Several Outflow Straight with Unspecified Identity and Data Security in Cloud Computing

[1]Malla Lakshmi Bhargavi, [2]N.Veeranjaneyulu

[1,2]Dept of CSE, Sri Chundi Ranganayakulu Engineering College, Chilakaluripet, Guntur Dt, AP, India

Abstract

Cloud computing is new model where we can get platform as a service, software as a service and infrastructure as a service. The straight supports multiple read and writes on the data stored in the cloud. It proposing security preserving authenticated access control model In the scheme a user to create a file and store and securely in the cloud. We propose new secure cloud storage system that addresses security and storage model for cloud computing locations. This Scheme prevents Replay attack which mean Eaves Dropping can be avoided, Support Creation of data inside storage, Changes the data unknown people and Reading data stored in Cloud. The model prevents replay attacks and supports creation, modification, and reading data stored in the cloud. We also address user revocation. Our authentication is access control scheme and decentralized and robust, unlike other access control schemes designed for clouds which are centralized. Our Identity and access control model is decentralized and security unlike other access control schemes designed for clouds which are centralized. The communication, computation, and storage overheads are comparable to centralized model. An efficient and secure dynamic auditing scheme is desired to change data owners that the data is correctly stored in the cloud. There are security models in the client and the service provider. The communication, computation, and storage overheads are comparable to centralizedmodels.

Keywords

Access Control, Anonymity Authentication, Key Management, Homomorphism Encryption, Cloud Storage, Renewal Policy, Matching Dependencies(MDs)

I. Introduction

The Security of storage is not only enough to store , the user must also check the anonymity of the user. For example the user wants to post a comment on Article but doesn't want his/her to disclosed. There are three cryptographic protocols such as Ring Signature, Mesh Signature ,Group Signature [1]. The Ring Signature which meam a large number of users are been involved so it is not feasible. Security and privacy protection in clouds are being explored by many researchers. Wang. [2] addressed storage security using Reed-Solomon erasure-correcting codes. Authentication of users using public key crypto- graphic techniques has been studied in [5]. Many homo- morphic encryption techniques have been suggested [6-7] to ensure that the cloud is not able to read the data while performing computations on them. Using homomorphic encryption, the cloud receives ciphertext of the data and performs computations on the ciphertext and returns the encoded value of the result [3]. 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. A third party auditor (auditor) that has expertise and capabilities can do a more efficient work and convince both cloud service providers and user. For the third party auditing in cloud storage systems, it has several important requirements which have been proposed in some previous works. The auditing protocol should have the following properties:

1. Confidentiality

The auditing protocol should keep user's data confidential against the auditor.

2. Dynamic Auditing

The auditing protocol should support the dynamic updates of the data which is stored in the cloud.

3. Batch Auditing

The auditing protocol should also be able to support the batch auditing for multiple user and multiple clouds [4]. Accountability of clouds is a very challenging task and involves technical issues and law enforcement. Neither clouds nor users should deny any operations performed or requested. It is important to have log of the transactions performed; however, it is an important concern to decide how much information to keep in the log. Accountability has been addressed in TrustCloud [8]. Secure provenance has been studied in [9].

A. Our Contributions in this Paper are Multirole

- To identify whether the user is protected from the cloud during authentication.
- The architecture is decentralized, meaning that there should be several KDCs for key management.
- The access control data and authentication are both collusion resistant, that means two users can collude and access data or authenticate themselves, if they are individually not authorized.

In proposed a system in which a sender can encrypt a message specifying an attribute set and a number d, such that only a recipient with at least d of the given attributes can decrypt the message. the deployment implications of their scheme may not be entirely realistic[10]in that it assumes the existence of a single trusted party who monitors all attributes and issues all decryption keys. Instead, we often have different entities responsible for monitoring different attributes of a person, e.g. the Department of Motor Vehicles tests whether you can drive, a university can certify that you are a student, etc. Thus, Chase gave a multi-authority ABE scheme which supports many different authorities operating simultaneously, each handling out secret keys for a different set of attributes. One method for alleviating some of these problems is to store data in encrypted form. Thus, if the storage is compromised the amount of information loss will be limited. One disadvantage of encrypting data is that it severely limits the ability of users to selectively share their encrypted data at a fine-grained level [11].

B. Motivation

Existing methods works on access control in cloud are centralized in nature. Except some all other schemes use ABE. The schemes use a symmetric key approach and does not support authentication. The most previous schemes do not support authentication as well. Much of the previous work takes 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. Therefore, the expert 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.

III. Existing System

Existing work on access control in clouds are centralized in nature. All schemes use ABE or symmetric key approach and does not support authentication. Earlier work provides privacy preserving authenticated access control in cloud. 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 [12]. 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. Although a decentralized approach is proposed in some of the existing papers, their technique does not authenticate users, who want to remain anonymous while accessing the cloud. In an earlier work, a distributed access control mechanism in clouds was proposed. The scheme did not provide user authentication. The other draw back was that a user can create and store a file and other users can only read the file. Write access was not permitted to users other than the creator Many homomorphism 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 [13]. The user is able to decode the result, but the cloud does not know what data is has operated on. In such circumstances, it must be possible for the user to verify that the cloud returns correct results.

IV. Related Work

ABE was proposed by Sahai and Waters [17]. In 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 [18] 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 ([19-20]), 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. Chase [21] proposed a multiauthority ABE, in which there are several KDC authorities which distribute attributes and secret keys to users. Multiauthority ABE protocol was studied in [14], which required no trusted authority which requires every user to have attributes from at all the KDCs. Recently, Lewko and Waters 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 [15]. To get over this problem, Green proposed to outsource the decryption task to a proxy server, so that the user can compute with minimum resources the presence of one proxy and one KDC makes it less robust than decentralized approaches. Both these approaches had no way to authenticate users, anonymously. Yang presented a modification of, authenticate users, Chase [13] proposed a scheme in which there are several KDC authorities which distribute attributes and secret keys of the users. However, the presence of one proxy and one KDC makes it less robust than decentralized approach. A new scheme given by Maji et al. takes a decentralized approach and provides authentication without disclosing the identity of the users. the presence of single proxy and one KDC makes it less robust than decentralized approaches. Both approaches had no way to authenticate users, anonymously. Authenticate users, who need to remain anonymous while accessing the cloud [16].

V. Proposed Work

The proposed architecture is a decentralized one where multiple numbers of KDCs are present for key distribution and management. These KDCs are geographically dispersed few users a scenario is presented with user 1 as owner of the file, user 2 as reader and user 3 as writer. These users are organized according to their roles based on their designation in the organization. If the user 1 wants to upload his file to the cloud he first needs to get registered to his corresponding KDCs. The output of this registration process is the generation of a unique user identifier for that user by the KDC. This user ID will be further used for all operations being performed by the user in the cloud. First level of authentication is achieved by a registration process where the users are identified as a legitimate [5]. The trustee can be assumed as a trusted third party such as a government organization who uniquely identifies the users with some proof for instance, passport, vote id, driving license etc. This trustee system will generate a token for the user once he produces his unique Id to the trustee system.

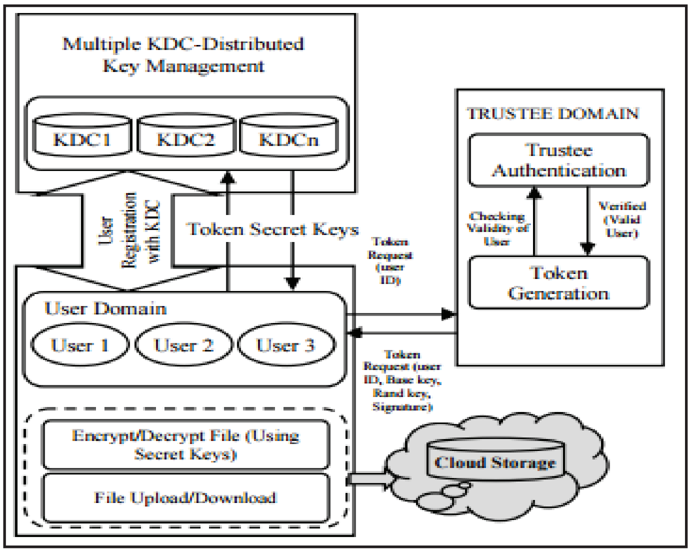


Fig. 1: Overall View of Proposed Architecture

A. Security Related Improvements

1. We implement a decentralized architecture is implemented with multiple Key Distribution Centre (KDC) structure [1].
2. We implement a Role Based Access Control (RBAC) [11].
3. We achieve anonymous authentication is achieved by implementing a strong digital signature algorithm (SHA -1 hash function) where the attributes of users are hidden from cloud [13].
4. The access policies that are set by users are hidden from other users by implementing Query driven approach.

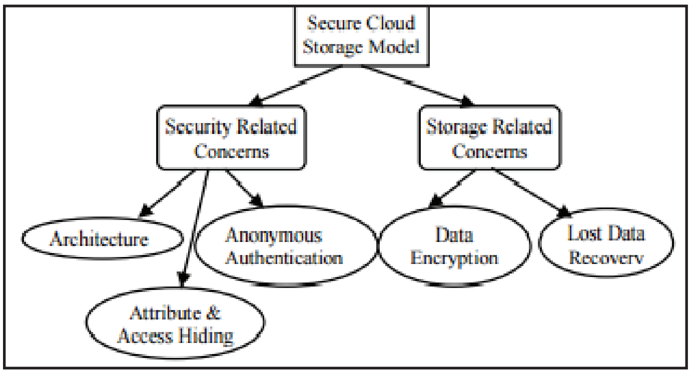


Fig.2. Overview of Proposed Work

B. Cryptographic Approach

1. Blind Signatures

This set of solutions can be classified as using Trusted Third Party architecture, because they assume the existence of an entity besides the service provides which manages the authentication of users. In [17] the authors present a scheme to generate an authorized anonymous ID, which replaces the real user ID. The scheme contemplates two phases: (1) Registration (generation of anonymous ID using blind signatures) and (2) Controlled Connection (Connection is provided given a valid anonymous ID is presented). An additional phase of re-confusion is introduced to replace an old anonymous ID. In [18] the authors show a new mechanism for improving the registration and reconfusion phases in [17]. The mechanism implements blind signatures based on bilinear pairings, which seeks to delete the likability of the real ID with the authorized anonymous ID. In [19] the authors describe a privacy protection scheme to preserve user's privacy during authentication and access control phases. It provides mutual authentication while allowing the users to anonymously interact with the desired service. They use the cryptographic primitives of blind signatures and hash chains in order to achieve these objectives through the authentication and key establishment protocols. The solutions in [17-19] are vulnerable to a location tracking attack, where attackers analyze the moving path of a mobile user with the aid of location information, which they have collected. It provides mutual explicit authentication between the mobile user and LBS provider and at the same time allows the user to interact anonymously. Blind signature scheme provides the generation of an authorized anonymous ID and ring signatures are used to mix the anonymous ID with a group of other authorized IDs

C. Fuzzy Identity-Based Encryption

A. Sahai and B. Waters,(2005) Proposed a new Identity-Based Encryption (IBE) scheme that is called as Fuzzy Identity-Based Encryption ,A Fuzzy IBE private key was identity by ω whereas the ciphertext encrypted is identified by ω'.It identities ω and ω' are close to each other as measured by the "set overlap" distance metrics. It used to apply the Encryption by obtaining the biometric input as identifier which inherently will have some noise each time they are sampled.Thus it is used for a type of application that we term "attribute-based encryption".In this paper two construction of Fuzzy IBE scheme are involved where the Identity-Based Encryption of a message under several attributes that compose a (fuzzy) identity. Hence in this scheme both are error-tolerant and secure against collusion attacks.

D. Cloud App

Cloud App is a module which acts as an intermediate between the user and other modules. User has access only to the Cloud App and other functions are being carried out through forwarding the requests to appropriate modules.

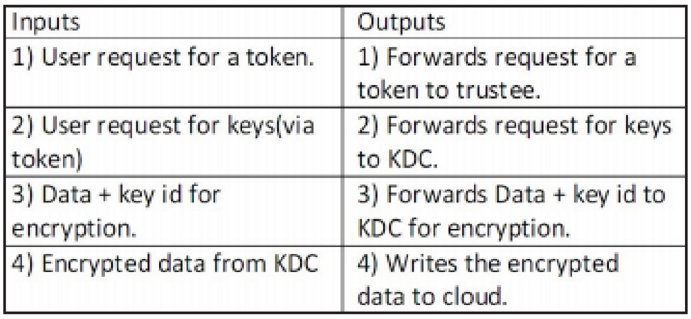
Inputs:

1. User request for a token. When an employee wants to write data to a cloud, he has to request for a token first. Token is like an identifier. This is the way in which anonymous authentication is carried out.[7]
2. User request for keys. Keys can be requested only if the token is obtained and is a valid one. Keys are used for encryption of data. Requests for a keys by passing tokens along with the request.
3. Data +key pointers for encryption. Once the request for keys is validated and if the token happens to be a valid one.He sends this key id along with the data to Cloud App for data encryption. [19]
4. Encrypted data from KDC. After the key id +data encrypted by the KDC, the encrypted data is sent to the Cloud App for uploading it to cloud.

Corresponding Outputs:

1. Forwards request for token to trustee. Once the employee requests for a token, this request is forwarded to the trustee module where the token is generated and admin logs in from the trustee module to approve or reject the token.[4]
2. Forwards request for keys to KDC. Once the token is approved and is sent to employee, he requests for keys for encryption purpose. This request is forwarded to KDC by the Cloud App.
3. Forwards Data+key id to KDC for encryption. After user gets key pointers, he sends the data along with the key pointers to Cloud App for encrypting the data. Cloud App [8]
4. Writes the encrypted data to Cloud. KDC encrypts the data and sends this encrypted data to Cloud App. It is the responsibility of the Cloud App to write this encrypted data to the cloud.

Table 1: Work Flow of Cloud App



Inputs	Outputs
1) User request for a token.	1) Forwards request for a token to trustee.
2) User request for keys(via token)	2) Forwards request for keys to KDC.
3) Data + key id for encryption.	3) Forwards Data + key id to KDC for encryption.
4) Encrypted data from KDC	4) Writes the encrypted data to cloud.

To encrypt every data block with a different key the flexible cryptography-based access control is used. Through this key derivation methods, the owner should maintain only a few secrets in the storage. and this key derivation procedure is used in hash functions which will introduce very limited computation .Thus to use over-encryption and or lazy revocation to prevent revoked users from getting access to updated data blocks. A Mechanism is used to handle both updates to outsourced data and changes in user access rights. Hence it is investigated in the overhead and safety of the proposed approach.

VI. Performance Analysis

A. Time Performance

The performance of this paper was analysed under various file sizes. At first the time performance of this paper is evolved for different file sizes. Then the cryptographic operation time is evolved. The only achievement of this paper is, it supports random time duration for any size of files to download

Table 2: Time Performance for Transaction on Cloud

FileSize	Upload (sec)	Download (sec)
10bytes	15	0
1kb	17	3
10kb	19	0
100kb	20	7
1mb	22	7

B. Upload

File uploading time is not a constant one. For same size file the time taking for uploading is randomly different. Using the time taken to upload the file one can identify the encryption standard. To confuse the hacker the random time delay is achieved.

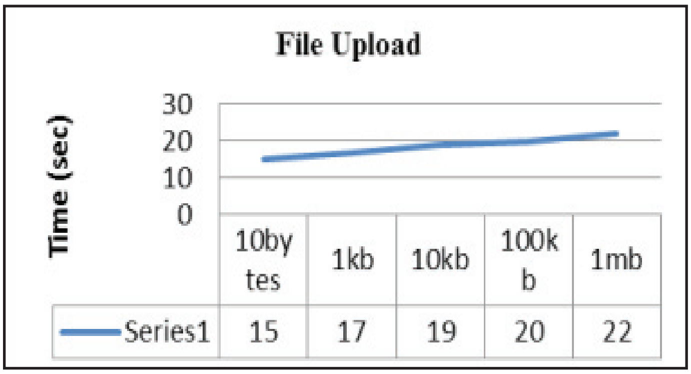


Fig. 3: Performance Analysis of File Upload Process

C. Download

File downloading time is also not a constant one. For same size file the time taking for downloading is randomly different. Using the time taken to download the file one can identify the encryption standard. To confuse the hacker the random time delay is achieved.

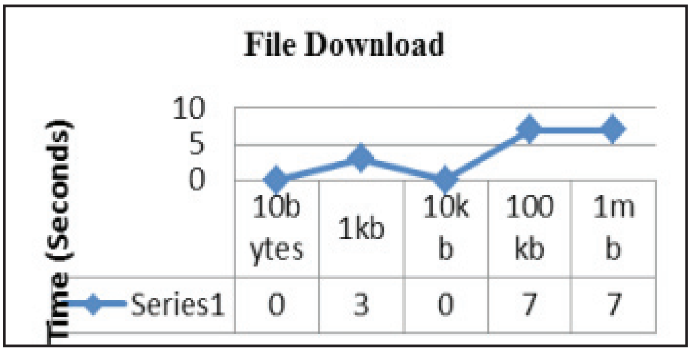


Fig. 4: Performance Analysis of File Download Process

VII. Conclusion

The Cloud which is a Secured storage area where the anonymous authentication is used the permitted users can be accessed. Decrypting of data can be viewed only by a valid users and can also stored information only by Valid users. This Scheme prevents Replay attack which mean Eaves Dropping can be avoided, Support Creation of data inside storage, Modifying the data by unknown users. The cloud does not know the identity of the user who stores information, but only verifies the user's credentials. Key distribution is done in a decentralized way. One limitation is that the cloud knows the access policy for each record stored in the cloud. We have presented the identity and authentication of data stored in cloud, which prevents replay attacks. The cloud does not know the user identity who stores information in cloud. Key distribution is done only in decentralized way not centralized.

VIII. Future Work

In future we would like to hide the attributes and access policy of a user Further, storage related security issues are enhanced by implementing a Homomorphism encryption technique to encrypting the outsourced data. Also, the cloud servers are prone to various types of attacks that can cause data loss or leakage. This issue is addressed by implementing a string matching algorithm that detects deviations and automatically retrieves the lost data using backed-up data. the file access policy can be implemented with Multi Authority based Attribute based Encryption. Using the technique we can avoid the number of wrong hits during authentication. Create a random delay for authentication, so the hacker can confuse to identify the algorithm.

References

[1] S. Ruj, M. Stojmenovic, A. Nayak,"Privacy Preserving Access Control with Authentication for Securing Data in Clouds," Proc. IEEE/ACM Int'l Symp. Cluster, Cloud and Grid Computing, pp. 556- 563, 2012.

[2] C. Wang, Q. Wang, K. Ren, N. Cao, W. Lou,"Toward Secure and Dependable Storage Services in Cloud Computing," IEEE Trans. Services Computing, Vol. 5, No. 2, pp. 220-232, Apr.- June 2012.

[3] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, W. Lou,"Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," Proc. IEEE INFOCOM, pp. 441-445, 2010.

[4] S. Kamara, K. Lauter,"Cryptographic Cloud Storage," Proc. 14th Int'l Conf. Financial Cryptography and Data Security, pp. 136- 149, 2010.

[4] C. Wang, Q. Wang, K. Ren, N. Cao, W. Lou,"Toward Secure and Dependable Storage Services in Cloud Computing", IEEE Transactions on Services Computing, Vol. 5, No. 2, pp. 220-232, 2012.

[5] G. Wang, Q. Liu, J. Wu,"Hierarchical attribute-based encryption for fine-grained access control in cloud storage services", ACM Conference on Computer and Communications Security, pp. 735-737, 2010

[6] W. Wang, Z. Li, R. Owens, B. Bhargava,"Secure and efficient access to outsourced data", ACM Cloud Computing Security Workshop (CCSW), pp. 55-66, 2009

[7] A. Rahumed, H.C.H. Chen, Y. Tang, P.P.C. Lee, J.C.S. Lui,"A Secure Cloud Backup System with Assured Deletion and Version Control," Proc. Third Int'l Workshop Security in Cloud Computing, 2011.

[8] S. Yu, C. Wang, K. Ren, W. Lou,"Attribute Based Data Sharing with Attribute Revocation," Proc. Fifth ACM Symp. Information, Computer and Comm. Security (ASIACCS), Apr. 2010.

[9] W. Wang, Z. Li, R. Owens, B. Bhargava,"Secure and Efficient Access to Outsourced Data," Proc. ACM Workshop Cloud Computing Security (CCSW), Nov. 2009.

[10] J. Bethencourt, A. Sahai, B. Waters,"Ciphertext-Policy Attribute-Based Encryption," Proc. IEEE Symp. Security and Privacy, May 2006.

[11] F. Zhao, T. Nishide, K. Sakurai,"Realizing FineGrained 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.

[12] A. Sahai, B. Waters,"Fuzzy Identity-Based Encryption," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473, 2005.

[13] M. Chase,"Multi-Authority Attribute Based Encryption," Proc. Fourth Conf. Theory of cryptography (TCC), pp. 515-534, 2007.

[14] S. Kamara, K. Lauter,"Cryptographic Cloud Storage," Proc. 14th Int'l Conf. Financial Cryptography and Data Security, pp. 136- 149, 2010.

[15] H. Li, Y. Dai, L. Tian, H. Yang,"IdentityBased Authentication for Cloud Computing," Proc. First Int'l Conf. Cloud Computing (CloudCom), pp. 157-166, 2009.

[16] S. Ruj, A. Nayak, I. Stojmenovic,"DACC: Distributed Access Control in Clouds," Proc. IEEE 10th Int'l Conf. Trust, Security and Privacy in Computing and Communications (TrustCom), 2011.

[17] [Online] Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-corespec-cs01-en.pdf, 2013.

[18] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A View of Cloud Computing. Comm. of the ACM, 53(4), pp. 50– 58, Apr 2010.

[19] M. Chase, S. S. M. Chow,"Improving privacy and security in multi authority attribute-based encryption", In ACM Conference on Computer and Communications Security, pp. pp. 121–130, 2009.

[20] R.L. Rivest, A. Shamir, Y. Tauman,"How to Leak a Secret," Proc. Seventh Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 552-565, 2001.

[21] X. Boyen,"Mesh Signatures," Proc. 26th Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 210-227, 2007.

Malla Lakshmi Bhargavi born in Podili town, Prakasam Dt, Andhra Pradesh. She received B.Tech in CSE from Prakasam Engineering College, JNTU Kakinada in the year 2012. Presently she is pursuing M.Tech in CSE from Sri Chundi Ranganayakulu Engineering College, Chilakaluripet, Guntur Dt, Andhra Pradesh, India.

N.Veeranjaneyulu received M.Sc (Computers) Degree from Acharya Nagarjuna University, Guntur in 2008 and M.Tech (CSE) Degree from JNTU Kakinada in 2013. He has seven years of teaching experience. He joined as Assistant Professor in Sri Chundi Ranganayakulu Engineering College, Chilakaluripet, Guntur Dt, Andhra Pradesh. Presently he is working as Assistant Professor in CSE Department. He published international journal on Adaptive Provisioning of Human Expetise in Service Oriented systems. He attended Various National and International Workshops and Conferences related to Computer topics and Business Administrations.

.