

PERBANDINGAN DISTRIBUTED REPLICATED DENGAN STRIPED REPLICATED UNTUK MEREPLIKA FILE DALAM GLUSTERFS PADA COMPUTER CLUSTER

Fandy Mahendra¹, Fazmah Arief Yulianto², Gandeva Bayu Satrya³

¹Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Tingkat ketersediaan file pada server di dalam suatu jaringan menjadi tuntutan yang harus selalu dipenuhi. Para pengguna tentunya ingin selalu dapat mengakses file kapan pun. Padahal apabila terjadi suatu gangguan, misalnya server down, maka file tidak dapat diakses. Maka dari itu dibutuhkan adanya replikasi file untuk mengatasi permasalahan tersebut. Jadi, ketika terjadi server down, maka pengguna tetap dapat mengakses file yang mereka butuhkan. File yang ada pada server direplikasi ke server-server lain. Semua server tersebut tergabung di dalam suatu Cluster, yang kemudian disebut dengan computer Cluster. Dan salah satu file system yang dapat digunakan untuk mereplikasi file pada computer Cluster adalah GlusterFS.

GlusterFS memiliki beberapa metode untuk mereplikasi file, diantaranya adalah Distributed Replicated dan Striped Replicated. Pada Distributed Replicated, file didistribusikan ke server-server yang berada di dalam satu Cluster. Sedangkan Striped Replicated memiliki satu tahap yang lebih banyak, yaitu sebelum mendistribusikan file ke server-server yang berada di satu Cluster, file tersebut dipecah-pecah terlebih dahulu. Dalam penelitian kali ini, yang dianalisis adalah tingkat efektifitas dan efisiensi dari kedua metode tersebut pada saat proses replikasi file berlangsung.

Hasil dari penelitian ini, kedua metode tersebut sama-sama efektif dalam mereplikasi file. Sedangkan dari tingkat efisiensi, secara keseluruhan, Distributed Replicated lebih efisien daripada Striped Replicated.

Kata Kunci : GlusterFS, replikasi, Distributed Replicated, Striped Replicated, Cluster

Abstract

File availability is a requirement that must be fulfilled. Users always want to access their files whenever they want. Although there are problems occur, for an example, server down, so the files cannot be accessed. Thus, data replication is needed to solve the problem. So whenever the server is down, users still can access their files. Those files are replicated to other servers. All these servers are connected each other, create a new system called computer Cluster. One of the file system that support file replication in computer Cluster is GlusterFS.

GlusterFS has some file replication methods, such as Distributed Replicated and Striped Replicated. In the Distributed Replicated algorithm, the files are distributed to other servers that connected to a Cluster, on the other hand, Striped Replicated has further step before distribute the files to another servers, each file is splitted at first. In this research, the performance values that being analyzed are effectivity and efficiency rate from those two algorithms when doing file replication with GlusterFS.

And the result of this research, these two algorithms both are effective to replicate the files. Based on the efficiency rate, the result shows that Distributed Replicated has more efficient than Striped Replicated.

Keywords : GlusterFS, replication, Distributed Replicated, Striped Replicated, Cluster

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam suatu jaringan, sebuah *server* sangatlah berperan penting. *Server* bertugas untuk menyediakan *file* yang diminta oleh user. Pada awalnya di dalam suatu environment hanya terdapat satu *server*. Keadaan ini memiliki celah, yaitu apabila ada masalah pada *server* tersebut, maka user tidak dapat mengakses *server* dengan baik. Hal ini memunculkan ide untuk dikembangkannya *computer Cluster*. *Computer Cluster* terdiri dari beberapa komputer, dalam hal ini berupa komputer *server*. Namun dari sisi lain, yaitu user, melihat bahwa hanya ada satu komputer saja yang bertindak sebagai *server* di dalam environment tersebut. *Computer Cluster* dapat dijadikan solusi dalam menjamin selalu adanya ketersediaan *file* pada *server* karena apabila terjadi masalah pada salah satu *server*, maka masih ada *server* lain yang siap mengambil alih peran untuk memenuhi permintaan *file* dari user. Salah satu *distributed file system* yang dapat digunakan untuk membangun *computer Cluster* adalah dengan menggunakan GlusterFS.

Dalam beberapa *server* yang tergabung dalam *Cluster computing*, kesamaan *file* di setiap *server* juga harus dijamin kualitasnya. Maka untuk mendukung kesamaan *file* dalam setiap *server* pada *computer Cluster*, juga harus diterapkan teknik replication [4]. Replikasi adalah kunci untuk ketersediaan tinggi di dalam *computer Cluster*. Tingkat ketersediaan *file* yang tinggi untuk memenuhi request dari user merupakan hal yang dituntut oleh user kepada *server*. Dengan teknik replikasi, maka diharapkan kesamaan *file* diantara satu *server* dengan *server* yang lainnya dalam satu *Cluster computing* dapat selalu terjaga. Keuntungan dengan adanya replikasi *file* adalah user cukup hanya dengan satu kali melakukan penyimpanan *file*, maka kemudian *file* tersebut direplika ke beberapa *server* yang berada dalam satu *Cluster computing* secara otomatis, begitu juga dengan proses menghapus *file* [5]. Kelebihan lain adalah pada saat terjadi masalah pada salah satu *server*, maka *server* lain dapat bertindak sebagai back-up *file*.

Metode replikasi *file* yang dimiliki oleh GlusterFS diantaranya adalah *Distributed Replicated* dan *Striped Replicated* [3]. Seperti namanya, dalam proses replikasi *file*, *Distributed Replicated* langsung mendistribusikan *file* yang direplika ke masing-masing *server*. Sedangkan pada *Striped Replicated*, hasil replikasi berupa pecahan *file* yang sudah direplika dan berada pada *server* yang berbeda.

Masalah yang timbul adalah belum adanya penelitian yang membuktikan efektifitas dan membandingkan tingkat efisiensi diantara kedua metode tersebut.

Jadi, dari sisi penyedia layanan dan pengguna replikasi *file* masih belum mengetahui tentang efektifitas dan perbandingan efisiensi kedua metode tersebut pada saat mereplika *file* dengan GlusterFS sebagai *distributed file system* pada environment mereka, dalam hal ini adalah pada *computer Cluster*. Dari segi penyedia layanan replikasi *file*, lebih memperhatikan faktor beban pada jaringan yang berkaitan dengan parameter *overhead byte*. Sedangkan dari sisi pengguna atau klien, memprioritaskan lama waktu replikasi dan penggunaan *resources* (CPU dan RAM) pada saat proses replikasi *file* berlangsung.

1.2 Perumusan Masalah

Berdasarkan latar belakang permasalahan di atas, maka penulis merumuskan permasalahan yang dibahas dalam Tugas Akhir ini, yaitu :

1. Apakah *Distributed Replicated* dan *Striped Replicated* efektif dalam mereplikasi *file* dan bagaimana tingkat efisiensi masing-masing?

1.3 Tujuan

Berdasarkan perumusan masalah di atas, maka tujuan dari Tugas Akhir ini, yaitu :

1. Mengetahui efektifitas *Distributed Replicated* dan *Striped Replicated* dalam mereplikasi *file*.
2. Membandingkan tingkat efisiensi antara *Distributed Replicated* dengan *Striped Replicated* berdasarkan parameter pengujian.

1.4 Hipotesa

Dari segi efektifitas, *Distributed Replicated* dan *Striped Replicated* sudah efektif untuk mereplika *file*. Sedangkan jika dibandingkan tingkat efisiensinya, *Striped Replicated* lebih efisien daripada *Distributed Replicated*, karena sebelum direplika, *file* dipecah-pecah terlebih dahulu, baru kemudian hasil pecahan *file* tersebut direplika ke *server*.

1.5 Batasan Masalah

Adapun batasan masalah pada tugas akhir ini, yaitu:

1. Replikasi merupakan suatu aksi atau proses yang dilakukan untuk mengkopi data atau *file* di tempat yang terpisah secara geografis [2]. Sedangkan dari pengertian tersebut dapat diturunkan menjadi suatu replikasi *file* dikatakan efektif apabila replikasi *file* dapat menangani kondisi:
 - a. *Add file*. Maksudnya jika ada *file* baru yang disimpan di komputer klien, maka *file* tersebut juga tersimpan di komputer-komputer *server* dalam satu *Cluster* secara otomatis.

- b. *Delete file*. Maksudnya apabila ada *file* yang dihapus di komputer klien, maka *file* yang sama yang terdapat di komputer-komputer *server* dalam satu *Cluster* juga ikut terhapus secara otomatis.
2. Parameter yang digunakan dalam mengukur tingkat efisiensi saat replikasi *file* berlangsung adalah lama waktu replikasi, penggunaan *resources* (CPU dan RAM) [9], dan *overhead byte*.
3. PC yang digunakan dalam penelitian kali ini berjumlah lima buah, dengan rincian satu buah PC untuk komputer klien dan empat buah PC sebagai komputer *server*.
4. Tidak menggunakan *routing protocol* karena topologi jaringan yang digunakan masih berada dalam satu subnet yang sama.
5. Tidak menambahkan beban pada jaringan atau background traffic.
6. Tidak mengukur Quality of Service (QoS) karena topologi jaringan masih berada di dalam satu subnet.

1.6 Metodologi Penyelesaian Masalah

Metodologi yang digunakan untuk memecahkan permasalahan-permasalahan dalam Tugas Akhir ini terdiri dari tahap-tahap berikut:

1. Tahap Studi Literatur

Pada tahap ini dilakukan pencarian referensi dan materi yang ada di internet serta memahami dan mempelajarinya sehingga dapat digunakan untuk menyelesaikan permasalahan dalam Tugas Akhir ini. Pencarian referensi berkaitan dengan topik Tugas Akhir ini, yaitu *computer Cluster*, dan lebih mendalam mengenai GlusterFS dan metode replikasi *file* yang dimiliki oleh GlusterFS, serta berbagai pustaka seperti jurnal, makalah penelitian dan literatur lainnya yang membahas penelitian yang terkait. Pustaka yang dipelajari tersebut berasal dari berbagai penelitian dari beraneka ragam sumber informasi yang pernah dilakukan oleh orang lain sebelumnya.
2. Tahap Perancangan Percobaan

Pada tahap ini dilakukan analisis sistem dan menentukan kebutuhan dari sistem yang dibangun seperti kebutuhan fungsional sistem, spesifikasi perangkat lunak dan perangkat keras, serta gambar topologi jaringan yang dibangun. Kemudian dilakukan proses perancangan percobaan untuk menguji metode replikasi *file Distributed Replicated* dan *Striped Replicated* pada GlusterFS di dalam *computer Cluster*. Diantara rancangan pengujian yang dilakukan adalah dengan mereplika beberapa *file* yang memiliki variasi ukuran yang beragam dan variasi jumlah *file*.
3. Tahap Melakukan Percobaan

Pada tahap ini dilakukan implementasi berdasarkan rancangan sistem yang telah dibuat sebelumnya. Kemudian dilakukan pengujian sesuai dengan rancangan percobaan.

4. Tahap Analisis Data Hasil Percobaan dan Menarik Kesimpulan

Pada tahap ini dilakukan pengambilan data hasil pengujian, berdasarkan parameter yang digunakan sebagai alat ukur tingkat performansi metode replikasi *file*. Serta dilakukan analisis terhadap data tersebut agar bisa ditarik kesimpulan. Pada tahap ini dinilai kemampuan *system* dalam melakukan replikasi *file*.

5. Tahap Menyusunan Laporan Tugas Akhir

Pada tahap ini dilakukan dokumentasi tahap-tahap selama kegiatan penelitian berlangsung dan hasil yang didapat selama pengujian ke dalam laporan Tugas Akhir. Dijelaskan pula mengenai langkah-langkah secara detail dari awal, seperti menganalisis kebutuhan sistem, perancangan sistem, implementasi, pengujian, serta analisis hasil pengujian.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari hasil penelitian dan analisa yang telah dilakukan adalah:

- a. *Distributed Replicated* dan *Striped Replicated* sama-sama efektif untuk mereplika *file* dalam GlusterFS pada *computer Cluster*.
- b. Dari segi lama waktu replikasi, *Distributed Replicated* lebih efisien daripada *Striped Replicated*.
- c. Dari segi penggunaan *resources* CPU, kedua *Distributed Replicated* dan *Striped Replicated* membutuhkan jumlah yang sama.
- d. Dari segi penggunaan *resources* RAM, *Striped Replicated* lebih efisien daripada *Distributed Replicated*.
- e. Dari segi jumlah *overhead byte*, *Distributed Replicated* lebih efisien daripada *Striped Replicated*.
- f. Secara keseluruhan, dilihat dari sisi pengguna, *Distributed Replicated* lebih efisien daripada *Striped Replicated*. Begitu juga jika dilihat dari sisi penyedia layanan replikasi *file*, yang lebih efisien diantara keduanya adalah *Distributed Replicated*.

5.2 Saran

Saran yang dapat diberikan untuk penelitian lebih lanjut mengenai topik ini adalah:

- a. Menggunakan jumlah PC klien dan *server* yang lebih banyak lagi.
- b. Menggunakan lebih dari satu subnet pada topologi jaringan.
- c. Menambahkan *routing protocol* pada jaringan.
- d. Menambahkan beban pada jaringan dengan background traffic.
- e. Mengukur Quality of Service (QoS).

DAFTAR PUSTAKA

- [1]Agrawal, D. and El Abbadi, A. 1990. *The Tree Quorum Protocol: An Efficient Approach for Managing Replicated Data*. Department of California Santa Barbara.
- [2]Bell W. H., Cameron D. G., Capozza L., Millar A. P., Stockinger K., Zini F. 2003. *OptorSim – A Grid Simulator for Studying Dynamic Data Replication Strategies*. International Journal of High Performance Computing Applications. November 2003 vol. 17 no. 4 page 403-416.
- [3]Beloglazov, Anton., Piraghaj, Sareh Fotuhi., Alkorayan, Mohammed., and Buyya Rajkumar. 2012. *Deploying OpenStack on CentOS Using the KVM Hypervisor and GlusterFS Distributed File System*. Department of Computing and Information Systems, The University of Melbourne, Australia.
- [4]Bzoch, Pavel and Safarik Jiri. 2012. *Algorithms For Increasing Performances In Distributed File Systems*. Department of Komputer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Pilzen, Czech Republic.
- [5]Dahlberg, Tobias. 2012. *Distributed Storage and Processing of Image Data*. Department of Komputer and Information Science, Linkoping University, Sweden.
- [6]Dias, Daniel Manuel and Mukherjee, Rajat. 2001. *Real-Time Shared Disk System For Computer Clusters*. International Business Machines Corporation, Armonk, New York, United States.
- [7]Goel, Sushant and Buyya Rajkumar. 2007. *Data Replication Strategies in Wide-Area Distributed Systems*. University of Melbourne, Australia.
- [8]Hsu, Chen-Ting; Guo-Heng Luo and Shyan-Ming Yuan. 2013. *Personalized Cloud Storage System: A Combination of LDAP Distributed File System*. Institute of Computer Science and Engineering, National Chiao-Tung University, Taiwan.
- [9]Kutcher, Jeffrey Spring. 2000. *Computer Cluster Performance Monitoring Utility*. European Patent Office.
- [10]Kwidama, Sevickson. 2008. *Streaming and Storing CineGrid Data: A Study on Optimization Methods*. Master of Science in System and Network Engineering, University of Amsterdam.

- [11] Liu, Pangfeng. 2010. *Virtual Machine Dynamic Provision and Image Management for Unicode*.
- [12] Lokeshwari, Y. V.; B. Prabavathy and Chitra Babu. 2011. *Optimized Cloud Storage with High Throughput Deduplication Approach*. Computer Science and Department of Engineering SSN College of Engineering, Anna University, Chennai – 603110, Tamil Nadu, India.
- [13] Muntimadugu, Divya and Sriram, Anjana Suparna. 2013. *Red Hat Storage 2.0, Administration Guide, Configuring and Managing Red Hat Storage Server, Edition 1*. Red Hat Inc.
- [14] Naderuzzman, Mohammad and Akhtar, Dr. Md. Nasim. 2012. *An Asynchronous Replication Model to Improve Data Available into a Heterogeneous System*. Department of Computer Science & Engineering. Dhaka University of Engineering & Technology.
- [15] Nordstrand, Rickard. 2009. *A Cross-Platform, High Performance Shared Storage System*. Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden.
- [16] Perkov, Luka; Nikola Pavković and Juraj Petrović. 2011. *High-Availability Using Open Source Software*. Croatia. Dalam Mipro, page 167-170. IEEE.
- [17] Pisacka, J., Hron, M., Janky, F., Panek, R. 2012. *Cluster Storage for COMPASS Tokamak*. Fusion Engineering and Design, Volume 87, Issue 12, Desember 2012, Page 2238-2241.
- [18] platformusers.net. 2013. *Advantages of Cluster Computing*. <http://www.platformusers.net/Cluster-computing/advantages-of-Cluster-computing.html> Di-download pada tanggal 10 Oktober 2013.
- [19] Red Hat®, Inc. *Cluster Basics*. http://www.centos.org/docs/5/html/Cluster_Suite_Overview/s1-clstr-basics-CSO.html Didownload pada tanggal 10 Oktober 2013.
- [20] Sanchez, Daniel. 2007. *Design of Store-and-Forward Servers for Digital Media Distribution*. Master of Science in System and Network Engineering, University of Amsterdam.
- [21] Schwartz, B; Peter Zaitsev & Vadim Tkachenko. 2008. *High Performance MySQL, 2nd Edition*. United States of America: O'Reilly Media, Inc.
- [22] Verkuil, Steven. 2013. *A Comparison of Fault-Tolerant Cloud Storage File Systems*. University of Twente, Netherland.

- [23] Zhao, Liang; S. Sakr and A. Liu. 2012. *Application-Managed Replication Controller for Cloud-Hosted Databases*. School of Computer Science and Engineering University of New South Wales, Australia. IEEE.

