 3. Each node in the tree represents a secret key, while the leaf node represents the keys for individual cipher text classes. Filled circles represent the keys for the classes to be delegated and circles circumvented by dotted lines represent the keys to be granted. Note that every key of the non-leaf node can derive the keys of its descendant nodes.

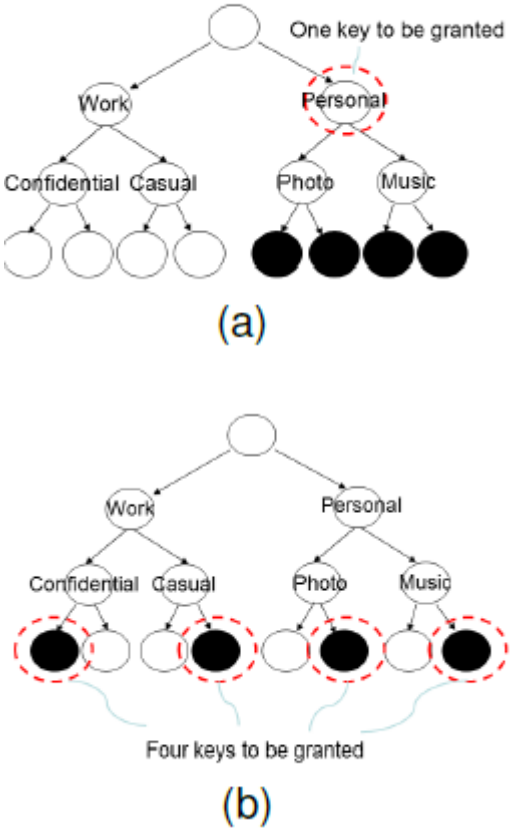

Fig. 4. Compact key is not always possible for a fixed hierarchy

In parent 4(a), if Alice wants to percentage all the files in the "personal" class, she only wishes to supply the important thing for the node "private", which automatically presents the delegate the keys of all the descendant nodes ("picture", "tune"). This is the precise case, wherein most classes to be shared belong to the identical branch and consequently a figure key of them is sufficient. However, it's far nonetheless difficult for widespread cases. As proven in figure 4(b), if Alice stocks her demo tune at paintings ("paintings"! "Casual"! "Demo" and "paintings"! "Confidential"! "demo") with a colleague who also has the rights to look a number of her private information, what she will be able to do is to present greater keys, which leads to an increase in the overall key size. You'll see that this method is not bendy whilst the classifications are extra complicated and she desires to proportion distinctive sets of files to distinctive human beings. For this delegate in our example, the wide variety of granted mystery keys turns into the same as the range of training take the tree structure as an example. Alice can first classify the cipher text classes according to their subjects like Figure 4. Each node in the tree represents a secret key, while the leaf nodes represent the keys for individual cipher text classes. Filled circles represent the keys for the classes to be delegated and circles circumvented by dotted lines represent the keys to be granted. Note that every key of the non-leaf node can derive the keys of its descendant nodes.

3.2 Compact Key in Symmetric-Key Encryption

Stimulated via the identical hassle of supporting bendy hierarchy in decryption energy delegation (however in symmetric-key placing), Beano et al. [8] supplied an encryption scheme that's in the beginning proposed for concisely transmitting huge range of keys in broadcast situation. the construction is easy and we in short evaluation its key derivation method here for a concrete description of what are the applicable houses we want to attain.

3.3 Compact Key in Identity-Based Encryption

Identification-primarily based encryption (IBE) is a form of public-key encryption in which the general public-key of a person can be set as an identification-string of the user (e.g., an electronic mail deal with). there may be a relied on birthday party called non-public key generator (PKG) in IBE which holds a master-mystery key and issues a mystery key to each consumer with appreciate to the user identification. The encryption can take the public parameter and a person identification to encrypt a message. The recipient can decrypt this cipher text by means of his mystery key.

3.4 Other Encryption Schemes

Characteristic-based totally encryption (ABE) permits every cipher text to be related to an attribute, and the

ISSN 2348 – 7968

grasp-secret key holder can extract a mystery key for a policy of those attributes so that a cipher text may be decrypted by this key if its associated characteristic conforms to the coverage. for example, with the secret key for the coverage, you can actually decrypt cipher text tagged with elegance 2; three; 6 or eight. However, the fundamental difficulty in ABE is collusion-resistance however no longer the compactness of secret keys. Indeed, the dimensions of the key often increase linearly with the range of attributes it encompasses, or the cipher text-size is not consistent.

4. Simulated Result

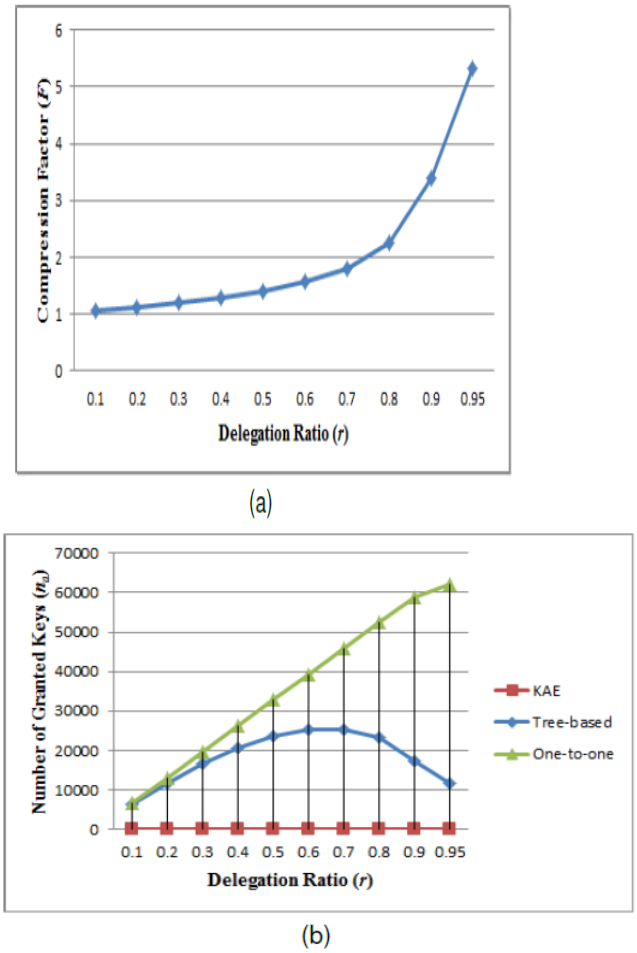

Fig. 5 (a) Compression achieved by the tree-based approach for delegating different ratio of the classes (b) Number of granted keys (na) required for different approaches in the case of 65536 classes of data

Figure 5(a) illustrates the relationship among the compression thing and the delegation ratio. somewhat particularly, we observed that F = three:2 even for delegation ratio of r =0:nine, and F < 6 for r = zero: ninety five, which deviates from the intuition that only a small variety of "powerful" keys are wanted for delegating most of the classes. We are able to most effective get a high (but still small) compression component while the delegation ratio is near 1. A contrast of the variety of granted keys among three techniques is depicted in discern 5(b). We are able to see that if we grant the important thing one by one, the number of granted keys might be identical to the number of the delegated cipher text instructions. With the tree-based shape, we can keep a number of granted keys in keeping with the delegation ratio. On the opposite, in our proposed approach, the delegation of decryption may be correctly implemented with the combination key, which is handiest of constant length. In our experiment, the delegation is randomly selected. It models the state of affairs that the needs for delegating to distinctive users won't be predictable as time goes by way of, even after cautious initial making plans. This gives empirical evidences to support our thesis that hierarchical key mission does not store lots in all cases.

4. Literature Survey

4.1 Study about Privacy-Preserving Public Auditing for Secure Cloud Storage

Using Cloud garage, users can remotely store their information and experience the on-call for excessive excellent applications and offerings from a shared pool of configurable computing resources, without the load of nearby records garage and renovation. However, the fact that customers not have physical possession of the outsourced information makes the information integrity protection in Cloud Computing an impressive assignment, mainly for users with limited computing resources. Moreover, users must be able to just use the cloud storage as if it is nearby, without annoying about the want to confirm its integrity. for this reason, permitting public audit capability for cloud storage is of crucial importance so that customers can inn to a 3rd birthday party auditor (TPA) to check the integrity of outsourced records and be worry-unfastened. To safely introduce an powerful TPA, the auditing technique need to carry in no new vulnerabilities toward person information privatives, and introduce no additional online burden to user. In this paper, we endorse a at ease cloud storage gadget helping privacy-preserving public auditing. We in addition expand our end result to allow the TPA to carry out audits for a couple of users simultaneously and efficaciously. Vast security and overall performance analysis show the proposed schemes are provably relaxed and extraordinarily efficient.

ISSN 2348 – 7968

4.1 Study about Storing Shared Data on the Cloud via Security-Mediator

Nowadays, many businesses outsource data garage to the cloud such that a member of a company (facts owner) can without difficulty percentage facts with different members (customers). Because of the existence of safety issues inside the cloud, both proprietors and users are recommended to verify the integrity of cloud statistics with Provable records ownership (PDP) before in addition usage of data. However, preceding methods either unnecessarily reveal the identity of a facts owner to the untrusted cloud or any public verifiers, or introduce vast overheads on verification metadata for keeping anonymity. On this paper, we advocate a easy, green, and publicly verifiable technique to ensure cloud information integrity without sacrificing the anonymity of facts proprietors nor requiring considerable overhead. Specifically, we introduce a safety-mediator (SEM), which is able to generate verification metadata (i.e., signatures) on outsourced data for information owners. Our method decouples the anonymity protection mechanism from the PDP. as a result, an corporation can appoint its very own nameless authentication mechanism, and the cloud is oblivious to that because it most effective deals with ordinary PDP-metadata, consequently, the identity of the statistics proprietor is not found out to the cloud, and there may be no more storage overhead in contrast to present anonymous PDP answers. The one-of-a-kind features of our scheme also encompass facts privatives, such that the SEM does no longer study anything approximately the statistics to be uploaded to the cloud in any respect, and as a consequence the trust on the SEM is minimized. in addition, we extend our scheme to work with the multi-SEM model, that can keep away from the capacity single factor of failure. Protection analyses prove that our scheme is at ease, and experiment effects demonstrate that our scheme is efficient.

4. Conclusions and Future Work

How to guard customers' records privatives is a significant query of cloud garage. With greater mathematical gear, cryptographic schemes have become extra flexible and frequently involve more than one key for a unmarried application. In this newsletter, we don't forget how to "compress" mystery keys in public-key cryptosystems which aid delegation of mystery keys for different cipher text instructions in cloud garage. No matter which one many of the power set of classes, the delegate can continually get a combination key of consistent length. Our approach is extra flexible than hierarchical key project that may best shop areas if all key-holders proportion a comparable set of privileges.

An issue in our work is the predefined certain of the wide variety of most cipher text instructions. In cloud storage, the quantity of cipher texts typically grows rapidly. So we should reserve sufficient cipher text lessons for the future extension. in any other case, we need to extend the public-key as we described in segment . Although the parameter may be downloaded with cipher texts, it would be higher if its size is unbiased of the maximum number of cipher text instructions. On the opposite hand, whilst one carries the delegated keys around in a mobile tool without the usage of unique relied on hardware, the secrets activate to leakage, designing a leakage resilient cryptosystem but permits efficient and bendy key delegation is also an exciting course.

References

[1] S. S. M. Chow, Y. J. He, L. C. K. Hue, and S.-M. You, "SPICE -Simple Privacy-Preserving Identity Management for Cloud Environment," in Applied Cryptography and Network Security – ACNS 2012, ser. LNCS, vol. 7341. Springer, 2012, pp. 526–543.
[2] L. Hardesty, "Secure computers aren't so secure," MIT press, 2009, http://www.physorg.com/news176107396.html.
[3] C. Wang, S. S. M. Chow, Q. Wang, K. Reno, and W. Lou, "Privacy Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362–375, 2013.
[4] B. Wang, S. S. M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," in International Conference on Distributed Computing Systems - ICDCS 2013. IEEE, 2013.
[5] S. S. M. Chow, C.-K. Chu, X. Huang, J. Zhou, and R. H. Deng, "Dynamic Secure Cloud Storage with Provenance," in Cryptography and Security: From Theory to Applications - Essays Dedicated to Jean-Jacques Quisquater on the Occasion of His 65th Birthday, ser. LNCS, vol. 6805. Springer, 2012, pp. 442–464.
[6] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," in Proceedings of Advances in Cryptology - EUROCRYPT '03, ser. LNCS, vol. 2656. Springer, 2003, pp. 416–432.
[7] M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.
[8] J. Beano, M. Chase, E. Horvitz, and K. Lauter, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," in Proceedings of ACM Workshop on Cloud Computing Security (CCSW '09). ACM, 2009, pp. 103–114.
[9] F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi-Identity Single-Key Decryption without Random Oracles," in Proceedings of Information Security and Cryptology (Inscrypt '07), ser. LNCS, vol. 4990. Springer, 2007, pp. 384–398.
[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications
Security (CCS '06). ACM, 2006, pp. 89–98.

G. Suresh received the B.Tech Degree in Computer Science and Engineering from Chadalawada Ramanamma Engineering College, University of JNTUA in 2012.He is currently working towards the Master's Degree in Computer Science and Engineering, in Shree Institute of Technology & Sciences University of JNTUA. He interest lies in the

ISSN 2348 – 7968

areas of Web Development Platforms, SQL, and Cloud Computing Technology.

A. Narayana received PhDs in SVU. Currently he is an Assistant Professor in the Department of Computer Science and Engineering at Shree Institute of Technology & Sciences-Tirupati.