Jurnal Ilmiah Ilmu Komputer Vol. 6, No.1, Bulan April 2020 (P) ISSN 2442-451X
Fakultas Ilmu Komputer (O) ISSN 2503-3832
Universitas AL Asyariah Mandar

BACKUP DATABASE DENGAN MULTI MASTER REPLIKASI PADA KLUSTER SERVER

Ahmad Heryanto[1,2] Yuyun Hartati[3]

[1]Laboratorium Jaringan Komputer dan Komunikasi Data, Sistem Komputer Universitas Sriwijaya
[2]Teknik Elektro Universitas Sriwijaya
[3]Teknik Komputer Univesitas Sriwijaya
hery@unsri.ac.id[1], yunha270395@gmail.com[2]

ABSTRAK

Sistem *database* yang selalu tersedia merupakan kebutuhan mutlak pada insfrastruktur setiap organisasi. Data yang tersimpan harus terus tersedia pada saat dibutuhkan. Banyak hambatan dan gangguan dalam menyimpan data dalam suatu *database*. Hambatan dan gangguan tersebut tersebut bisa diakibatkan dari berbagai macam hal, seperti *maintenance*, kerusakan *database*, kerusakan media, data c*orruption* dan bencana yang tak terduga (bencana alam). Oleh karena itu, diperlukan suatu Teknik replikasi yang baik untuk menjaga keutuhan data. Teknik replikasi Multi Master (Master-Master) telah diimplementasikan pada penelitian ini. Hasil penelitian ini, untuk melakukan replikasi 3 *node* server diperlukan waktu dengan rentang 0,1-0,2 detik.

Kata kunci: Replikasi, *Database*, Server, Multi Master, Kluster

ABSTRACT

A database system that is always available is an absolute necessity in the infrastructure of every organization. Stored data must continue to be available when needed. Many obstacles and disruptions in storing data in a database. These obstacles and disruptions can be caused by various things, such as maintenance, database damage, media damage, data corruption, and natural disasters. Therefore, we need a good replication technique to maintain data integrity. Multi-Master Replication Techniques have been implemented in this study. The results of this study, to replicate 3 server nodes take a time span of 0.1-0.2 seconds.

Keywords: Replication, Database, Server, Multi Master, Cluster

Copyright © 2020 FIKOM – UNASMAN

1. PENDAHULUAN

Pada era digital seperti sekarang ini, kebutuhan akan *database* merupakan bagian yang tidak terpisahkan dari suatu organisasi, *database* digunakan oleh organisasi untuk menyimpan, dan mengolah data dengan rapi dan baik. Suatu organisasi memandang data sebagai aset yang bernilai tinggi dan tidak dapat digantikan dengan benda lain. Oleh karena itu, sangat penting sekali menjaga *database* tersebut dengan baik. Banyak hambatan dan gangguan dalam pengelolaan *database*. Hambatan dan gangguan tersebut tersebut bisa diakibatkan dari berbagai macam hal, seperti *maintenance*, kerusakan *database*, kerusakan media, data c*orruption* dan bencana yang tak terduga (bencana alam)[1]. Hambatan dan gangguan pada suatu *database*, dapat mengakibatkan hilangnya data yang yang tersimpan di dalamnya. Kehilangan data ini dapat menyebabkan kerugian bagi pemilik.

Dalam mencegah kehilangan data pada suatu *database* biasanya digunakan teknik *backup*, Teknik *backup* yang dilakukan oleh kebanyakan orang adalah dengan menyalin file-file *database* kedalam suatu media penyimpanan tertentu seperti hardisk, ssd, dan flashdisk. Cara manual ini tidak efektif, karena harus dilakukan secara berkala dan terus-menerus secara manual. Jika suatu hari ada perubahan/penambahan/penghapusan terhadap data yang dilakukan pada media penyimpanan utama maka tidak akan mengubah data yang ada pada media penyimpanan cadangan, sehingga harus dilakukan *update* secara manual pada data data yang mengalami perubahan. Ketersediaan data yang baik secara *realtime* terkadang menjadi kebutuhan yang rumit pada masalah tertentu. Masalah tersebut berkaitan dalam hal kinerja sistem, jumlah pengaksesan data, serta lalu lintas jaringan yang dalam perkembangannya semakin bertambah padat. Hal tersebut memicu timbulnya teknologi replikasi data pada masa ini[2].

Proses replikasi data adalah suatu proses penggandaan data. Penggandaan data pada *database* sering dilakukan pada sistem *database* terdistribusi. Sistem *database* terdistribusi merupakan kumpulan kluster *database* yang dapat berpartisipasi dalam proses transaksi data pada *database*. Data dapat diakses pada kluster dengan menggunakan *query* tertentu. Terdapat beberapa hal yang mendasari dibentuknya kluster *database*, seperti pemakaian bersama (*share)*, kehandalan (*reliability*), ketersediaan (*availability)* dan kecepatan pemrosesan *query*[2], [3]. Replikasi sangat berguna untuk melindungi ketersedian data yang akurat. Teknik replikasi yang baik untuk *database* adalah menggunakan teknik replikasi *synchronous*. Teknik ini dapat melakukan *backup database* secara *realtime* kepada semua *node* pada kluster server *database*. Kluster *database* pada penelitian ini menggunakan Galera Cluster. Galera Cluster menerapakan teknik replikasi multi-master dimana setiap *node* berperan sebagai master dan client dapat melakukan *query insert*, *delete*, *drop* ataupun *update* dari *node* manapun. Galera Cluster akan menjamin perubahan data pada suatu *node* akan memicu perubahan pada *node* lainnya secara *realtime*. Transaksi data pada suatu kluster *database* secara *realtime* dapat digunakan sebagai *backup* data pada suatu sistem *database*.

2. TINJAUAN PUSTAKA

2. 1. Database

Istilah *database* berasal dari bidang Ilmu Komputer, yang menjelaskan tentang sekumpulan informasi yang disimpan dalam komputer secara sistematik dan terstruktur. sehingga data dapat disimpan dan diolah dengan menggunakan suatu program komputer. program komputer yang dipakai untuk *database* dikenal dengan *database* management system (DBMS). Banyak produk-produk DBMS yang popular digunakan pada bidang komputer, yaitu[4]

1. MYSQL
2. MariaDB
3. Oracle *Database*
4. PostgreSQL
5. MongoDB
6. MemSQL
7. Interbase
8. Dan lain sebagainya.

Data yang tersimpan di dalam *database* terdiri dari dua jenis, yaitu data yang terstruktur data dan data semi terstruktur. Oleh karena itu, data yang tersimpan di dalam *database* memiliki bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom. Proses penyimpanan, pengolahan dan pengambilan data pada *database* menggunakan *query*. *Query* yang digunakan memiliki bentuk dan kombinasi khusus, bahkan pada beberapa DBMS *query* tersebut dapat dibuat menjadi suatu program batch khusus. Penggunaan *query* yang tepat akan menentukan performa, akurasi dan presisi data yang akan di retrieve.

2.2 Kluster Database

Klaster *database* merupakan kumpulan dari beberapa server *database* yang memiliki fungsi yang sama. Kumpulan server ini bisa dipandang sebagai satu kesatuan sistem *database*. Server-server *database* tersebut terhubung melalui jaringan komputer yang berkecepatan tinggi. Sebuah server *database* pada klaster dikenal dengan sebutan *node*. *Node* memiliki perangkat keras untuk mendukung komputasi (CPU, *memory*, *disk*, I/O dan lain-lain) serta perangkat lunak (sistem operasi, aplikasi dan *service*) yang bekerja secara independen[5].

Copyright © 2020 FIKOM – UNASMAN

(P) ISSN 2442-451X
(O) ISSN 2503-3832

2.3 Replikasi Database

Replikasi *database* adalah teknik mengirimkan data yang tersimpan dalam suatu *database* ke dalam *database* yang berbeda. Replikasi bertujuan untuk membentuk salinan data pada suatu sistem *database*. Pada pelaksanaannya implementasi teknik replikasi pada sistem *database* akan berbeda, tergantung pada *engine database* yang digunakan. Secara umum jenis repikasi *database* terbagi menjadi dua kelompok, yaitu replikasi *Sychronous* dan *Asychronous*[2].

1. Replikasi *Sychronous*
   a. Proses berlangsung secara real time antara *master* dan juga *slave*.
   b. Penulisan data pada penyimpanan master dan slave secara bersamaan
   c. Menyediakan data *recovery* yang konsisten

2. Replikasi *Asychronous*
   a. Replikasii data berlangsung bertahap/perbagian.
   b. Data akan ditulis pada penyimpanan master terlebih dahulu, kemudian proses replikasi dilakukan di slave.
   c. Tidak ada jaminan data aktual yang terdapat pada master akan sama dengan data pada slave.

Replikasi *database* dapat dilakukan dengan tiga cara yang berbeda, antara lain[2];

1. *Snapshot Replication*
   Melakukan salinan data pada suatu *database* ke dalam server lain secara penuh,

2. *Merging Replication*
   Menggabungkan beberapa *database* ke dalam satu *database*.

3. *Transactional Replication*
   Melakukan salinan *database* berdasarkan bagian tertentu yang mengalami perubahan

3. METODE PENELITIAN

Penelitian ini menggunakan perangkat keras (hardware) dan perangkat lunak komputer (software). Adapun kebutuhan hardware dan software yang akan digunakan dalam penelitian adalah sebagai berikut:

3.1. Perangkat keras (hardware)

Pada penelitian digunakan komputer server dengan prosesor 64 Core, Memori 126 GB dan kapasitas Hardisk sebesar 1TB. Adapun spesifikasi hardware dapat dilihat pada gambar berikut ini.

Gambar 1. Spesifikasi Perangkat Keras

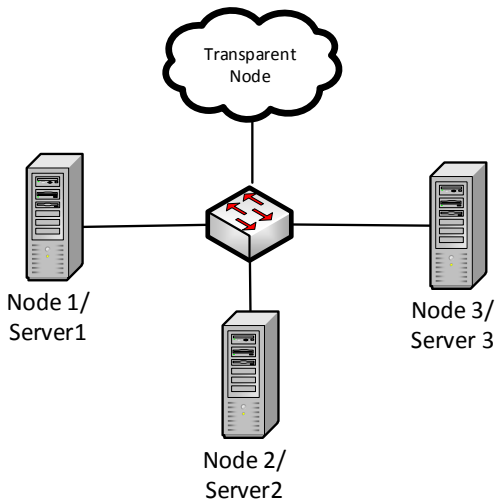
Hardware ini digunakan untuk membentuk cluster komputasi dari *database* server, Server hardware ini akan digunakan untuk menjalankan software cloud computing, seperti proxmox atau vmware.

3.2. Perangkat Lunak (Software)

Penunjang lain dalam penelitian selain perangkat keras adalah sangat dibutuhkannya perangkat lunak dalam pembuatan cluster *database*.

1. Perangkat lunak virtualisasi yang digunakan untuk membuat virtual machine adalah *Proxmox.*
2. Perangkat lunak *Ubuntu Server 16.0* sebagai aplikasi yang digunakan dalam pembuatan komputer server.
3. Pembuatan Clustering menggunakan *MySQL Cluster Database* dengan *Galera Cluster* sebagai *tools*nya.

Peneliti membuat cluster *database* menggunakan MySQL galera cluster dengan jumlah 3 buah *node* yang masing-masing *node* berperan sebagai server pada dengan Ubuntu Server 16.0 sebagai sistem operasinya. Perancangan desain jaringan dalam pembangunan multi-master *database* menggunakan galera dilakukan dengan jaringan virtual yang terhubung dengan jaringan fisik.



Gambar 2. Toplogi jaringan sistem replikasi *database*

Copyright © 2020 FIKOM – UNASMAN

(P) ISSN 2442-451X
(O) ISSN 2503-3832

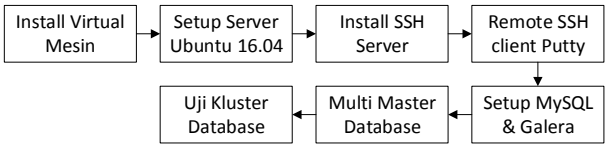
Topologi jaringan yang digunakan ditunjukan pada gambar di atas. Sistem replikasi terdiri dari 3 buah *node* yang semuanya berperan sebagai server, masing-masing server mempunyai hak untuk mereplikasi *database* dengan izin (password). Untuk pengalamatan jaringan pada sistem ini, server 1 dikenakan pengalamatan dengan IP Address 192.168.70.11/24, server 2 dengan IP Adress 192.168.70.21/24 dan server 3 dengan IP Adress 192.168.70.31/24.

Tabel 1. Rancangan Server

SERVER	PROSESOR	CPU	RAM	HARDDISK	INTERFACE
Server1/*Node1*	3.4 GHz	4	15 GB	100 GB	LOCAL
Server2/*Node2*	3.4 GHz	4	15 GB	100 GB	LOCAL
Server3/*Node3*	3.4 GHz	4	15 GB	100 GB	LOCAL

Tabel 1 menjelaskan rincian spesifikasi server, mulai dari RAM, Prosesor, Hardisk serta jaringan network yang digunakan oleh server-server *database*.

3.3. Diagram Blok



Gambar 3. Diagram Blok Jalannya Penelitian

Diagram Blok diatas menjelaskan tentang tahapan pembangunan sistem replikasi *database* yang akan dibuat, yakni;

1. Instalasi *virtual machine* pada proxmox dan buat maesin baru sebanyak 3 mesin yang akan jadi server dengan memasukan ip *static* pada setiap mesin.
2. Masing-masing server basis data akan dilakukan instalasi sistem operasi *Ubuntu Server* 16.04
3. Instalasi ssh supaya sistem bisa di-*remote* menggunakan aplikasi *Putty*.
4. Instalasi *mysql* dan *galera* pada setiap server dengan melakukan konfigurasi pada setiap server.
5. Buat kluster pada server 1 dan gabungkan server 2 dan 3 menjadi satu cluster yang sama dengan server 1
6. Setelah tergabung, uji setiap cluster untuk memastikan sudah tergabung dengan mencoba memodifiksi *database*.
7. Implementasi ke cloud dengan menggunakan ip yang disediakan dan di-*remote* melalui *putty*.

Mode replikasi yang digunakan adalah replikasi multi master. Replikasi multi master dapat membuat proses replikasi berjalan secara full duplex. Proses pertukaran data pada *database* server akan tersinkronisasi secara *realtime* pada setiap *node*. Jika salah satu *node* pada kluster *database* melakukan perubahan data (*Insert*, *Delete*, *Update*) maka perubahan tersebut akan terjadi juga pada semua *node* pada kluster tersebut.

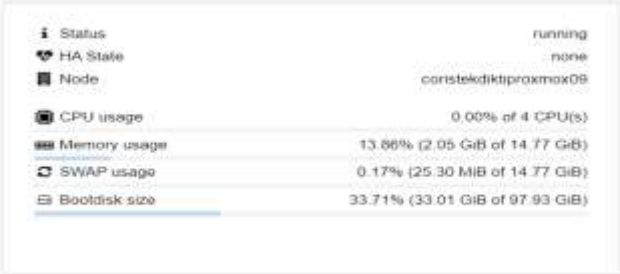
4. HASIL DAN PEMBAHASAN

Penelitian tentang replikasi *database* server dengan metode multi master telah berhasil dilakukan, replikasi telah bekerja sebagaimana mestinya. Kluster komputer dapat saling bekerja sama dalam melayani setiap *query* request dari client *database*.

Kluster server *database* terdiri dari *node-node* komputer yang saling bekerja sama dan dihubungkan melalui jaringan komputer yang berkecepatan tinggi. Adapun insfrastruktur jaringan dan server ditunjukan pada gambar berikut ini.

Gambar 4. Insfrastruktur Jaringan dan Server Penelitian

Adapun *node* yang telah terimplementasi dalam kelompok kluster *database* memiliki spesifikasi seperti pada gambar berikut ini.



Gambar 5. Spesifikasi *node* pada kluster *database*

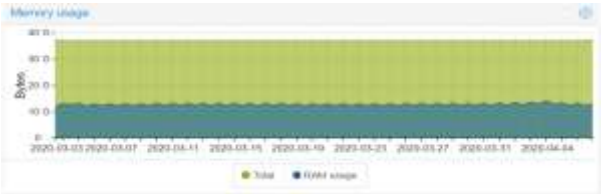
Kluster *node* akan bekerja secara mandari dan otonom selama layanan *database* bekerja, replikasi akan terus terjadi setiap ada perubahan data pada masing-masing *node* yang digunakan. Adapun penggunaan resource CPU pada kluster ditunjukan pada gambar berikut ini.

Gambar 6. Statistik CPU usage per host

Copyright © 2020 FIKOM – UNASMAN

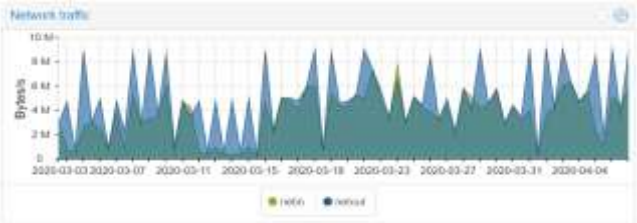
(P) ISSN 2442-451X
(O) ISSN 2503-3832

Statistik penggunaan resource CPU selama 1 bulan penelitian adalah 10 % - 45%. Pada saat proses replikasi terjadi menunjukan bahwa resource CPU tidak terbebani secara penuh untuk melakukan replikasi. Sedangkan penggunaan memori rata-rata 13 GB dengan presentase +-25%. Data penggunaan memori ditunjukan pada gambar berikut ini.



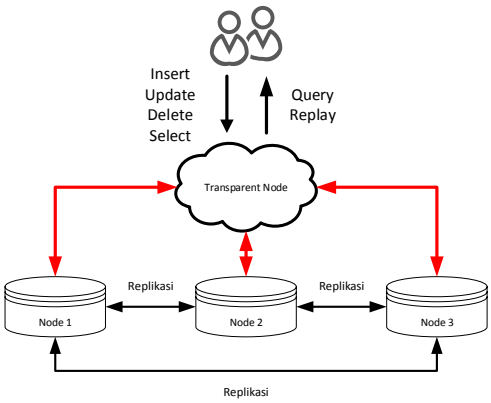
Gambar 7. Statistik Memory Usage

Proses replikasi harus ditopang oleh insfrastruktur jaringan yang handal dan berkecepatan tinggi. Pada percobaan penelitian kami menggunakan *link* fiber optic untuk menghubungkan *node-node database*. Sehingga dihasilkan kecepatan dan performa network yang terbaik. Adapun statistik penggunaan *link* komunikasi pada kluster *database* ditunjukan pada gambar berikut ini.



Gambar 8. Statistic Network Traffic

Waktu yang diperlukan untuk melakukan replikasi adalah waktu proses pengiriman data dari salah satu *node* (*insert*, *delete*, modif) kepada member kluster pada insfrastruktur *database* server. Waktu yang dibutuhkan untuk melakukan replikasi dengan 3 *node* adalah kurang dari 0.2 detik. Proses replikasi di ilustrasikan seperti gambar berikut ini.



Gambar 9. Proses Replikasi

Setiap ada perubahan data dari salah satu *node*, maka perubahan data tersebut akan langsung direplikasi kepada *node-node* server lainnya. Waktu yang diperlukan oleh masing-masing *node* untuk melakukan replikasi adalah 0.1-0.2 detik.

Tabel 2. Waktu Replikasi Master-Master

No	Kueri	*Node*1	*Node*2	*Node*3	Rata-Rata
1	*Node*1	0	0.2	0.1	0.1
2	*Node*2	0.2	0	0.1	0.1
3	*Node*3	0.1	0.2	0s	0.1
Rata-Rata		0.1	0.13	0.06	0.29\0. 3

5. KESIMPULAN

Kesimpulan yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Replikasi mode multi master dapat menjadi solusi untuk ketersediaan data.
2. Replikasi yang tersinkron dengan baik membuat semua server memiliki data yang mutakhir.
3. Layanan *database* selalu tersedia pada saat client mengirimkan *query* request pada sistem kluster.

DAFTAR PUSTAKA

[1] N. Dhanujati and A. S. Girsang, "Data Center-Disaster Recovery Center (DC-DRC) for High Availability IT Service," in *2018 International Conference on Information Management and Technology (ICIMTech)*, Sep. 2018, pp. 55–60, doi: 10.1109/ICIMTech.2018.8528170.

[2] A. Heryanto and A. Albert, "Implementasi Sistem Database Terdistribusi Dengan Metode Multi-Master Database Replication," *J. MEDIA Inform. BUDIDARMA*, vol. 3, no. 1, pp. 30–36, Mar. 2019, doi: 10.30865/mib.v3i1.1098.

[3] "Remote disaster recovery system architecture based on database replication technology - IEEE Conference Publication." https://ieeexplore.ieee.org/document/5544352 (accessed Apr. 14, 2020).

[4] R. Čerešňák and M. Kvet, "Comparison of *query* performance in relational a non-relation databases," *Transp. Res. Procedia*, vol. 40, pp. 170–177, Jan. 2019, doi: 10.1016/j.trpro.2019.07.027.

[5] "A high availability (HA) MariaDB Galera Cluster across data center with optimized WRR scheduling algorithm of LVS - TUN - IEEE Conference Publication."

Copyright © 2020 FIKOM – UNASMAN

(P) ISSN 2442-451X
(O) ISSN 2503-3832

https://ieeexplore.ieee.org/document/7440452 (accessed Apr. 14, 2020).

Copyright © 2020 FIKOM – UNASMAN