# Enhanced Integrity Preserving Homomorphic Scheme for Cloud Using Diffie Hellman

[1]Tamanna Gupta, [2]Ritika Mehra
[1,2]Dept. of Computer Science and Engineering , RPIIT, Bastara Karnal, Karnal, Haryana, India

**Abstract**
Distributed computing, the new term for the since quite a while ago imagined vision of processing as an utility empowers helpful, on-request organize access to a unified pool of configurable figuring assets (e.g., systems, applications, and administrations) that can be quickly sent with incredible effectiveness and insignificant administration overhead. The stunning preferences of Cloud Computing include: on-request self-benefit, pervasive system get to, area autonomous asset pooling, quick asset versatility, utilization based valuing, transference of hazard, and so forth… Thus, Cloud Computing could without much of a stretch advantage its clients in maintaining a strategic distance from huge capital costs in the arrangement and administration of both programming and equipment. Without a doubt, Cloud Computing brings remarkable outlook changing and benefits ever of. As Cloud Computing winds up common, more touchy data are being incorporated into the cloud, for example, messages, individual well being records, private recordings and photographs, organization back information, government archives, and so forth. By putting away their information into the cloud, the information proprietors can be eased from the weight of information stockpiling and support in order to appreciate the on-request excellent information stockpiling administration. Be that as it may, the way that information proprietors and cloud server are not in the same trusted area may put the outsourced information in danger, as the cloud server may never again be completely confided in such a cloud domain because of various reasons: the cloud server may spill information data to unapproved elements or be hacked. Information security and protection are the basic issues for remote information stockpiling. A protected client upheld information get to control component must be given before cloud clients have the freedom to outsource touchy information to the cloud for capacity. With the development of sharing secret corporate information on cloud servers, it is basic to receive a proficient encryption framework with a fine-grained get to control to scramble outsourced information. Quality based encryption is an open key based encryption that empowers get to control over scrambled information utilizing access arrangements and credited properties. In this work, we are going to examination completely Homomorphic plans for encryption and honesty check utilizing MD5 Algorithm.

**Keyword**
Cloud Computing, Integrity Preserving Homomorphic, Diffie Hellman

## I. Introduction

Cloud computing, as an growing computing paradigm aiming to allocate storage, computation, and services transparently amid a large users, has gathered outstanding momentum from not merely industry but additionally academia. In core, cloud computing [1] overlaps countless continuing thoughts, such as distributed, grid and utility computing. Though, driven mainly by marketing and ability offerings from large company contestants like Google, IBM and Amazon, cloud computing has evolved out of these thoughts and come to be a new buzz word concentrating on“ cloud”—more hypothetical resource and services’ delivery. After cloud computing steps into our daily lifetimes, each innately stored data, such as email, word processing documents and spreadsheets, might be remotely stored in a cloud. Then, we can use each terminals, e.g., computer, laptop and PDA etc., to admission this data at anytime, anywhere. Due to these enthusing characteristics, cloud computing has come to be increasingly appealing to the public. The “cloud” in cloud computing can be described as the set of hardware, webs, storage, services, and interfaces that join to hold aspects of computing as a service [2-3]. Cloud services contain the transport of multimedia, groundwork, and storage above the Internet established on user demand. Cloud computing has four vital characteristics: elasticity and the skill to scale up and down, self-service provisioning and automatic DE provisioning, request software design interfaces (APIs), billing and metering of ability custom in a pay-as-you-go ideal Figure 1.1 below displays a normal cloud period on the web. This flexibility is what is appealing people and companies to move to the cloud. Pursuing are the insufficient gains of possessing an request hosted on the cloud:

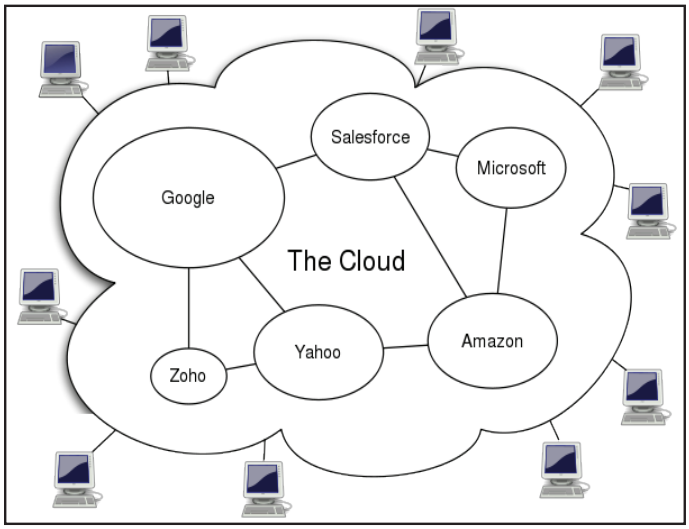


Fig. 1: General Architecture in Cloud Computing Environment

- It is the best cost efficient method to maintain, use and also to upgrade.
- Cloud computing is more cheap and also reduces the company’s expenditure. Here you can pay only for the cloud space you need.
- You can get the ultimate storage according to plans provided by the cloud provider.
- The cloud service provider who is responsible for IT assets and maintenance.
- It is easy to access information all over the world using internet connection .you can also store the documents to your office staff.
- It also decreases company’s carbon discharge by 35% .

Cloud computing can completely change the method firms use knowledge to ability clients, partners, and suppliers. A little companies, such as Google and Amazon, by now have most of their IT resources in the cloud. They have discovered that it can remove countless of the convoluted constraints from the established computing nature, encompassing space, period, domination, and cost.

## II. Literature Survey

**Liu, C., Yang, C., Zhang, X., and Chen, J., (2015) [5],** HighlightsSecurity of Big Data in cloud and IoT is turning into a noteworthy problem.Efficient outside honesty confirmation is an imperative piece of information security.We give a major picture through condensing and examination of the primary aftereffects of outer uprightness check plans for huge information in cloud. AbstractAs distributed computing is by and large broadly embraced for enormous information handling, information security is getting to be one of the real worries of information proprietors. Information honesty is a critical factor in any information and calculation related setting. It isn't just a single of the characteristics of administration, yet additionally a vital piece of information security and protection.

**Ora, P., and Pal, P. R., (2015) [6],** With the consistent headway in specialized field numerous advances are developing step by step, distributed computing is one of them. With the assistance of distributed computing client can without much of a stretch offer, store and recover their information from anyplace. Distributed computing gives equipment, programming and infrastructural stockpiling to numerous clients at once. The same number of clients share their information on a cloud the primary inquiry is about security of information display on cloud. In this exploration paper arrangement is given to keep up information security and information honesty.

**Yu, Y., Au, M. H., Mu, Y., Tang, S., Ren, J., Susilo, W., and Dong, L., (2015) [7],** Remote information uprightness checking (RDIC) empowers a server to demonstrate to an evaluator the trustworthiness of a put away document. It is a valuable innovation for remote stockpiling, for example, distributed storage. The examiner could be a gathering other than the information proprietor; henceforth, a RDIC confirmation is construct more often than not with respect to freely accessible data. To catch the need of information security against an untrusted examiner, Hao et al. formally characterized protection against outsider verifiers as one of the security prerequisites and proposed a convention fulfilling this definition. Notwithstanding, they watch that every current convention with open certainty supporting information refresh, including Hao et al.s proposition, require the information proprietor to distribute some meta-information identified with the put away information.

**Jiang, T., Chen, X., and Ma, J., (2016) [8],** The approach of the distributed computing makes stockpiling outsourcing turn into a rising pattern, which advances the safe remote information evaluating a hotly debated issue that showed up in the exploration writing. As of late some examination think about the issue of secure and proficient open information respectability inspecting for shared dynamic information. Be that as it may, these plans are as yet not secure against the conspiracy of distributed storage server and disavowed gather clients amid client repudiation in down to earth distributed storage framework. In this paper, they make sense of the intrigue assault in the leaving plan and give a productive open trustworthiness examining plan with secure gathering client denial in light of vector duty and verifier-nearby disavowal assemble signature.

## III. Result and Analysis

The Diffie-Hellman Key Transactions is one of the extra accepted and interestingmethods of key distribution. It is a public-key cryptographic systemwhose merely intention is for allocating keys. Diffie-Hellman is an example of aKey Association 2public-key allocation scheme (PKDS) whereby it is utilized to transactions a singlepiece of data, and whereas the worth obtained is normally utilized as asession key for a private-key scheme.

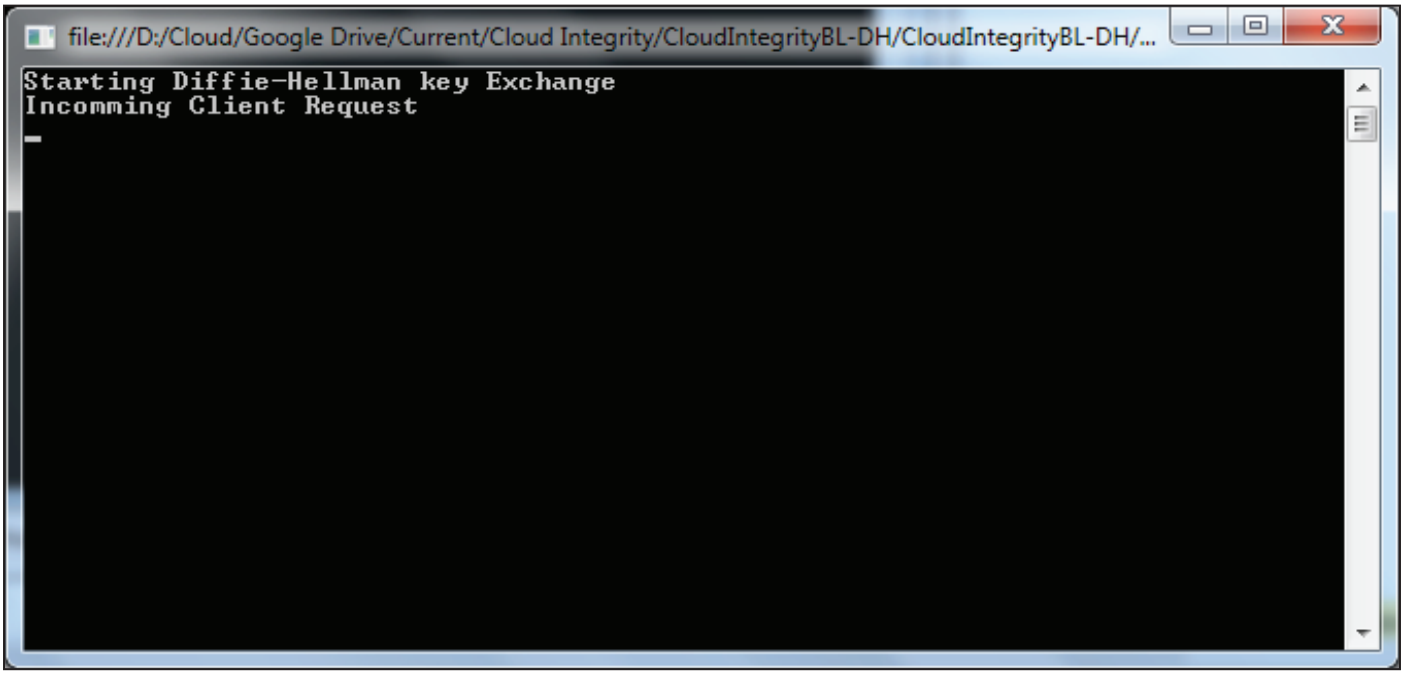


Fig. 1: Starting Diffie-Hellman key Exchange mechanism for Cloud key Exchanges, Client Generates a request to Server

*BiLinearDiffieHellman* request = *newBiLinearDiffieHellman*(32).GenerateRequest();
The intention of the algorithm is to enable two users to securely transactions a key that can next be utilized forSubsequent encryption of messages. The algorithm itself is manipulated to the transactions of hidden values. The Diffie-Hellman algorithm depends for its effectiveness on the difficulty of computing discrete logarithms.

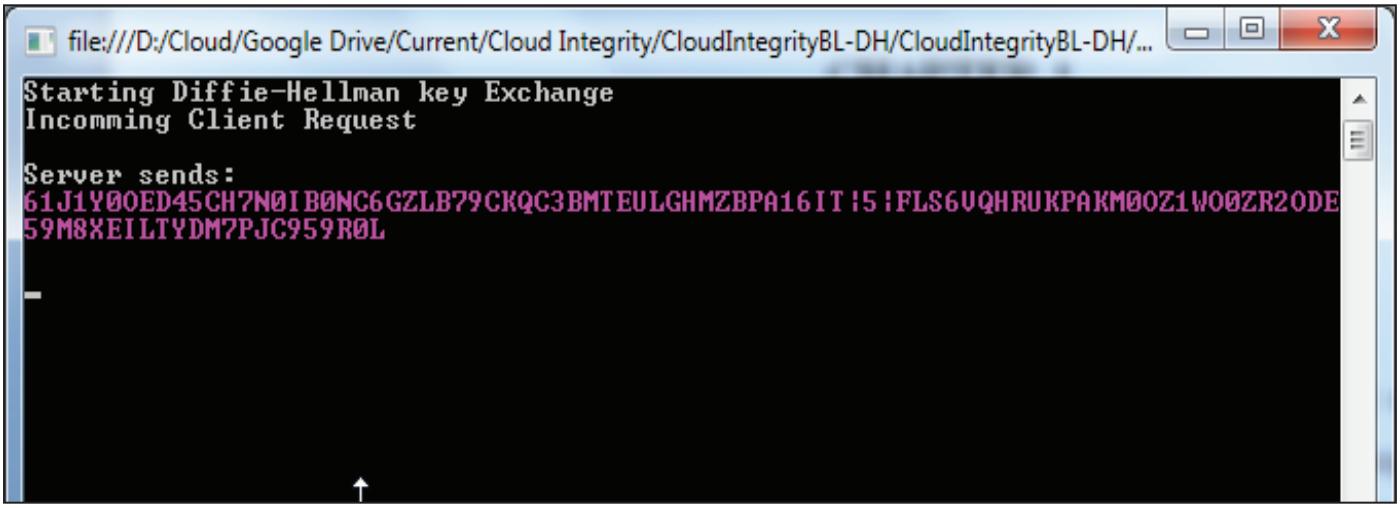

Fig. 2: Server Sends Authentication Keys to the Client

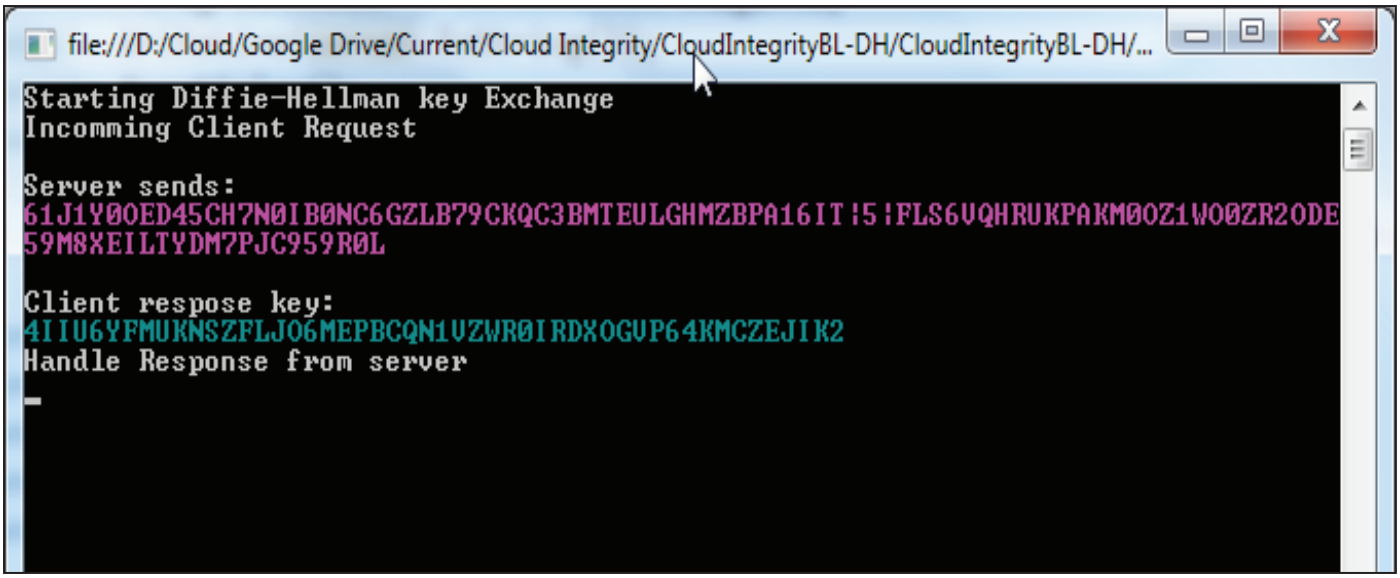

Fig. 3: Client Handles the Response from the Server

Fig. 5: Client Generates a Response Key and Forwards Encrypted Data to the Server

Above figures display a easy protocol that makes use of the Bilinear Diffie-Hellman calculation and exchange. Presume that user A wishes to set up a connection alongside user B and use a hidden key to encrypt memos on that connection. User A can produce a one-time confidential key XA, compute YA, and dispatch that to user B. User B replies by producing a confidential worth XB computing YB, and dispatching YB to user A. Both users can nowadays compute the key. The vital area benefits q and α should demand to be recognized in front of time. Alternatively, user A might select benefits for q and α and contain those in the early memo.

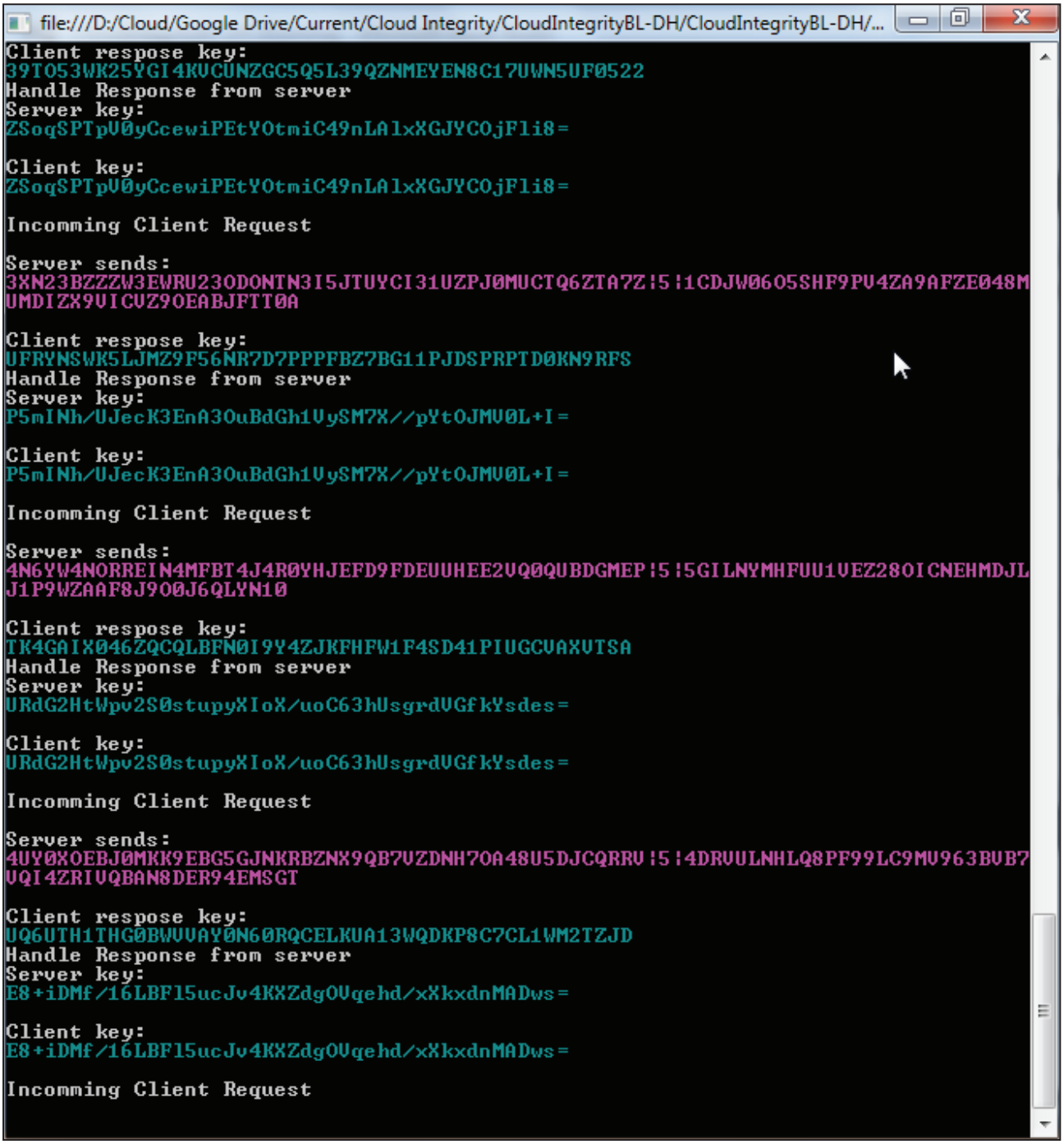


Fig. 5: The Process is Executed for Several Client Request

### A. Algorithm Parameters

| Parameter Name | Parameter Value |
|---|---|
| Key Size | 256-512 bytes |
| Key Storage and backend | XML |
| Verification Iterations | 50-1000 |
| TPA Verification | 2,50,000-1,000,000 |

In this period Homomorphic Client Key produce method is increased to produce area key and confidential key. XML is utilized for bypassing data to cloud Homomorphic authenticators and alongside alongside meta data such as Padding. The Encryption method seizes two arguments namely hidden key and file. The file gratified is tear into blocks. Next signature is computed for every single block. Every single block's hash program is seized and two nodes' hash is merged into one in order to produce the subsequent node.

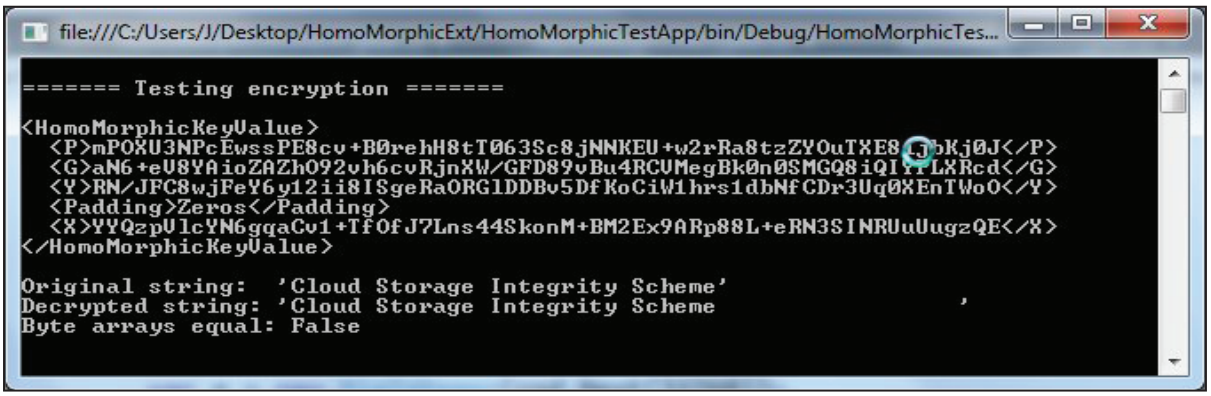


Fig 5.6: Testing Homomorphic Schemes, in this step the Algorithm creates, public key <P> and Client keys <G>, <Y>, <P>, <X> respectively

### B. Cloud Integration Verification

The content of outsourced data can be confirmed by whichever client or TPA. This is completed by invigorating server by providing a little file and block randomly. Up on the examination, the cloud storage server computes the origin hash program for the consented file and blocks and next returns the computed origin hash program and early stored hash program alongside alongside signature. Next the TPA or client uses area key and confidential key in order to decrypt the gratified and difference the origin hash program alongside the origin hash program returned by clients.

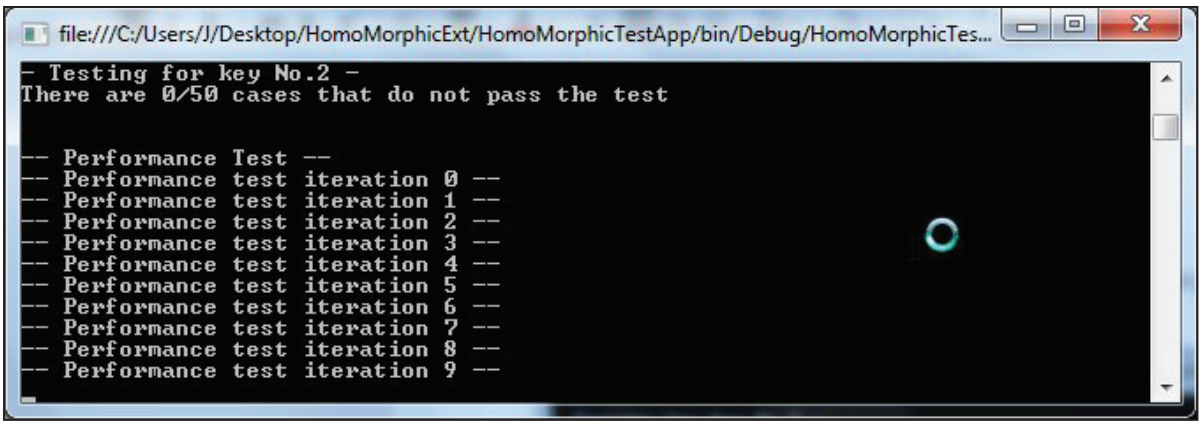


Fig. 7: Performance Test of the proposed Scheme using 50 iterations

The content of outsourced data can be confirmed by whichever client or TPA. This is completed by challenging server by providing a little file and chunk randomly. Up on the examination, the cloud storage server computes the origin hash program for the given file and blocks and next returns the computed origin hash program and primarily deposited hash program alongside alongside signature. Next the TPA or client uses area key and confidential key in order to decrypt the content and match the origin hash program alongside the origin hash program returned by clients.

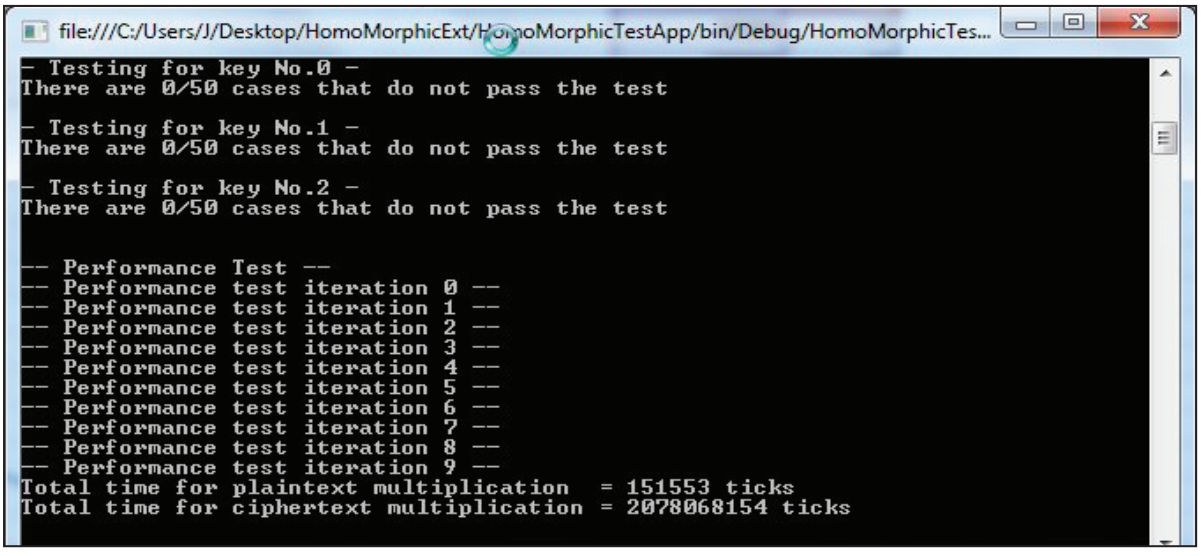


Fig. 8: Performance Test of the proposed Scheme using 50 iterations with 200,000 Request

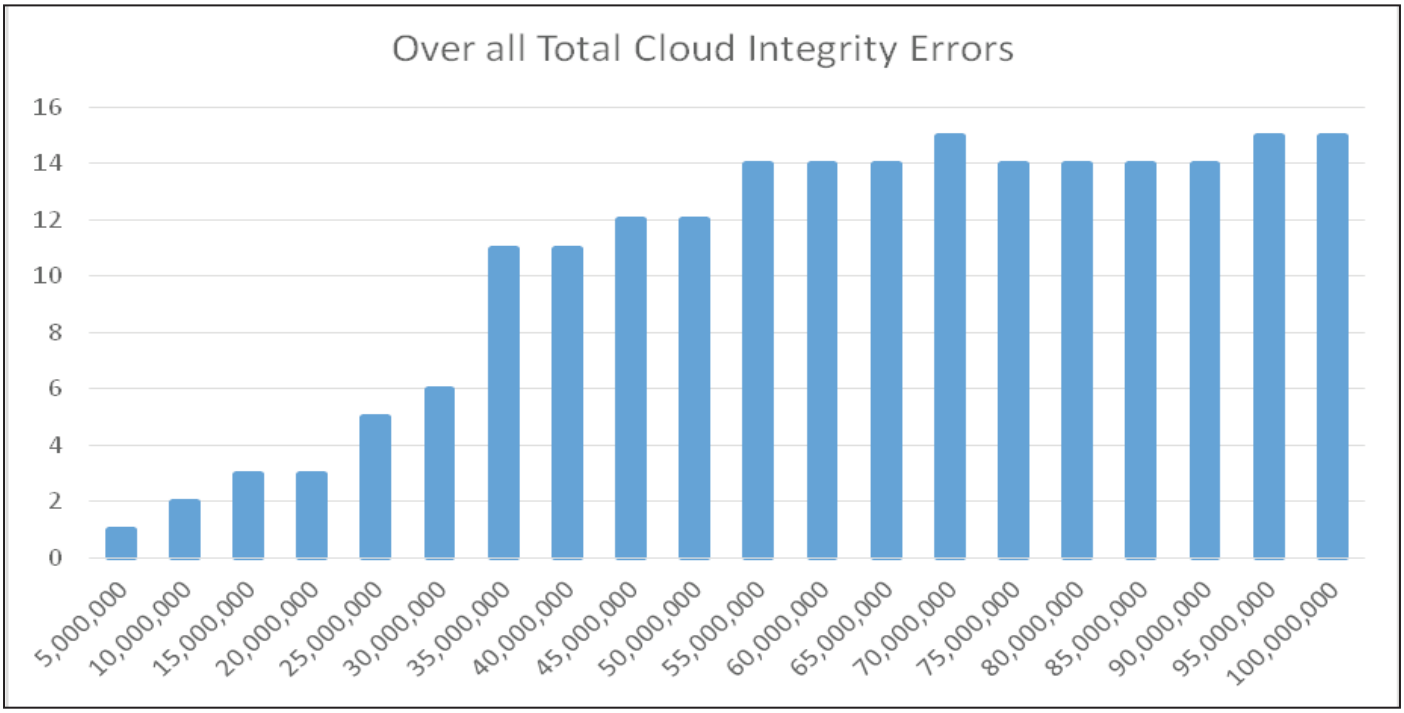


Fig. 9: Total Integrity Errors Given the Iterations

It is obvious from above graph and table that proposed respectability conspire produces least mistakes by and large under 1 Integrity error for every 1 Million Requests.

| 10000 Iterations | 20000 Iterations | 30000 Iterations | 40000 Iterations |
|---|---|---|---|
| 11904946 | 19589798 | 30251343 | 40912888 |
| 12810879 | 21456636 | 28190187 | 34923738 |
| 11394205 | 21089223 | 28773522 | 36457821 |
| 12909182 | 19055739 | 29145317 | 39234895 |
| 14417366 | 19105216 | 29394454 | 39683692 |
| 16351427 | 20680839 | 28208511 | 35736183 |
| 17557635 | 21177097 | 27972217 | 34767337 |
| 12775411 | 21858291 | 28519641 | 35180991 |
| 17141168 | 22141427 | 30349116 | 38556805 |
| 19871818 | 24058961 | 31432040 | 39805119 |

Table 5.1: Total Execution time taken for Cloud Key Generation and Evaluating the Checking Integrity.

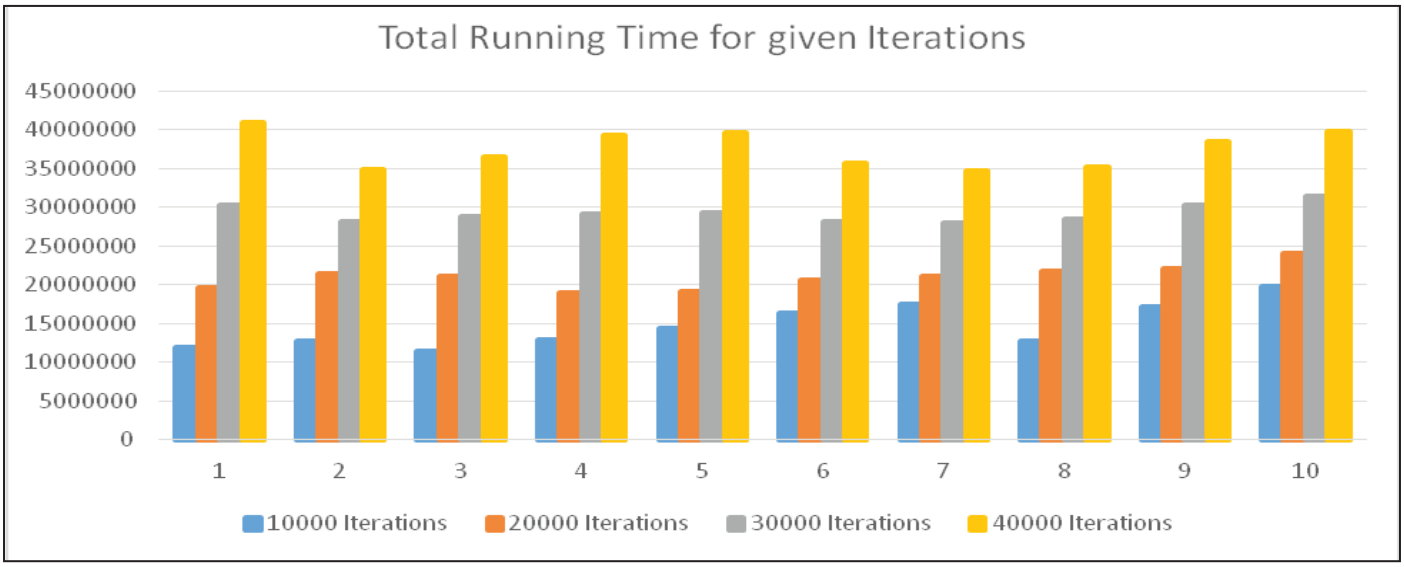


Fig. 10: Total Execution Time for Cloud Key Generation and Integrity Checks per request.

In this work, we have gave the design and implementation of Homographic scheme for verifying the integrity of multi-tenant cloud infrastructures. we have worked to enable the client in becoming a facts of integrity of the data that he wishes to store in the cloud storage servers alongside bare minimum prices and efforts. Our scheme was industrialized to cut the computational and storage overhead of the client as well as to minimize the computational overhead of the cloud storage server. Our Scheme produces minimum errors on average less than 1 error each Million Appeal.

Table 5.2: Comparison of Throughput of various Integrity Proofs Checks

| Payload Data | Throughput (Mb/Sec) | | | |
|---|---|---|---|---|
| | 3DES | **DES** | **AES** | **Homomorphic** |
| 10 | 20 | 40 | 60 | 100 |
| 20 | 50 | 70 | 90 | 180 |
| 30 | 70 | 90 | 100 | 260 |
| 40 | 90 | 110 | 130 | 320 |
| 50 | 100 | 130 | 180 | 440 |
| 60 | 130 | 150 | 270 | 550 |
| 70 | 160 | 200 | 380 | 700 |
| 80 | 220 | 300 | 522 | 800 |
| 90 | 290 | 400 | 600 | 900 |

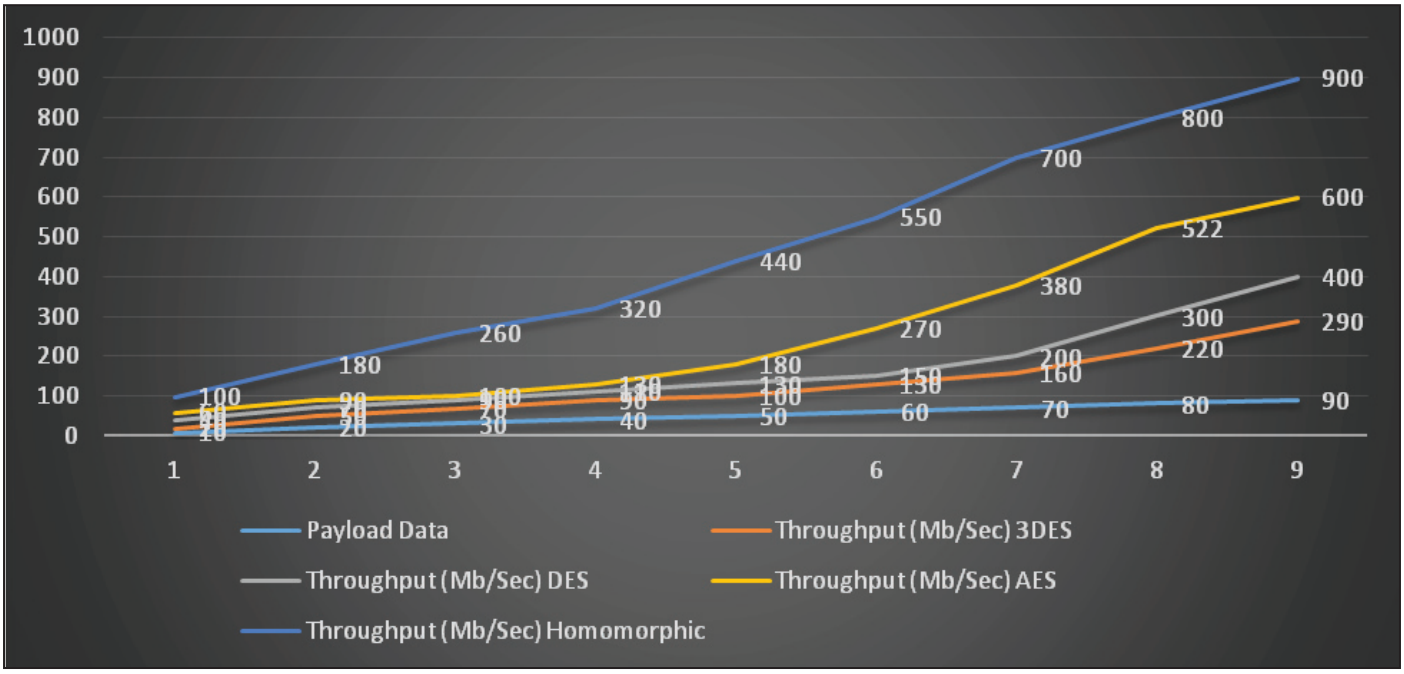


Fig. 11: Comparison of Throughput of Various Integrity Proofs Checks

Table 3: Comparative Analysis of Various Integrity Checking Algorithms

| **Protocol Used** | **Minimum (ms)** | **Avg (ms)** | **Maximum (ms)** |
|---|---|---|---|
| Homomorphic Encryption Time | 0.051 | 0.093 | 0.177 |
| Homomorphic Decryption Time | 0.048 | 0.109 | 0.238 |
| RSA Encryption Time | 0.056 | 0.11 | 0.277 |
| RSA Decryption Time | 0.055 | 0.111 | 0.317 |
| DSA Encryption Time | 0.058 | 0.128 | 0.388 |
| DSA Decryption Time | 0.06 | 0.187 | 0.469 |

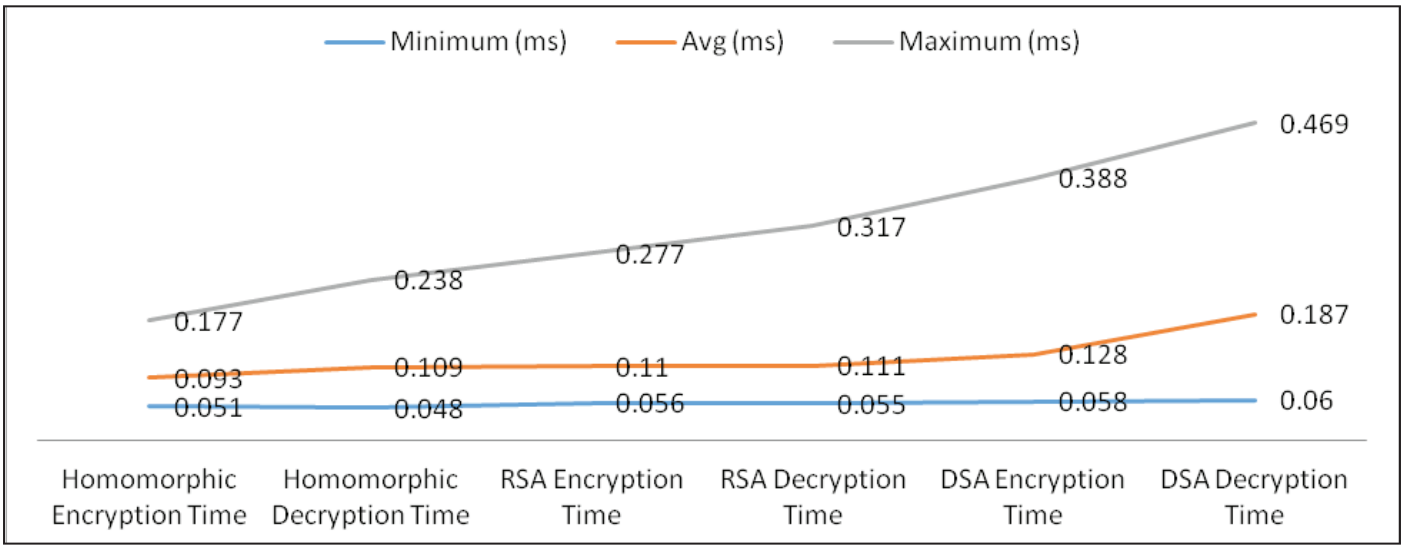


Fig. 12: Comparative Analysis of Various Integrity Checking Algorithms

## VI. Conclusion

In cloud, there are countless gains to relief the burden of data association for users, such as facile to admission, inexpensive storage space, convenient resource-sharing. After users ponder their own manipulated storage space, they yearn to relish the convenient colossal storage space ability in cloud. Users usually upload data to the cloud storage servers, next delete the innate copies. So users capitulated finished manipulation above the data itself.

After users confided their data to cloud storage, the most concern setback is that the cloud ability provider (CPS) could delete users' data or tamper alongside the users' data maliciously. For the sake of hobbies, the CPS have countless motivations to flounder the obligation of protecting the user data. Such as CPSs in order to save their own storage space and save working expenses, CPS delete data that users admission few; contraption obligation lead to defeat of data, CPS obscure data defeat incident; unintentionally delete user data after transfer data to new storage servers. Protection is mainly distressed alongside imposing good deeds and halting improper behaviour.

We counsel a safeguard data integrity checking decentralized erasure program cloud storage arrangement established on the Elative allocate scheme. Our storage arrangement that consists of storage servers and key servers can completely stop malicious servers from robbing our data that are partly decrypted. The counseled scheme can promise confidentiality of memos for a long era of time. We have utilized a homomorphic encryption scheme that permits increasing plaintext hidden inside of ciphertexts and after employing the homomorphic property joined alongside endeavored and tested "threshold cryptosystem", if in order to decrypt an encrypted memo, countless parties (more than a little threshold number) have to cooperate in the decryption.

### B. Future Works

There are a number of interesting Homomorphic variants to discover in upcoming up work. The protocols we have delineated above for Homomorphic merely furnish assurance for static files. We are investigating in present work design of comparable protocols that accommodate file updates. We trust that the Homomorphic methods we have gave in this work aid pave the method for priceless ways to Cloud file arrangement potential and Integrity. We can additionally work on:

- Applications of fully homomorphic encryption IaaS
- Environments Non-malleability and homomorphic encryption
- Fully homomorphic encryption and functional decryption

## References

[1] Yubin Yang; Hui Lin; Jixi Jiang,"Cloud analysis by modeling the integration of heterogeneous satellite data and imaging",IEEE,Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 2006

[2] Kaewpuang, R.; Uthayopas, P.; Srimool, G.; Pichitlamkhen, J.,"Building a Service Oriented Cloud Computing Infrastructure Using Microsoft CCR/DSS System",IEEE,Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on, 2009

[3] Tao Wu; Kun Qin,"Inducing Uncertain Decision Tree via Cloud Model",IEEE,Semantics, Knowledge and Grid, 2009. SKG 2009. Fifth International Conference on, 2009

[4] Yi Zhao; WenlongHuang,"Adaptive Distributed Load Balancing Algorithm Based on Live Migration of Virtual Machines in Cloud",IEEE,INC, IMS and IDC, 2009. NCM '09. Fifth International Joint Conference on, 2009

[5] Liu, C., Yang, C., Zhang, X., & Chen, J. (2015). External integrity verification for outsourced big data in cloud and IoT: A big picture. Future Generation Computer Systems, 49, 58-67.

[6] Ora, P., & Pal, P. R. (2015, September). Data security and integrity in cloud computing based on RSA partial homomorphic and MD5 cryptography. In Computer, Communication and Control (IC4), 2015 International Conference on (pp. 1-6). IEEE.

[7] Yu, Y., Au, M. H., Mu, Y., Tang, S., Ren, J., Susilo, W., & Dong, L. (2015). Enhanced privacy of a remote data integrity-checking protocol for secure cloud storage. International Journal of Information Security, 14(4), 307-318.

[8] Jiang, T., Chen, X., & Ma, J. (2016). Public integrity auditing for shared dynamic cloud data with group user revocation. IEEE Transactions on Computers, 65(8), 2363-2373.

[9] Khan, M. (2016, December). Cross layer design approach for Congestion Control in MANETs. In Advances in Electronics, Communication and Computer Technology (ICAECCT), 2016 IEEE International Conference on (pp. 464-468). IEEE.

[10] Ren, Y., Shen, J., Zheng, Y., Wang, J., & Chao, H. C. (2016). Efficient data integrity auditing for storage security in mobile health cloud. Peer-to-Peer Networking and Applications, 9(5), 854-863.
[11] Rostami, A., Cheng, B., Bansal, G., Sj÷berg, K., Gruteser, M., & Kenney, J. B. (2016). Stability challenges and enhancements for vehicular channel congestion control approaches. IEEE Transactions on Intelligent Transportation Systems, 17(10), 2935-2948.
[12] Sirajuddin, M. D., Rupa, C., & Prasad, A. (2016). Advanced Congestion Control Techniques for MANET. In Information Systems Design and Intelligent Applications (pp. 271-279). Springer, New Delhi.
[13] Tchernykh, A., Schwiegelsohn, U., Talbi, E. G., & Babenko, M. (2016). Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability. Journal of Computational Science.
[14] Wang, H., He, D., & Tang, S. (2016). Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud. IEEE Transactions on Information Forensics and Security, 11(6), 1165-1176.
[15] Yu, Y., Xue, L., Au, M. H., Susilo, W., Ni, J., Zhang, Y., ... & Shen, J. (2016). Cloud data integrity checking with an identity-based auditing mechanism from RSA. Future Generation Computer Systems, 62, 85-91.
[16] Agrawal, R., Sharma, P., & Malviya, V. (2017, February). A novel method for queqe managment using RED technique in mobile ad hoc network. In I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2017 International Conference on (pp. 175-180). IEEE.
[17] Akhtar, N., Khattak, M. A. K., Ullah, A., & Javed, M. Y. (2017, December). Efficient Routing Strategy for Congestion Avoidance in MANETs. In Frontiers of Information Technology (FIT), 2017 International Conference on (pp. 305-309). IEEE.