Feasibility study of Hide Data in Images in Cloud Environment

[1]Harsh Bhor
[1]K J Somaiya Institute of Engineering & IT, Sion, Mumbai, Maharashtra, India.
[1]hbhor@somaiya.edu

Abstract : Cloud computing is rapidly emerging as a technology trend that almost every industry, transmits information, such as pictures, videos, and text, can be transmitted rapidly on the internet. The MapReduce programming model can be used to process large-scale data sets in cloud environments. In this paper, we use the Hadoop system to build the cloud computing environment. By using data hiding technology to embed data into cover images. Thus, cloud computing provides a convenient platform and also decreases cost of the equipment required for processing large data.

***Keywords* — Hadoop, Map Reduce model, Data Hiding, Cloud environments.**

I. INTRODUCTION

Data hiding is a term encompassing a wide range of applications for embedding messages in content. Inevitably, hiding information destroys the host image even though the distortion introduced by hiding is imperceptible to the human visual system. We are living in the times of information technology, and information, such as pictures, videos, and text, can be transmitted rapidly on the Internet. By using a browser, people have access to information worldwide, and distance is no longer a limitation. However, in transmitting such information, the security of digital data has become an important issue. Thus, many studies have focused on techniques for protecting digital information from being stolen.

To protect digital information, the field of steganography, which develops various means for embedding confidential information in other intermediary media, has evolved to enhance the security of information during the process of transmission. The intermediary media can be other digital data, such as digital images, digital audio, and videos. The goal of data protection is achieved when data can be transmitted without people who might intercept the data being able to recognize that the intermediary media are embedded with confidential information.

The essence of the concept of cloud computing [7] is network computing. Through network connections, multiple machines are linked together to provide services to users. However, the users don't know where the order is sent, and, in fact, they don't need to know. Users only need to confirm that the result they received is correct. Traditional distributed and parallel computing technology requires additional workload. Computer programmers must do the additional work that is generated by using distributed and parallel computing, e.g., coordinating and allocating the tasks of the machines, synchronizing data, sharing resources, and detecting errors in tasks. In addition, it is more complicated to develop distributed and parallel computing applications than general applications, so the time required for computer programmers to develop software is increased.

II. HADOOP SYSTEM

Hadoop [2] is an open-source project of the Apache Software Foundation, and the concept was inspired from the search technology proposed by Google Inc. Using the Hadoop platform [5], programs can be developed that facilitate the processing of large amounts of data.

The advantages of Hadoop are that it is [2]:
Reliable. Hadoop can automatically back up several copies of the data. When the task fails, Hadoop reassigns the task.

Scalable. Hadoop is able to store and process information on PB (Peta Byte).
Distributed. Hadoop can use a combination of a group of ordinary equipment to distribute and carry out the mandate. By assigning tasks, Hadoop can process different data nodes at the same time, thereby increasing efficiency. Several sub-projects of Hadoop are listed below:

MapReduce [2] [5]. MapReduce is a programming model that was designed according to Hadoop from Google's papers [3]. The operation of MapReduce can be divided into two parts, i.e.,
1) Map and reduce; first, programmers have to analyze problems, design algorithms, and use the algorithms to determine the data that can be used for parallel computing.
2) The data are divided into small pieces, and through parallel processing, written into the program, Map.

A large number of machines can be required to process the Map program and simultaneously process each period of data. Finally, the results from the Map program and sent to the Reduce program where they are merged and compiled to produce the final results. MapReduce can simplify the processes of distributed and parallel computing. The program does not have to deal with traditional distributed and parallel computing technologies, which produce an additional burden of work, so it can focus on developing applied programs.
HDFS [2] (Hadoop Distributed File System). HDFS splits files into a number of data blocks and produces multiple copies. The copied data blocks are stored in a different node for the purposes of improving the reliability and efficiency of the file system when the file is read. HDFS was inspired by Google's GFS (Google File System) [8], and its development was based on GFS. HDFS is the Hadoop system's main storage platform. The structure of HDFS combines two software programs, i.e., Namenode and Datanode, and the execution platform is the Ubuntu system. HDFS uses the Java language, and any Java-based machine can run Namenode and Datanode software. Using the Java language means that HDFS can be easily transferred to different machines. A typical setting is in the cluster where only one machine is implementing Namenode software, and all of the other machines are implementing Datanode software. However, there is also a case in which a single computer runs multiple copies of Datanode software. Namenode and Datanode have built-in web servers, so users can always check the operational status of the cluster.

HDFS is a master-slave structure that consists mainly of Namenode software and multiple copies of Datanodes software to compose a cluster. The master server is responsible for managing the namespace and file operations of Datanode software, such as creating, deleting, and renaming files. A file is divided into multiple blocks and is stored on a group of Datanode software programs. Datanode software is responsible meeting users' needs of HDFS, such as reading or writing data and the implementation of block operations, such as create, delete, and copy from the order of Namenode software.

III. DATA HIDING METHOD

The data hiding method [1] [10] used here is a reversible data hiding program. By using binary trees, this method solves the histogram peaks that appear in pairs of spots. Differences in the distribution of pixels can hide more information, and the extent of distortion is also smaller. In addition, using histogram-shifting technology can prevent overflow and underflow. The performance and efficiency of the program are also better than those for existing programs.

A. MapReduce Model

The MapReduce programming model can be used to process large-scale data sets in cloud environments. It consists of three types of nodes: Master, Mappers and Reducer, as shown in Figure 1. The Master dispatches sub-jobs to a set of Mappers. After these Mappers complete their assigned jobs, the results are merged by the Reducer.

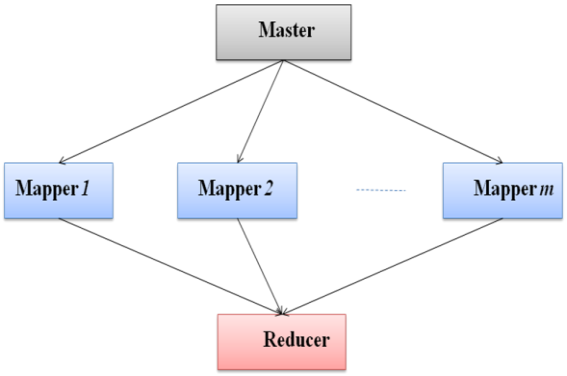

Fig 1. Map Reduce Model

Hadoop [2] implements the MapReduce model, which provides a high-level view to parallel programmers. With the help of Hadoop, programmers can focus on high-level operations. Based on the estimated information of workload distribution and node performance, we propose an MapReduce programming pattern for performance-

based workload distribution on cloud environments. This pattern consists of two modules: a Map module and a Reduce module. The Map module makes the scheduling decision and dispatches workloads to slaves. On the other hand, the Reduce module processes the assigned work. This algorithm is just a pattern, and the detailed implementation, such as data preparation, parameter passing, etc., might be different according to requirements of various applications.

IV. IMPLEMENTATION

In this study, we have implemented several scheduling schemes for the purpose of evaluation. The conventional static scheduling scheme is to equally distribute the total workload to each worker at compile time. However, this scheme is obviously not suitable for dynamic and heterogeneous environments. Therefore, a weighted static scheduling scheme is adopted in this experiment. The principle of partitioning is according to the CPU clock speed of each processor. A faster node will get more workloads than a slower one proportionally.

To reduce errors of experimental results, execution time in each experiment is obtained by averaging the results of five repetitive executions. A network and hardware structure diagram is shown in Fig 2.

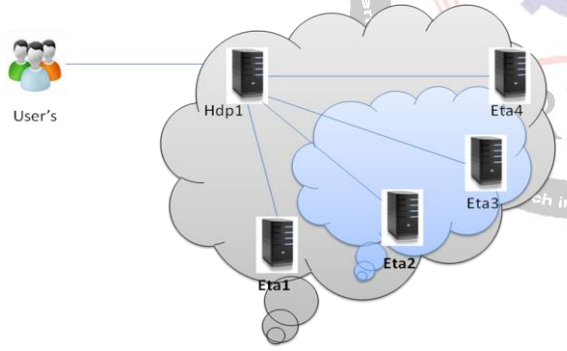

Fig 2. Cloud & Hardware Strucure

There are five machines used here are Hdp1, Eta1, Eta2, Eta3, and Eta4. Through the Internet cloud, the computers are linked together. The operating system used here is Ubuntu. The results of using the cloud computing platform of Hadoop.

Namenode. Hdp1 that segment program and distribute works to the slave; monitoring progress in the implementation subroutine and error detection.

Datanode. Eta 1~4 that accepting tasks from Hdp1and executing tasks.

In order to test and compare the difference between a single computer system and a cloud computing system, there must be a large amount of data being processed. The following is the number of hidden images and information we want to run. To hide so many pictures and data, we can foresee that there will be a huge amount of data, and this is exactly what we need.

The following steps explain data hiding flow in fig 3

1. Read image
2. Make pixel difference histogram
3. Make peak point
4. Execute data hiding

Fig 3. Feasibility Model for Data hiding in Map Reduce

In order to show the superior performance of a cloud computing system in processing large amounts of data, we designed a set of experiments that is divided into two parts; one part is the test group, and the other part is the control group.

Test group. We used Hdp1, Eta1, Eta2, Eta3, and Eta4 to build cloud computing environment as test group. We used the machinery and the Hadoop platform mentioned above to build the cloud computing environment. First, we took the images into the HDFS consisting of Data nodes and stored them. All files were saved by node in the cluster, and each image was divided into fixed-size blocks that were scattered throughout each node. The Hdp1 computer was used to implement data hiding and to compute the execution time associated with the parallel computing programs. Before the program was executed, we had to do the following,

(1) Define the format of the input data, here the data is image

(2) Design the data hiding program in the MAP function

(3) Design the calculation of the PSNR program in the Reduce function

(4) Define the format of the output image

When implementing the task, Hdp1 searched the disk address of the image that was being stored and then transferred the task to the disk address where the worker was. The worker transferred this address to HDFS to read out the image. Images were divided into sets of <key, value> by each worker's MapReduce program, with key being for the image name and value being an

image of the binary string. The Map function that was implementing data hiding in images produced intermediate <key, value> pairs in temporary memory. These intermediate <key, value> pairs were written to disk periodically, and the worker returned the hard disk storage addresses to Hdp1. After Hdp1 received the storage addresses, it distributed the Reduce task to store intermediate <key, value> pairs of nodes. The Reduce function we designed sorted value according to the intermediate key and confirmed that the result was correct. It returned an end value to Hdp1. If Hdp1 received all end values sent by Reduce task, data hiding was complete.

V. CONCLUSION

In this paper, we have presented the map-reduce technique for data hiding. We performed the data hiding technique using Hadoop system. However, in the cloud computing era, as long as there are good ideas and creativity, it is possible for only one person to create a large profit. Service providers, such as Google, Microsoft, Apple, and Yahoo, provide program development platforms and network operating system platforms that software developers around the world can use to develop and implement their ideas. In the foreseeable future, people will not need a PC, and all operations will be done on the web. There are network operating systems and web-based word processing software available, so consumers only need to pay fees according to the extent of their use rather than paying the cost of hardware. In addition, companies that provide services and software need only to verify the legitimacy of users, rather than being concerned about the software piracy problems. Through this paper, we can see that cloud computing environment compared to a single computer better position to handle large amounts of data, and demonstrate superior performance. This paper show cloud computing environment can effectively reduce the execution time of data hiding, so that more services can appear in the cloud computing environment.

REFERENCES

[1] W. L. Tai, C. M. Yeh, and C. C. Chang, ”Reversible Data Hiding Based on Histogram Modification of Pixel Differences,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 906-910, Jun. 2009.

[2] Apache Hadoop Project, Available :http://hadoop.apache.org/

[3] J. Dean, S. Ghemawat, “MapReduce: simplified data processing on large clusters,” *Proceedings of the 6th conference on Operating Systems Design & Implementation*, San Francisco, CA, December 2004, pp. 10- 10.

[4] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruer, “Bigtable: A Distributed Storage System for Structured Data,” *ACM Transactions on Computer Systems*, vol. 26, no. 2,pp. 1-26, Jun. 2008.

[5] E. Jaliya, P. Shrideep, and F. Geoffrey, ”MapReduce for Data Intensive Scientific Analyses,” *Proceedings of the IEEE Fourth International Conference on eScience*, pp. 277-284, December 2008.

[6] N. Leavitt, “Is Cloud Computing Really Ready for Prime Time?,” *IEEE Computer Magazine*, vol. 42, no. 1, pp. 15-20, 2009.

[7] C. Ranger, R. Raghuraman, A. Penmetsa, G. R. Bradski, and C. Kozyrakis, “Evaluating MapReduce for Multi-core and Multi-processor Systems,” *Proceedings of* the *IEEE 13th International Symposium on High Performance Computer Architecture (HPCA)*, Phoenix, Arizona, Feb. 2007, pp. 13-24.

[8] S. Ghemawat, H. Gobioff, and S. T. Leung, ”The Google File System,” *Proceedings of the 19th ACM Symposium on Operating Systems Principles*, Lake George, NY, Oct. 2003, pp. 29-43.

[9] J. Conner, ”Customizing Input File Formats for Image Processing in Hadoop,” *Technical report, Arizona State University*, Mesa, AZ, U.S.A.