ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

Secure P-MOD Scheme for Data-Sharing in Cloud Computing

[1]K. Venkateswarlu, [2]G. Siddu

[1] Assistant Professor, Dept. of Master of Computer Applications, Narayana Engineering College, Gudur.

[2] PG Scholar, Dept. of Master of Computer Applications, Narayana Engineering College, Gudur.

Abstract—Cloud computing has changed the way enterprises store, access and share data. Big data sets are constantly being uploaded to the cloud and shared within a hierarchy of many different individuals with different access privileges. With more data storage needs turning over to the cloud, finding a secure and efficient data access structure has become a major research issue. In this paper, a Privilege-based Multilevel Organizational Data-sharing scheme (P-MOD) is proposed that incorporates a privilege-based access structure into an attribute-based encryption mechanism to handle the management and sharing of big data sets. Our proposed privilege-based access structure helps reduce the complexity of defining hierarchies as the number of users grows, which makes managing healthcare records using mobile healthcare devices feasible. It can also facilitate organizations in applying big data analytics to understand populations in a holistic way. Security analysis shows that P-MOD is secure against adaptively chosen plaintext attack assuming the DBDH assumption holds. The comprehensive performance and simulation analyses using the real U.S. Census Income data set demonstrate that P-MOD is more efficient in computational complexity and storage space than the existing schemes.

Keywords: Cloud Computing, Big Data, Hierarchy, Privilege-Based Access, Sensitive Data, Attribute-Based Encryption, Mobile Healthcare.

1. INTRODUCTION

It was estimated that data breaches cost the United States' healthcare industry approximately $6.2 billion in 2016 alone [1]. To mitigate financial loss and implications on the reputation associated with data breaches, large multilevel organizations, such as healthcare networks, government agencies, banking institutions, commercial enterprises and etc., began allocating resources into data security research to develop and improve accessibility and storage of highly sensitive data.

One major way that large enterprises are adapting to increased sensitive data management is the utilization of the cloud environment. It was reported that more than half of all U.S. businesses have turned over to the cloud for their business data management needs [2]. The on-demand cloud access and data sharing can greatly reduce data management cost, storage flexibility, and capacity [3]. However, data owners have deep concerns when sharing data on the cloud due to security issues. Once uploaded and shared, the data owner inevitably loses control over the data, opening the door to unauthorized data access.

A critical issue for data owners is how to efficiently and securely grant privilege level-based access rights to a set of data. Data owners are becoming more interested in selectively sharing

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

information with data users based on different levels of granted privileges. The desire to grant level-based access results in higher computational complexity and complicates the methods in which data is shared on the cloud. Research in this field focuses on finding enhanced schemes that can securely, efficiently and intelligently share data on the cloud among users according to granted access levels.

A Privilege-based Multilevel Organizational Data-sharing scheme (P-MOD) is proposed. It builds on concepts presented in [6] to solve the problems of sharing data within organizations with complex hierarchies.

The main contributions in this project can be summarized as follows:

• We present multiple data file partitioning techniques and propose a privilege-based access structure that facilitate data sharing in hierarchical settings.

• We formally prove the security of P-MOD and show that it is secure against adaptively chosen plaintext attacks under the Decisional Bilinear Diffie-Hellman (DBDH) assumption.

• We present a performance analysis for P-MOD and compare it to three existing schemes [7]–[9] that aim to achieve similar hierarchical goals.

• We implement P-MOD and conduct comprehensive simulations under various scenarios using the real U.S. Census Income data set [10]. We also compare our results to simulations we have conducted for two other schemes [7], [9] under the same conditions.

2. RELATED WORK

Fuzzy Identity-Base Encryption (Fuzzy IBE) was introduced in [11] to handle data sharing on the cloud in a flexible approach using encryption. The ciphertext is shared on the cloud to restrict access to authorized users. In order for an authorized individual to obtain the data, the user must request a private key from a key-issuer to decrypt the encrypted data. Fuzzy IBE is a specific type of function encryption [12] in which both the private key of the data user and ciphertext are affiliated with attributes. Attributes are descriptive pieces of information that can be assigned to any user or object. Since attributes can be any variable, they provide more flexibility when granting data access. The scheme enables a set of descriptive attributes to be associated with a private key and the ciphertext shared on the cloud. If the private key of the data user incorporates the minimum threshold requirement of attributes that match those integrated within the ciphertext, the data user can decrypt it. Although this scheme allows complex systems to be easily defined using attributes, it becomes less efficient when used to express large systems or when the number of attributes increases.

Attribute-Based Encryption (ABE) schemes later emerged to provide more versatility when sharing data. These schemes integrate two types of constructs: attributes and access policies.

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

Access policies are statements that join attributes to express which users of the system are granted access and which users are denied. ABE schemes were introduced via two different approaches: Key-Policy Attribute-Based Encryption (KPABE) [13] and Ciphertext Policy Attribute-Based Encryption (CP-ABE) [7]. In KP-ABE, each ciphertext is labeled with a set off descriptive attributes, while each private key is integrated with an access policy. For authorized data users to decrypt the ciphertext, they must first obtain a private key from the key issuer to use in decryption. The key-issuer integrates the access policy into the keys generated. Data users can successfully decrypt a ciphertext if the set of descriptive attributes associated with the ciphertext satisfies the access policy integrated within their private keys. KP-ABE can achieve fine-grained access control and is more flexible than Fuzzy IBE. However, the data owner must trust the key-issuer to only issue private keys to data users granted the privilege of access. This is a limitation since the data owner ultimately forfeits control over which data users are granted access.

The main advantage of this scheme is that it provides leveled access structures which are integrated into a single access structure. As a result, storage space is saved as only one copy of the ciphertext is needed to be shared on the cloud for all data users. However, since this scheme uses a single access structure to represent the full hierarchy, the higher levels are forced to accommodate attributes of all the levels below. As the number of levels increases in the hierarchy, the number of attributes grows exponentially making this scheme infeasible on a large scale. A simplified and reduced access structure is proposed to reduce the computational complexity by removing all branches of the single access structure while keeping one full branch. The full branch consists of the root node, a set of transport nodes (one for each level), and the leaf nodes (attributes). However, in real-life applications, relationships within an organization are often built in a cross-functional matrix, making this a complicated solution when assigning privileges.

3. PROBLEM DEFENITION

Consider a data owner that possesses a data file F and wishes to selectively share different segments of it on the cloud among a set of data users based on certain access privileges. We assume that the data users can be ranked into a hierarchy that defines their access privileges.

Selectively sharing data files on the cloud becomes a burden on the data owner as the hierarchy grows (the access privileges increase in number) and/or as the access restrictions become more complex due to an increase in the sensitivity of the file segments. A trivial solution involves the data owner to use public key encryption. This solution would require the data owner to encrypt the same part of the data file once for each data user being granted

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

access then upload the resulting ciphertexts to the cloud. The data users would then fetch their uniquely encrypted parts of the file from the cloud and utilize their private keys to decrypt them. This method ensures that no unprivileged data user will gain access to any part of the data file even if that user is able to download the ciphertexts from the cloud. However, on a large scale, public key encryption becomes an inefficient solution due to the increase in the number of encryptions and large storage spaces required. Therefore, the challenge is to provide the data owners with an efficient, secure and privilege-based method that allows them to selectively share their data files among multiple data users while minimizing the required cloud storage space needed to store the encrypted data segments.

Design Goals

Based on the problem described above, we have the following design goals:

Privilege-Based Access: Data is shared in a hierarchical manner based on user privileges. Data users with more privileges (ranked at the higher levels of the hierarchy) are granted access to more sensitive parts of F than those with fewer privileges (ranked at the lower levels of the hierarchy).

Data Confidentiality: All parts of F are completely protected from unprivileged data users (including the storage space). Data users are entitled to access the parts of F corresponding to the levels they fall in and/or any other parts corresponding to the levels below with respect to their own.

Fine-Grained Access Control: The data owner has the capability to encrypt any part of F using any set of descriptive attributes he/she wishes, limiting access to only authorized data users. The set of descriptive attributes is defined by the data owner at the time of encryption and can be selected from an infinite pool.

Collusion Resistant: Two or more data users at the same/different level cannot combine their private keys to gain access to any part of F they are not authorized to access independently.

4. PROPOSED WORK

The general model of privilege-based data sharing among hierarchically-ranked data users is illustrated in Fig. 1. The system consists of four main entities: Data owner (DO), Data users (DU), Key-issuer.

As shown by Fig. 1, the data users ranked at the higher levels are granted access to more sensitive segments of file F than those ranked at lower levels. In our proposed scheme, the

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

hierarchy is not fixed nor predefined. It is defined by the data owners as they encrypt their files to be shared with a set of data users.

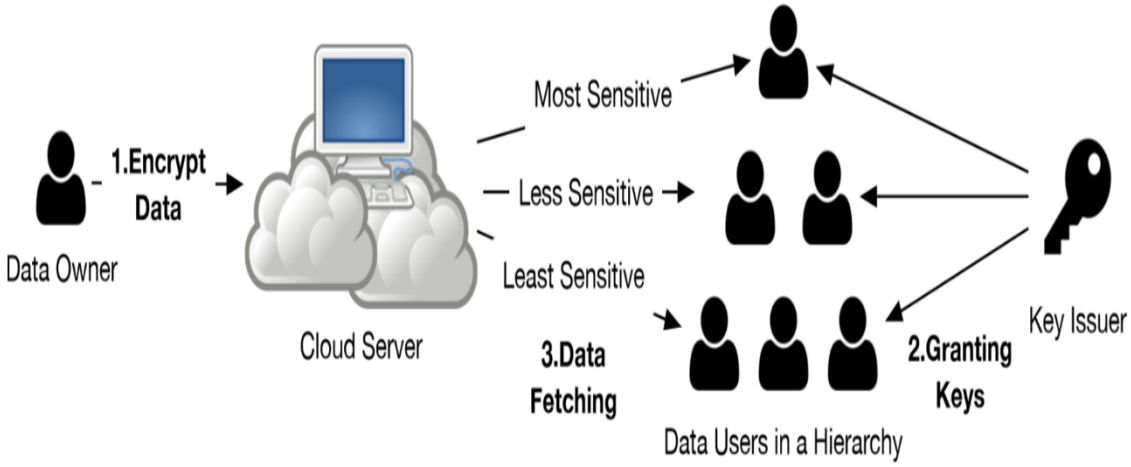


Fig. 1: General scheme of P-MOD

This system model consists and implements the following modules:

- **Data owner (DO):** An individual that owns a data file and wishes to selectively share it with multiple data users based on certain desired privacy preferences.
- **Data users (DU):** A set of hierarchically-ranked individuals interested in obtaining different segments of a shared data file. Data users fall into different levels within the hierarchy based on specific sets of attributes they possess.
- **Key-issuer:** A fully trusted entity that generates private keys for the data users that possess a correct set of attributes.
- **Cloud server:** A non-trusted entity used to store the encrypted segments of the data file.

Algorithm

1) *Setup:* This is a probabilistic function carried out by the key-issuer. The Setup function takes a security parameter and randomly chooses values as a input. The outputs of this function are public key PK and master key MK
2) *KeyGen:* This is a probabilistic function carried out by the key-issuer. The inputs to this function are MK generated by the Setup function, and the attribute set of data user, The KeyGen function outputs a unique private key SK for the data user.

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

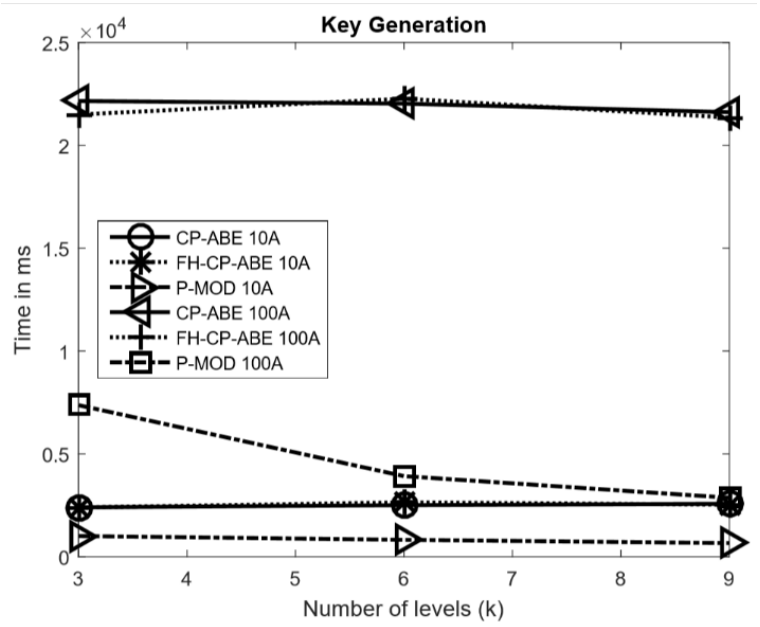
3) *Encrypt:* This is a probabilistic function carried out by the DO to encrypt the symmetric keys that are to be shared with the privileged data users. The inputs to this function are PK, the public key generated by the Setup function, the symmetric key *sk* representing the data that will be encrypted, and the access tree that defines the authorized set of attributes. The output of this function is the encrypted symmetric key.

4) *Decrypt:* This is a deterministic function carried out by the data user. The inputs to this function are an encrypted symmetric key corresponding to L_i, and the private key SK of the data user. This outputs the original file.

5. RESULTS & DISCUSION

In this section, we provide the proposed scheme comparative analysis results.

A) Key Generation Time-Cost:



(a) Key generation time

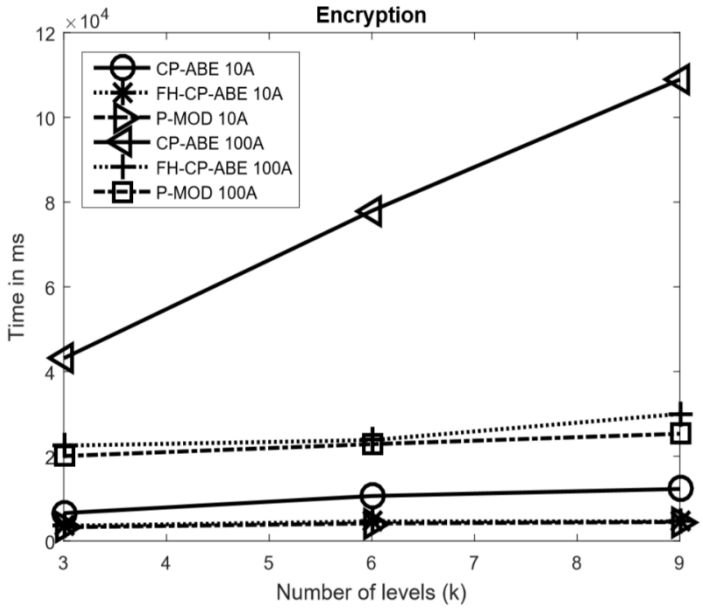
To measure the time to generate a private key for a user, the same attribute and level conditions are applied to all three schemes. P-MOD outperforms CP-ABE and FH-CP-ABE in all experimental evaluations. As illustrated in Fig. 2(a), the time taken to generate a private key for a user at the highest level in CP-ABE and FH-CP-ABE, is independent of the value of k and remains nearly constant when N is kept constant, for both values of N = {10,100} tested. In comparison, P-MOD reacts differently. When the total number of user attributes is normally distributed among the levels, the time taken to generate a private key by P-MOD is

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

approximately the reciprocal of the value of k multiplied by the equivalent time taken by CP-ABE or FH-CP-ABE to perform the same function.

B) Encryption Time-Cost:



(b) Encryption time

The encryption time-cost is the time it takes each scheme to perform the encryption function over all partitions of F. Fig. 2(b) represents the time expenditure of each scheme under all six experimental scenarios. P-MOD surpasses both CP-ABE and FH-CP-ABE in every experimental case. For example, compare the time duration of the three schemes at k = 9 and N = 100. The time duration for CP-ABE is approximately 4.3 times of P-MOD to perform encryption. Similarly, FH-CP-ABE is approximately 1.2 times of P-MOD to perform encryption.

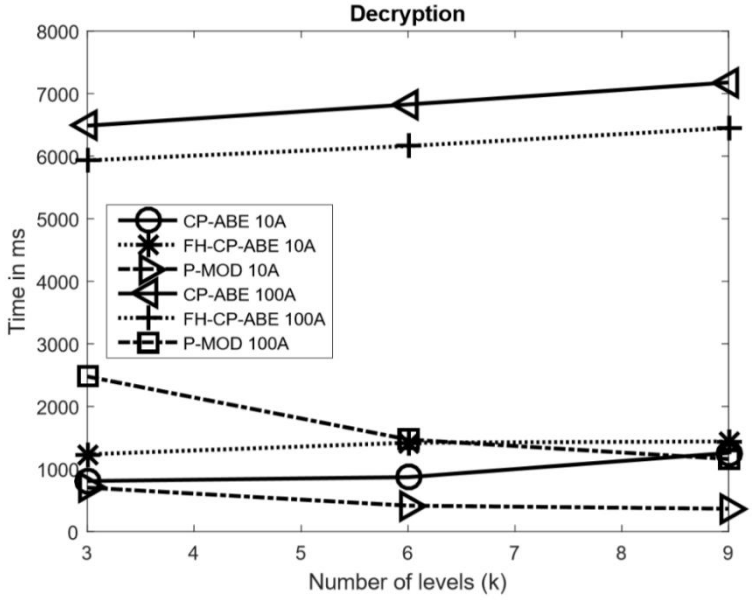
C) Decryption Time-Cost:

As previously discussed, the experiments are performed in the perspective of a user that appears at the highest level of the hierarchy. Taking this into account, the decryption time-cost is defined as the time for the user to successfully decrypt all ciphertexts EF_i corresponding to all partitions F_i, if the user possesses the correct set of user attributes. Fig. 2(c) illustrates the time to perform the decryption function by each scheme. The decryption function of both CP-ABE and FH-CP-ABE both involve e and fG_1 operations that are dependent on the values k and N.

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

As these values increase, the decryption time-cost increases linearly for both schemes, proving the correctness of the decryption complexity analysis. The decryption time-cost of P-MOD does not severely increase while the value N changes from 10 to 100. In contrast to this, CP-ABE and FH-CP-ABE are greatly affected, as seen in Fig. 2(c).



(c) Decryption time

Fig. 5: Performance comparison

In summary, for a hierarchical organization with many levels, the simulation results show that P-MOD is significantly more efficient at generating keys, encryption, and decryption than that of both CP-ABE and FH-CP-ABE schemes.

6. CONCLUSION

The numerous benefits provided by the cloud have driven many large multilevel organizations to store and share their data on it. This paper begins by pointing out major security concerns data owners have when sharing their data on the cloud. Next, the most widely implemented and researched data sharing schemes are briefly discussed revealing points of weakness in each. To address the concerns, this paper proposes a Privilege-based Multilevel Organizational Data sharing scheme (P-MOD) that allows data to be shared efficiently and securely on the cloud. P-MOD partitions a data file into multiple segments based on user privileges and data sensitivity. Each segment of the data file is then shared depending on data user privileges. We formally prove that P-MOD is secure against adaptively chosen plaintext attack assuming that the DBDH assumption holds. Our comprehensive performance and simulation comparisons with the three most representative schemes show that P-MOD can significantly reduce the computational complexity while

 Copyright © 2020 Authors

minimizing the storage space. Our proposed scheme lays a foundation for future attribute-based, secure data management and smart contract development.

REFERENCES

1. P. Institute, "Sixth annual benchmark study on privacy and security of healthcare data," tech. rep., Ponemon Institute LLC, 2016.
2. R. Cohen, "The cloud hits the mainstream: More than half of U.S. businesses now use cloud computing." http://www.forbes.com, April 2013. Online; posted 10-January-2017.
3. P. Mell and T. Grance, "The NIST definition of cloud computing," 2011.
4. A. C. O Connor and R. J. Loomis, "2010 economic analysis of role-based access control," NIST, Gaithersburg, MD, vol. 20899, 2010.
5. A. Elliott and S. Knight, "Role explosion: Acknowledging the problem.," in Software Engineering Research and Practice, pp. 349–355, 2010.
6. E. Zaghloul, T. Li, and J. Ren, "An attribute-based distributed data sharing scheme," in IEEE Globeocm 2019, (Abu Dhabi, UAE.), 9-13 December 2018.
7. J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attributebased encryption," in 2007 IEEE Symposium on Security and Privacy (SP'07), pp. 321–334, IEEE, 2007.
8. G. Wang, Q. Liu, J. Wu, and M. Guo, "Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers," Computers & Security, vol. 30, no. 5, pp. 320–331, 2011.
9. [9] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, pp. 1265–1277, 2016.
10. M. Lichman, "UCI machine learning repository," 2013.
11. A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 457–473, Springer, 2005.
12. D. Boneh, A. Sahai, and B. Waters, "Functional encryption: Definitions and challenges," in Theory of Cryptography Conference, pp. 253–273, Springer, 2011.
13. 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, pp. 89–98, Acm, 2006.

 Copyright © 2020 Authors

14. V.S. Viswanth, R. Ramanujam, and G. Rajyalakshmi, "A Novel MCDM Approach for Process Parameters Optimization in Eco-Friendly EDM of AISI 2507 Super Duplex Stainless Steel". Journal of Advanced Research in Dynamical and Control Systems - JARDCS, vol. 10, no. 7, pp. 54-64, 2018.
15. L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in Proceedings of the 14th ACM Conference on Computer and Communications Security, pp. 456–465, ACM, 2007.
16. S. Roy and M. Chuah, "Secure data retrieval based on ciphertext policy attribute-based encryption (CP-ABE) system for the dtns," tech. rep., Citeseer, 2009.
17. J. Lai, R. H. Deng, and Y. Li, "Fully secure cipertext-policy hiding CPABE," in International Conference on Information Security Practice and Experience, pp. 24–39, Springer, 2011.
18. J. Lai, R. H. Deng, and Y. Li, "Expressive CP-ABE with partially hidden access structures," in Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, pp. 18–19, ACM, 2012.
19. V.S. Viswanth, R. Ramanujam, and G. Rajyalakshmi, "Performance study of eco-friendly dielectric in EDM of AISI 2507 super duplex steel using Taguchi-fuzzy TOPSIS approach". International Journal of Productivity and Quality Management, vol. 29, no. 4, pp. 518-541, 2020.
20. F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE with constant-size keys for lightweight devices," IEEE Transactions on Information Forensics and Security, vol. 9, no. 5, pp. 763–771, 2014.
21. Penchalaiah P, Ramesh Reddy K, "Random Multiple Key Streams for Encryption with Added CBC Mode of Operation", Perspectives in Science (ISSN: 2213-0209), (ELSEVIER, UGC Journal no-62532), Volume 8, pp.57-62, April 2016.
22. Penchalaiah P, Ramesh Reddy K, "Secure and Cost Effective Cryptosystem Design Based on Random Multiple Key Streams", Journal of Information Security Research,(ISSN: 0976-4143) DIRF Publisher, Volume 7, Number 1, pp. 29-40, March 2016.
23. C. Gentry and A. Silverberg, "Hierarchical id-based cryptography," in International Conference on the Theory and Application of Cryptology and Information Security, pp. 548–566, Springer, 2002.

Author's Profile:

 Copyright © 2020 Authors

ISSN: 2278-4632
Vol-10 Issue-6 No. 7 June 2020

K. Venkateswarlu has received his MCA degree from Saraswathi Velu College of Engineering, Vellore affiliated to Anna University, Chennai in 2010 and MTech degree in Computer Science from PBR Vits, Kavali affiliated to JNTU, Ananthapur in 2014 respectively. He is dedicated to teaching field from the last 6years. He has guided 25 P.G students. At present he is working as an Assistant Professor in Narayana Engineering College, Gudur, Andhra Pradesh, India.

G. Siddu has Received his B.Sc Degree in Computer Science from Dr. CRR Degree College, Sydapuram affiliated to Vikrama Simhapuri University, Nellore in 2018 and pursuing PG Degree in Master of Computer Applications (M.C.A) from Narayana Engineering College, Gudur affiliated to JNT University, Anantapur, Andhra Pradesh, India.

 Copyright © 2020 Authors