BioGecko

Vol 12 Issue 03 2023
ISSN NO: 2230-5807

Virtual Machine Migration For Green Cloud Computing

Karre Sai Teja Goud[1]
Computer Science and Engineering
Bharath Institute of Higher Education and Research, Chennai, Tamilnadu, India

Y. Ganesh varaprasad[2]
Computer Science and Engineering
Bharath Institute of Higher Education and Research,
Chennai, Tamilnadu, India
N. Sanjay Kumar[3]
Computer Science and Engineering
Bharath Institute of Higher Education and Research, Chennai, Tamilnadu, India

S. Nirmal sam [4]*
Computer Science and Engineering
Bharath Institute of Higher Education and Research,
Chennai, Tamilnadu, India
nirmalsam.cse@bharathuniv.ac.in

***Abstract*—** Cloud computing technology has developed and unstructured data has exploded, as a result of which cloud storage technology has gained attention and improved development. Information and cloud data stored in the cloud are not relegated to the cloud provider's control. The majority of privacy protection schemes use encryption technology. Data in the cloud can be protected through a variety of privacy preserving methods. Our framework is based on fog computing and consists of three layers. Using the proposed framework, cloud storage can be fully utilized while protecting data privacy. The data is segmented in this case using the Hash-Solomon method. Data information is lost if one of the data parts is missing. Our scheme can demonstrate security and efficiency by securing data information with bucket concept algorithms in this framework. The algorithm also uses computational intelligence to calculate the distribution ratios are stored in the cloud, fog, and locally stored machines. Software as a Service is hosted version of the client's application was released that is accessible by multiple clients via a network. A possible exception to this is limited application configuration settings that are specific to the user. These settings do not require any management or control on the part of the client.

Keywords*— *Fog computing, Quality of Service(QoS) , AAL scenario, TSL framework, Access Control Lists (ACLs).

I. Introduction

Cloud computing is a term used in computer science to describe a sort of outsourcing of computer services, much like how energy is outsourced [1]. It is easy for users to utilize. They don't have to be concerned about how or from where the electricity is created. They pay for their consumption each month. A similar concept underlies cloud computing: The user doesn't have to consider how processing power, storage, or specially created development environments work; they can just access them [2]. Typically, cloud computing uses the internet. The cloud is an abstraction that hides the complex internet architecture because using computer network diagrams as a model, it serves as a metaphor for the internet [3].

The term "fog computing," which alludes to the increasing challenges in obtaining objective information, can be applied to both huge data structures and enormous cloud systems [4]. The stuff that is obtained as a result is of poor quality. Cloud and big data systems may experience a variety of repercussions from fog computing. Inaccurate content distribution is a problem that has been addressed by the development of measurements that aim to increase

accuracy, but one aspect that can be extracted consistently is this problem [5]. For networking, there are two planes, one for data and one for control. When applied to the data plane, fog computing allows computing services to be located at the network's edge rather than on servers [6]. Fog computing, as opposed to cloud computing, is more focused on close offers superior quality of service (QoS) and edge analytics/stream mining by being close to end users and client goals, reducing latency, and reducing backbone bandwidth, resulting in a superior user experience and redundancy in case of a failure, as well as applying to AAL scenarios [7]. To ensure user privacy, we propose a TLS architecture based on the fog computing idea. The TSL framework can successfully safeguard users' privacy while granting them some degree of management power [8]. The inside attack is challenging to fend off, as was already mentioned. Traditional methods are effective in addressing external attacks, but they are useless when CSP is experiencing issues. In contrast to conventional methods, our system uses encoding technology to split user input into three segments of varying sizes [9]. Each of them will be missing some essential information necessary for secrecy. The three data components will be saved on the cloud server, the fog server, and the user's local workstation in that order, starting with the biggest, in accordance with the fog computing concept [10]. Even if the attacker collects every bit of data from the user from a single server, he will not be able to restore it using this method. For the CSP, since both the local machine and the fog server are user-controlled, they are also unable to obtain any relevant information without the data saved on those two devices [11].

II. METHODOLOGY

A. Login Module

When a person opens the website, they are presented with this. A valid phone number and a password that was set up during registration are required for users to log onto the website [12]. The user will be able to successfully visit the website if their information corresponds to the data in the database table; otherwise, a message indicating that the login attempt failed is displayed, and the user must enter their information correctly again. New user registration is also made possible via a link to the register action [13].

B. Registration Module

Before logging in, a new user must to access the website, you must first register there. The registered activity becomes open when the register button is clicked in the login activity. As part of the registration process, a new user must provide their full name, password, and phone number [14]. In the text box labeled "Confirm Password," the user must reenter the password. The user is taken back to the login screen after entering the necessary information in all text boxes and clicking the register button to save it to the database. After that, to access the website, a registered user must log in. To ensure that the website functions properly, validations are applied to each textbox. The text boxes for name, contact information, password, and confirm password must all have information in them in order for you to register; therefore, they cannot be left blank. An alert will be displayed by the app if any of these text boxes are empty [15]. For registration to be successful, the data in the confirm password and password field must also match. A contact number that is legitimate and has 10 digits is another requirement. The user must re-register if any of these validations are broken, which will result in registration failure. When a field is empty, the website will display a message. Users will be sent to login activity for webpage login if all such information is accurate.

C. Storage Module

Users of this module have three separate storage servers where they can store their stuff. Data that has been uploaded to the cloud is no longer within the data owner's control [16]. The original data are encrypted into three layers in this module. Before being stored in the cloud, each layer of data can be encrypted using a distinct cryptographic technique and encryption key.

D. Recovery Module

In this module, users can utilize one of three file recovery techniques to get their data back from one of three different storage servers: a local system, a Fog server, or a Cloud server. The Hash-Solomon code algorithm is being used here because it may be used to separate data into multiple pieces. Data information is lost if one data component is absent. In this section, we use a range of experiments, including encoding, decoding, and testing of various data volumes, to assess the functionality and sustainability of the fog computing-based TLS framework.

Privacy is protected by a three-layer cloud storage architecture that employs computational intelligence and fog computing. Our main concern is the protection of privacy; we are not concerned about current attacks. Three different types of data pieces are stored in the three-layer cloud storage. Data information is lost if one data component is absent. Using the algorithms in this suggested framework that are based on the bucket notion. The BCH code is the algorithm we've chosen. It is quite adaptable.

Numerous security issues are also brought on by cloud storage. Users lose control over how their data is physically stored when they use cloud storage, which separates data ownership and administration. We develop a Hash-Solomon algorithm and suggest a fog computing model-based TLS framework was created to address the issue with cloud storage privacy security. The approach is demonstrated to be practical through the theoretical safety analysis. By evenly dispersing the amount of data blocks kept across several servers, we can make sure that each server's data is private. On the other side, the encoding matrix cannot be broken theoretically. Additionally, hash transformation might shield incomplete information.

Data structure software architecture, procedural specifics, algorithm, etc., as well as the interface between modules are the center of a multi-step design process. Before any coding is done, the requirements are transformed into a display of software that may undergo quality assurance testing during the design process. As new techniques, greater analysis, and a better knowledge of borders emerged, computer software design changed regularly. The revolution in software design is still in its early stages. The depth, flexibility, and quantitative nature typically associated with more traditional engineering disciplines are not present in software design methodology. Although there are still methods for creating software designs, there are criteria for desirable design traits that can be used, as well as design notation.

III. ALGORITHMS

A. Bucket

The buckets' Access Control Lists (ACLs) in Google Cloud Storage are represented via the Bucket Access Controls resource. You may control who gets access to your data and to what degree using ACLs. The three separate types of data pieces are stored in the three-layer cloud storage. Buckets within Google Cloud Storage's Access Control Lists (ACLs) if one data component is missing, the data information is lost.

B. BCH code algorithm

The Hamming code for multiple-error correction has been impressively generalized by the Bose–Chaudhuri–Hocquenghem codes (BCH codes), which comprise a significant class of efficient random error-correcting cyclic codes. In this lecture note, we just take into account binary BCH codes. There will be discussion of non-binary BCH codes such Reed-Solomon codes.

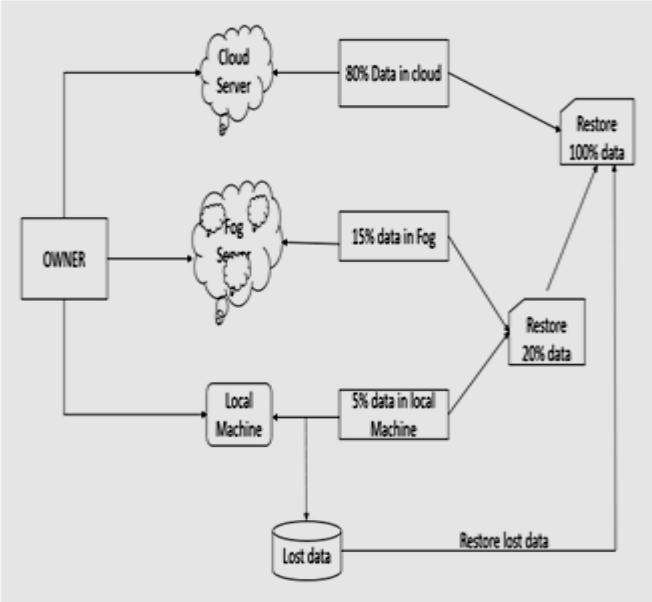

Figure 1: Fog Server data processing

In above figure 1, Fog server's data processing and storage capacity are fully utilized by TLS framework. The local machine, fog server, and three layers of the architecture are all present. According to users' allocation strategies, each server keeps a particular volume of data; the storage percentage is set by this. First, the user's local PC will encrypt their data. Afterward, permit the machine to save 1% of the encoded data. Then send 99% of the remaining data to the fog server. Second, we use the data arriving from the user's PC on the fog server to carry out similar actions. Only 4% of the data will remain on the fog server, with the remaining 96% being sent to the cloud server. Hash-Solomon coding is utilized in the aforementioned operations. The fog server then receives 99.9% of the data. Then, using the data that comes from the user's computer, we do similar operations on the fog server. The fog server will retain about 4% of the data, while the cloud server will receive the remaining 96%. Creating m redundant pieces of data. In these k+m parts of data, the hash-Solomon code has the property that if someone obtains at least k parts, he can retrieve the entire data. To put it another way, no one can retrieve the entire set of data from less than k components. In our system, we limit the amount of data that may be saved in the higher server, which has a larger storage capacity, to k-1 parts, and allow the remaining data to be stored in the lower server in accordance with this property of the Hash-Solomon code. In this method, even if data from one of the three layers was taken, the stealer cannot recover the entire set of data. We can protect user data privacy in this way.

IV. RESULTS AND DISCUSSION

Design is a multi-step process that concentrates on the interface between modules, procedural details, algorithm, and data structure software architecture. Before any coding is done, the requirements are also translated into a presentation of software that can be tested for quality during the design phase. As new techniques, improved analysis, and a better grasp of borderlines emerged, computer program design is always changing. The revolution in software design is only just beginning. Software design technique hence lacks the breadth, adaptability, and quantitative nature typically associated with more traditional engineering disciplines. Although there are still methods for creating software designs, criteria for desirable design traits are available, and design notation can be used.

A. Existing System

The technology of cloud computing has advanced recently. The rapid expansion of unstructured data has increased interest in and advancements in cloud storage technology. The advancement of computer technology has been rapid. Cloud computing has developed throughout time thanks to the efforts of numerous people. The user's data is entirely saved on cloud servers according to the current storage format. If a user's right to data control is compromised, their privacy may be at stake. Most methods for protecting personal information rely on encryption technology. These techniques are ineffective against attacks coming from a cloud server's inside.

Disadvantages

Changes in the understanding of risk as a result of extending the datacentre into the cloud. Low latency and location awareness.

B. Proposed System

The technology of cloud computing has advanced recently. The rapid expansion of unstructured data has increased interest in and advancements in cloud storage technology. The advancement of computer technology has been rapid. Cloud computing has developed throughout time thanks to the efforts of numerous people. The user's data is entirely saved on cloud servers according to the current storage format. If a user's right to data control is compromised, their privacy may be at stake. The majority of encryption technology is used in privacy protection techniques. These techniques are ineffective against attacks coming from a cloud server's inside.

Advantages:

To reduce data wastage and process times, our system employs the bucket concept. BCH (Bose-Chaudhuri-Hocquenghem) code is the algorithm we are utilizing. Extremely flexible. BCH codes have low levels of redundancy and are employed in many communications applications.

V. CONCLUSIONS

There are many advantages to the growth of cloud computing. Users can increase their storage capacity with the use of the practical technology known as cloud storage. However, cloud storage also contributes to a number of security issues. Users that use cloud storage experience due to the fact that they have no control over the data's actual physical storage, ownership and management of the data must be separated. BCH Code algorithm and TLS framework are used to solve the problem of privacy protection in cloud storage. The scheme's viability is demonstrated by the theoretical safety study. We can make sure that each server's data is kept private by distributing the number of data blocks stored on other servers fairly. Additionally, the fragmentary information can be protected by applying hash transformation. This approach successfully completed while testing encoding and decoding, the experiment had no impact on how well cloud storage worked. To further attain maximum efficiency, we construct an acceptable comprehensive efficiency measure. We also find that the Cauchy matrix performs the coding procedure better.

Acknowledgment

The encouragement and support from Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India is gratefully acknowledged for providing the laboratory facilities to carry out the research work."

References

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," Nat.Inst. Stand. Technol., vol. 53, no. 6, pp. 50–50, 2009.

[2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," Wireless Commun. Mobile Comput., vol. 13, no. 18, pp. 1587–1611, 2013.

[3] J. Chase, R. Kaewpuang, W. Yonggang, and D. Niyato, "Joint virtual machine and bandwidth allocation in software defined network (sdn) and cloud computing environments," in Proc. IEEE Int. Conf. Commun., 2014, pp. 2969–2974.
[4] H. Li, W. Sun, F. Li, and B. Wang, "Secure and privacy-preserving data storage service in public cloud," J. Comput. Res. Develop., vol. 51, no. 7,pp. 1397–1409, 2014.
[5] Y. Li, T.Wang, G.Wang, J. Liang, and H. Chen, "Efficient data collection in sensor-cloud system with multiple mobile sinks," in Proc. Adv. Serv. Comput., 10th Asia-Pac. Serv. Comput. Conf., 2016, pp. 130–143.
[6] L. Xiao, Q. Li, and J. Liu, "Survey on secure cloud storage," J. Data Acquis. Process., vol. 31, no. 3, pp. 464–472, 2016.
[7] R. J. McEliece and D. V. Sarwate, "On sharing secrets and reed-solomon codes," Commun. ACM, vol. 24, no. 9, pp. 583–584, 1981.
[8] J. S. Plank, "T1: Erasure codes for storage applications," in Proc. 4th USENIX Conf. File Storage Technol., 2005, pp. 1–74.
[9] R. Kulkarni, A. Forster, and G. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey," IEEE Commun. Surv. Tuts., vol. 13, no. 1, pp. 68–96, First Quarter 2011.
[10] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacypreserving and copy-deterrence content-based image retrieval scheme in cloud computing," IEEE Trans. Inf. Forensics Security, vol. 11, no. 11, pp. 2594–2608, Nov. 2016.
[11] J. Shen, D. Liu, J. Shen, Q. Liu, and X. Sun, "A secure cloud-assisted urban data sharing framework for ubiquitous-cities," Pervasive Mobile Comput., vol. 41, pp. 219–230, 2017.
[12] Z. Fu, F. Huang, K. Ren, J.Weng, and C.Wang, "Privacy-preserving smart semantic search based on conceptual graphs over encrypted outsourced data," IEEE Trans. Inf. Forensics Security, vol. 12, no. 8, pp. 1874–1884, Aug. 2017.
[13] J. Hou, C. Piao, and T. Fan, "Privacy preservation cloud storage architecture research," J. Hebei Acad. Sci., vol. 30, no. 2, pp. 45–48, 2013.
[14] Cynthia, J., Sankari, M., Suguna, M., & kumar, D. R. (2018, December). Survey on Disaster Management using VANET. 2018 4th International Conference on Computing Communication and Automation (ICCCA). 2018 4th International Conference on Computing Communication and Automation (ICCCA). https://doi.org/10.1109/ccaa.2018.8777331.
[15] Arvindhan, M., Rajesh Kumar, D. (2022). Analysis of Load Balancing Detection Methods Using Hidden Markov Model for Secured Cloud Computing Environment. In: Deepak, B.B.V.L., Parhi, D., Biswal, B., Jena, P.C. (eds) Applications of Computational Methods in Manufacturing and Product Design. Lecture Notes in Mechanical Engineering. Springer, Singapore. https://doi.org/10.1007/978-981-19-0296-3_53.
[16] Arvindhan, M., & Dhanaraj, R. K. (2022). The firefly technique with courtship training optimized for load balancing independent parallel computer task scheduling in cloud computing. International Journal of Health Sciences, 6(S1), 8740–8751. https://doi.org/10.53730/ijhs.v6nS1.6999.