International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-3 Issue-2, July 2013

The Design and Test of a Private Cloud Storage System, Part II

Zhaorui Wang, Christopher Fortson

Abstract*—Currently, cloud computing is a popular techniques. Many large-scale problems in practice require cloud computing and cloud storage. Even if public cloud is available, many private companies plan to build their private cloud for security reasons. This paper presents testing results of the proposed private cloud architecture in part I of this paper.

Index Terms*— Cloud computing, private cloud and YCSB.

I. INTRODUCTION

Based on private cloud storage system design, and the details of the test environment and test program given by Part I, this Part details the testing and analysis results. The tools and software used in the test can be found in [1]-[4] and also in [13]-[19]. The general research of cloud computing system can be found in [34]-[40]. The data for this testing is from a wireless sensor network (WSN). Because WSNs can be used in many areas and data from WSNs have diverse properties, it is a good choice for testing the proposed cloud storage system using data from WSNs. Research and different operation scenarios of WSNs can be found in [5]-[12] and also in [20]-[33].

II. TESTTING AND RESULTS ANALYSIS

Enter the appropriate commands to start the test.

Test Case1：./run_ycsb.sh yg0340 100k

Test Case2：./run_ycsb.sh yg0340 1m

Test Case3：./run_ycsb.sh yg0312 100k

Test Case4：./run_ycsb.sh yg0312 1m

By default, the mongodb.maxconnections parameter is 10, while the number of threads on the client is too many (e.g. greater than 50), it will throw an exception:

"com.mongodb.DBPortPool$SemaphoresOut: Out of semaphores to get db connection".

When connection pool resources are exhausted, this leds to incomplete load test data. To avoid this problem, mongodb.maxconnections parameter is set to a larger value (such as 100).

According to the test record, draw bar chart is as follows:

Workload A: Analysis of test results

1) *Workload A (Load)：Insert 100,000 records*

Manuscript published on 30 July 2013.
***Correspondence Author(s)**
Dr. Zhaorui Wang, National Astronomical Observatories, Chinese Academy of Sciences, Beijing, China.

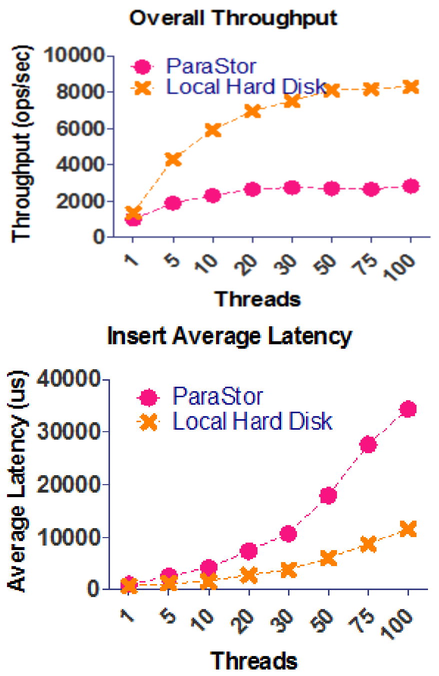
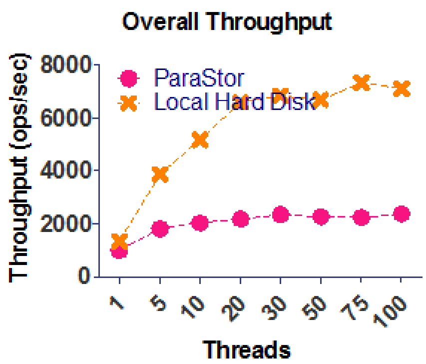


Figure 1. Insert operation（100k) : throughput and average delay

Distributed File System Based on ParaStor, the MongoDB database (In the following, only "ParaStor" is used) has the throughput of the insert operation 2827ops/sec. Based on the local hard drive MongoDB database, (Only "A local Hard Disk" is used later), an insert operation has throughput of 8295ops/sec. Only when performing a written operation, the performance of the use of a local hard disk is better than the performance of the use of ParaStor. Insertion delay increases as the number of concurrent threads increases. Using the local hard disk, the insertion delay significantly lower than the delay of using ParaStor.

2)Workload A (Load)：Insert 1 million records



Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

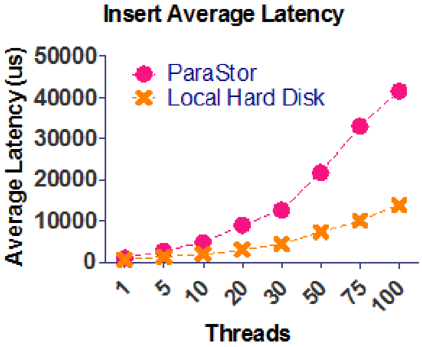


Figure 2. Insert operation (1M): throughput and average delay

ParaStor Throughput: 2388ops/sec, Local Hard Disk; throughput: 7346ops/sec. Local Hard Disk insert operation is superior to ParaStor. Insertion delay increases as the number of concurrent threads increases. Insert operation delay of Local Hard Disk is significantly lower than the delay of ParaStor.

3) Workload A (Run): 10 million records 50% + 50% read operations

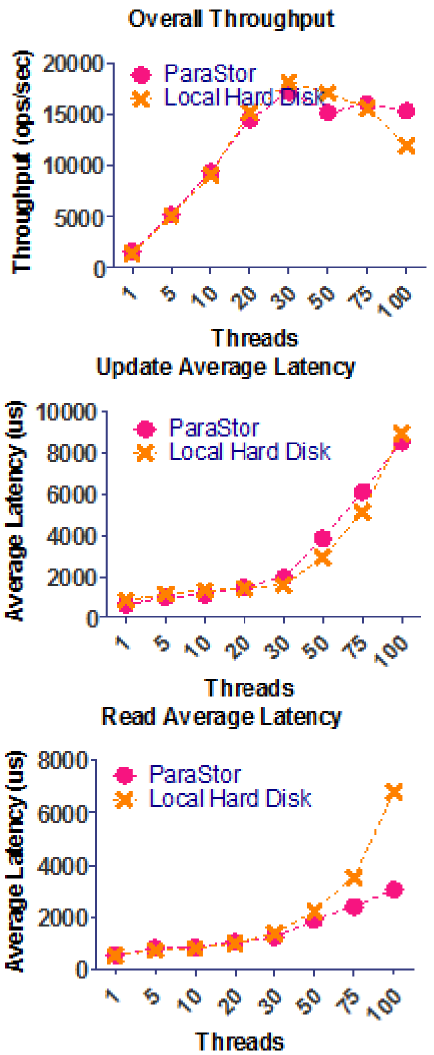


Figure 3. Insert operation（100k) : throughput and average delay

ParaStor Throughput: 17142ops/sec, Local Hard Disk; throughput: 18124ops/sec. Local Hard Disk Performance with ParaStor, and the difference is not obvious. Update delay increases as the number of concurrent threads increases. Local Hard Disk with ParaStor update operation delayed considerably; ParaStor slightly lower. Read latency increases as the number of concurrent threads increases. Local Hard Disk read operation delay is greater than the delay of ParaStor.

4)Workload A (Run)：1,000,000 records 50% + 50% read operations update

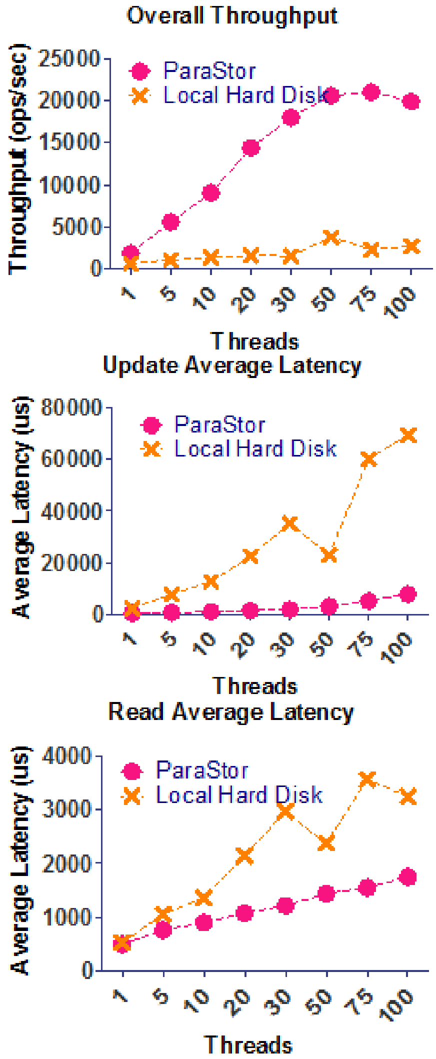


Figure 4. Workload A（1M): throughput and average delay

ParaStor Throughput: 21016ops/sec, Local Hard Disk; throughput: 3774ops/sec. Referring to Figure 5, when the data set increases, Local Hard Disk performance reduces dramatically. Update delay increases as the number of concurrent threads increases; Local Hard Disk update operation delay increases rapidly, thus leading to lower overall operating throughput. Read latency increases as the number of concurrent threads increases. Local Hard Disk read operation delay is greater than ParaStor.

Workload B: Analysis of test results

1)Workload B (Run)：100,000 records 95% + 5% read the update operation

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-3 Issue-2, July 2013

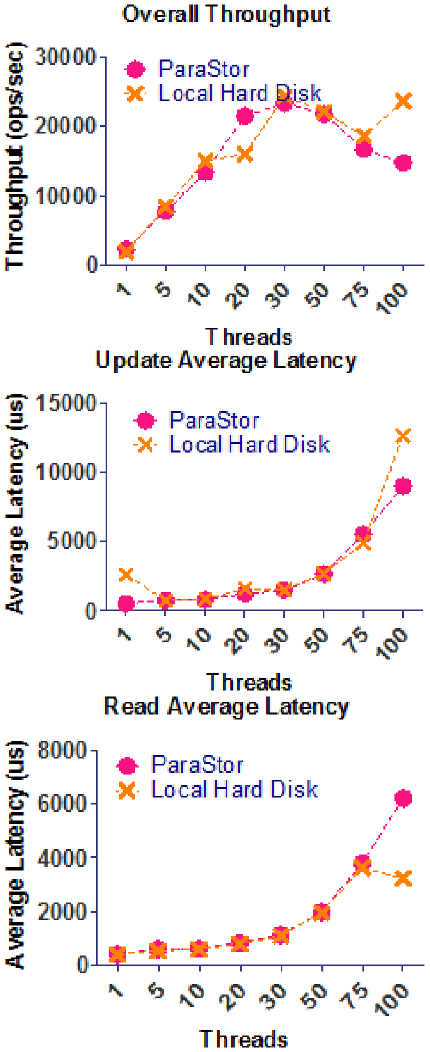
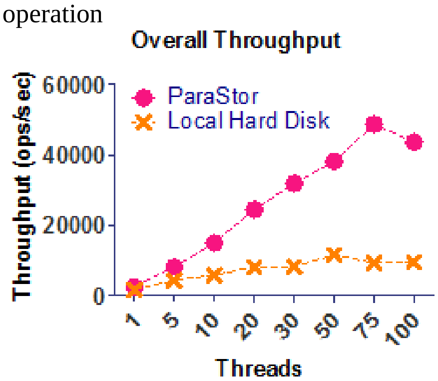


Figure 5. Workload B (100K): Throughput, average delay of update operations and average delay of read operations

ParaStor Throughput: 23346ops/sec, Local Hard Disk; throughput: 24275ops/sec. Local Hard Disk Performance with ParaStor is fair. Update delay increases as the number of concurrent threads increases. Local Hard Disk update operation delay is greater than ParaStor. Read latency increases as the number of concurrent threads increases. Local Hard Disk read operations has low latency than ParaStor.

2）Workload B (Run)：1,000,000 records 95% + 5% read the update operation



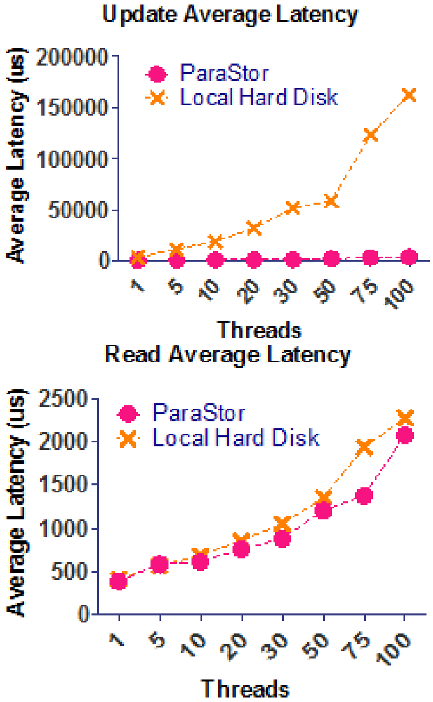


Figure 6. Workload B (1M): Throughput, average delay of update operations and average delay of read operations

ParaStor Throughput: 48796ops/sec, Local Hard Disk; throughput: 11733ops/sec. Local Hard Disk performance significantly worse than ParaStor. Update delay increases as the number of concurrent threads increases. Local Hard Disk delay of the update operation is much higher than ParaStor. Read latency increases as the number of concurrent threads increases. Local Hard Disk read operations delayed slightly than ParaStor.

Workload C: Analysis of test results

1)Workload C (Run) 100000 records read-only operations

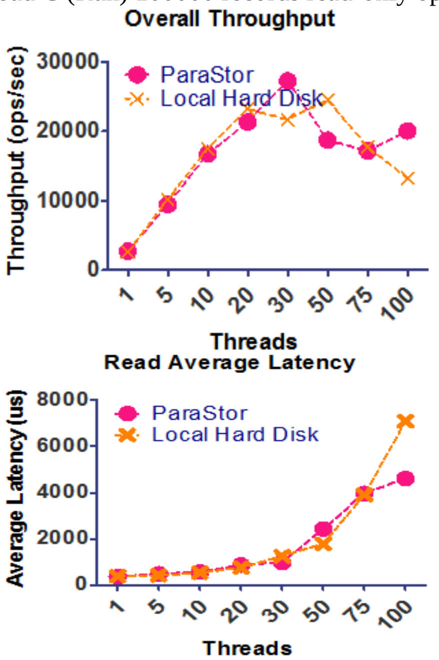


Figure 7. Workload C (100K): Throughput, average delay of update operations and average delay of read operations

Retrieval Number: B1018073213/13©BEIESP
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

ParaStor Throughput: 27260ops/sec, Local Hard Disk; throughput: 24528ops/sec. Local Hard Disk and ParaStor have little performance difference; ParaStor slightly better. Read latency increases as the number of concurrent threads increases. Local Hard Disk read operations delayed slightly more than ParaStor.

2) Workload C (Run) : 1,000,000 records read-only operations

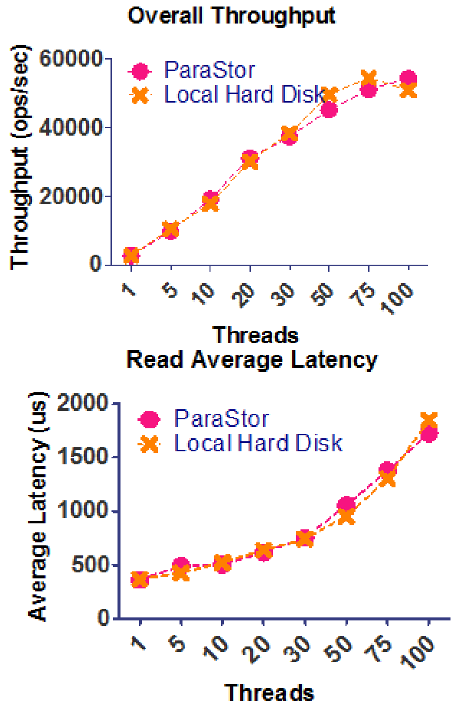


Figure 8. Workload C (1M): Throughput, average delay of update operations and average delay of read operations

ParaStor Throughput: 54318ops/sec, Local Hard Disk; throughput: 54400ops/sec. Local Hard Disk performances similarly with ParaStor. Read latency increases as the number of concurrent threads increases. Local Hard Disk and ParaStor have similar operation delay.

Workload F: Analysis of test results

ParaStor Throughput: 16988ops/sec, Local Hard Disk; throughput: 2073ops/sec. When the data set increases, Local Hard Disk performance reduces dramatically. Operational delay increases as the number of concurrent threads increases. When the data set increases, Local Hard Disk update operation delay greatly increased, resulting in overall performance dropped significantly.

III. Test Summary

Table 1. Operation Throughput Summary（ops/sec）

	Data Loaded	A	B	C	F
ParaStor（100k）	2887	17142	23346	27260	13428
ParaStor（1M）	2388	21016	48796	54318	16988
Local Hard Disk (100k)	8295	18124	24275	24528	15667
Local Hard Disk（1M）	7346	3774	11733	54400	2073

Note: In the table, A, B, C, F denotes Workload A, Workload B, Workload C, and Workload F, respectively.

Table 1 is a summary table of throughput. One can see: when performing the insert operation, ParaStor performance is better than the local hard disk; When data is small, besides the insert operation, ParaStor overall performance is not very different, compared to the local hard disk; when dealing with large data sets, due to the impact of the update operation, ParaStor's overall performance is better than the local hard drive.

IV. Conclusion

This paper presents a popular open source software based private cloud storage system design. The program distributed file systems and databases are designated as the dawn of the company's ParaStor parallel file system and the MongoDB NoSQL database. ParaStor and MongoDB were tested to see performance when they work together. The results showed that ParaStor and MongoDB have normal performance when working together and throughput meets the general needs of enterprises and institutions' applications. ParaStor performances are better when dealing with large data sets, compared to the local hard disk. This private cloud design has low hardware requirements. The cost of the use of open source software is low. Therefore, this design has good application prospects.

REFERENCES

1. OpenStack. Available: http://www.openstack.org/.
2. Eucalyptus. Available: http://www.eucalyptus.com/.
3. K. Ahmed, and M. Gregory, "Integrating Wireless Sensor Networks with Cloud Computing," 2011 Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN), pp.364-366, Dec. 2011.
4. W. Kurschl and W. Beer, "Combining cloud computing and wireless sensor networks", in Proceedings of the 11th ACM International Conference on Information Integration and Web-based Applications & Services, New York, NY, USA, pp. 512-518.
5. Z. X. Luo and T. C. Jannett, "Optimal threshold for locating targets within a surveillance region using a binary sensor network," in *Proc. of the International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering*, Dec., 2009.
6. Z. X. Luo and T. C. Jannett, "Energy-based target localization in Multi-hop wireless sensor networks," in *Proc. of the 2012 IEEE Radio and Wireless Symposium*, Santa Clara, CA, Jan. 2012.
7. H. Chen and P. K. Varshney, "Nonparametric quantizers for distributed estimation," IEEE Trans. Signal Process., vol 58, no 7, pp. 3777-3787, July 2010.
8. Z. X. Luo and T. C. Jannett, "Modeling sensor position uncertainty for robust target localization in wireless sensor networks," in *Proc. of the 2012 IEEE Radio and Wireless Symposium*, Santa Clara, CA, Jan. 2012.
9. H. Chen and P. K. Varshney, "Performance Limit for Distributed Estimation Systems with Identical One-Bit Quantizers," IEEE Transaction on Signal Processing, Vol. 58, No. 1, pp. 466-471, Jan. 2010.
10. Z. X. Luo and T. C. Jannett, "Performance comparison between maximum likelihood and heuristic weighted average estimation methods for energy-based target localization in wireless sensor networks," in *Proc. of the 2012 IEEE SoutheastCon Conference*, Orlando, FL, Mar. 2012.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

11. Q. Cheng, P. K. Varshney, J. H. Michels, and C. M. Belcastro, "Distributed fault detection with correlated decision fusion," IEEE Trans. Aerosp. Electron. Syst., Volume 45, No. 4, pp.1448 – 1465, October 2009.
12. Z. X. Luo and T. C. Jannett, “A multi-objective method to balance energy consumption and performance for energy-based target localization in wireless sensor networks,” in Proc. of the 2012 IEEE SoutheastCon Conference, Orlando, FL, Mar. 2012.
13. Lustre. Available: http://lustre.org/.
14. HDFS. Available: http://hadoop.apache.org/.
15. MongoDB Available:http://www.mongodb.org/.
16. Tomcat. Available: http://tomcat.apache.org/.
17. Nginx. Available: http://nginx.org/.
18. Yahoo! Cloud Serving Benchmark. Available:
19. http://research.yahoo.com/Web_Information_Management/YCSB.
20. ParaStor. Available:
21. http://www.sugon.com/product/detail/productid/37.html.
22. Z. X. Luo, “A censoring and quantization scheme for energy-based target localization in wireless sensor networks,” *Journal of Engineering and Technology*, vol.2, no.2, Aug. 2012.
23. K. C. Ho, “Bias reduction for an explicit solution of source localization using TDOA,” *IEEE Trans. Signal Processing*, vol. 60, pp. 2101-2114, May 2012.
24. Z. X. Luo, “A coding and decoding scheme for energy-based target localization in wireless sensor networks,” *International Journal of Soft Computing and Engineering*, vol.2, no. 4, Sept. 2012.
25. Y. Li, K. C. Ho, and M. Popescu, “A microphone array system for automatic fall detection,” IEEE Trans. Biomedical Engineering, vol. 59, pp. 1291-1301, May 2012.
26. K. C. Ho and R. Rabipour, “A design and use case for inhibiting the adaptation of echo canceller without using external control,” Contribution C816R1, *ITU WP1/SG16 Standard Meeting*, Geneva, Switzerland, May 2012.
27. Z. X. Luo, “Anti-attack and channel aware target localization in wireless sensor networks deployed in hostile environments,” *International Journal of Engineering and Advanced Technology*, vol. 1, no. 6, Aug. 2012.
28. Y. Shang, W. Zeng, K. C. Ho, D. Wang, Q. Wang, Y. Wang, T. Zhuang, A. Lobzhanidze, and L. Rui, “NEST: networked smartphones for target localization,” in Proc. IEEE CCNC 2012, Las Vegas, Jan. 2012, pp. 732-736.
29. Z. X. Luo, “Robust energy-based target localization in wireless sensor networks in the presence of byzantine attacks,” *International Journal of Innovative Technology and exploring Engineering*, vol. 1, no.3, Aug. 2012.
30. L. Yang and K. C. Ho, “Alleviating sensor position error in source localization using calibration emitters at inaccurate locations,” IEEE Trans. Signal Processing, vol. 58, pp. 67-83, Jan. 2010.
31. Z. X. Luo, “A new direct search method for distributed estimation in wireless sensor networks,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 1, no. 4, Sept. 2012.
32. T. Glenn, J. N. Wilson, and K. C. Ho, “A multimodal matching pursuits dissimilarity measure applied to landmine/clutter discrimination,” in Proc. IEEE Int. Geoscience and Remote Sensing Symp. IGARSS, Honolulu, July 2010.
33. M. Popescu, K. E. Stone, T. C. Havens, J. M. Keller, and K. C. Ho, “Anomaly detection in forward-looking infrared imaging using one-class classifiers,” in Proc. SPIE Conf. Detection and Remediation Technologies for Mines and Minelike Targets XV, Orlando, Apr. 2010.
34. Z. X. Luo, “Distributed estimation and detection in wireless sensor networks,” *International Journal of Inventive Engineering and Sciences*, vol. 1, no. 3, Feb. 2013.
35. T. C. Havens, C. J. Spain, K. C. Ho, J. M. Keller, Tuan T. Ton, D. C. Wong, and M. Soumekh, “Improved detection and false alarm rejection using FLGPR and color imagery in a forward-looking system,” in Proc. SPIE Conf. Detection and Remediation Technologies for Mines and Minelike Targets XV, Orlando, Apr. 2010.
36. George Kousiouris, Tommaso Cucinotta, Theodora Varvarigou, "The Effects of Scheduling, Workload Type and Consolidation Scenarios on Virtual Machine Performance and their Prediction through Optimized Artificial Neural Networks"[19] , The Journal of Systems and Software (2011),Volume 84, Issue 8, August 2011, pp. 1270-1291, Elsevier, doi:10.1016/j.jss.2011.04.013.
37. Ko, Ryan K. L. Ko; Kirchberg, Markus; Lee, Bu Sung (2011). "From System-Centric Logging to Data-Centric Logging - Accountability, Trust and Security in Cloud Computing". Proceedings of the 1st Defence, Science and Research Conference 2011 - Symposium on Cyber Terrorism, IEEE Computer Society, 3–4 August 2011, Singapore.
38. Ko, Ryan K. L.; Jagadpramana, Peter; Mowbray, Miranda; Pearson, Siani; Kirchberg, Markus; Liang, Qianhui; Lee, Bu Sung (2011). "TrustCloud: A Framework for Accountability and Trust in Cloud Computing". Proceedings of the 2nd IEEE Cloud Forum for Practitioners (IEEE ICFP 2011), Washington DC, USA, July 7–8, 2011.
39. Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, Dmitrii Zagorodnov, “The Eucalyptus Open-source Cloud-computing System”, Computer Science Department, University of California - Santa Barbara, Santa Barbara, California 93106.
40. Mike P. Papazoglou, “Service -Oriented Computing: Concepts, Characteristics and Directions”, Tilburg University, INFOLAB.
41. Rajiv Ranjan, Rajkumar Buyya, “Decentralized Overlay for Federation of Enterprise Clouds”, Grid Computing and Distributed Systems (GRIDS) Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne, Australia.
42. Marianne C. Murphy, Marty McClelland, “Computer Lab to Go: A “Cloud” Computing Implementation”, Proc ISECON 2008, v25.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)