

**ANALISIS PERANCANGAN *CLUSTER REPLICATION* PADA
DATABASE POSTGRESQL MENGGUNAKAN METODE *HOT
STANDBY***

SKRIPSI

Didit Supriyadi

20210050088



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK KOMPUTER DAN DESAIN
UNIVERSITAS NUSA PUTRA
2023**

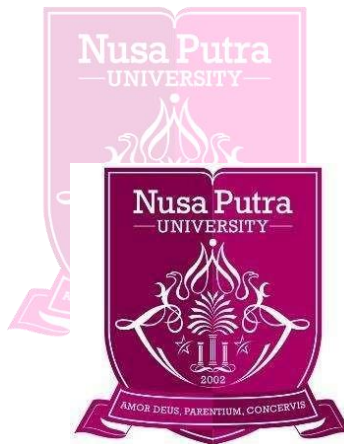
**ANALISIS PERANCANGAN *CLUSTER REPLICATION* PADA
DATABASE POSTGRESQL MENGGUNAKAN METODE *HOT
STANDBY***

SKRIPSI

*Diajukan Untuk Memenuhi Salah Satu Syarat Dalam Menempuh
Gelar Sarjana Komputer*

Didit Supriyadi

20210050088



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK KOMPUTER DAN DESAIN
UNIVERSITAS NUSA PUTRA**

2023

PERNYATAAN PENULIS

JUDUL : Analisis Perancangan *Cluster Replication* Pada *Database Postgresql* Menggunakan Metode *Hot Standby*
NAMA : Didit Supriyadi
NIM : 20210050088

“Saya menyatakan dan bertanggung jawab dengan sebenarnya bahwa Skripsi ini adalah hasil karya saya sendiri kecuali cuplikan dan ringkasan yang masing-masing telah saya jelaskan sumbernya. Jika pada waktu selanjutnya ada pihak lain yang mengklaim bahwa Skripsi ini sebagai karyanya, yang disertai dengan buktibukti yang cukup, maka saya bersedia untuk dibatalkan gelar Sarjana Komputer saya beserta segala hak dan kewajiban yang melekat pada gelar tersebut”.

Sukabumi, 4 Juli 2023



DIDIT SUPRIYADI

Penulis

PENGESAHAN SKRIPSI

JUDUL : ANALISIS PERANCANGAN CLUSTER REPLICATION
PADA DATABASE POSTGRESQL MENGGUNAKAN METODE
HOT STANDBY

NAMA : DIDIT SUPRIYADI

NIM : 20210050088

Skripsi ini telah diujikan dan dipertahankan di depan Dewan Penguji pada Sidang
Skripsi tanggal 4 Juli 2023 menurut pandangan kami, Skripsi ini memadai dari
segi kualitas untuk tujuan penganugerahan gelar Sarjana Komputer.

Sukabumi, 4 Juli 2023

Pembimbing I



Falentino Sembiring, M.Kom

NIDN: 0408029102

Pembimbing II



Cecep Warrhan, M.Kom

NIDN: 0405119501

Ketua Penguji



Habi Baturohmah, M.Kom

NIDN: 0414069701

Ketua Program Studi Sistem Informasi



Adhitia Erfina, ST., M.Kom

NIDN: 0417049102

Dekan Fakultas Teknik Komputer dan Desain

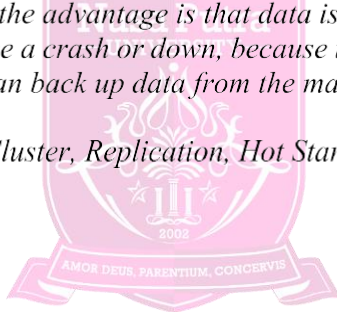
Ir. Paikun, S.T., M.T., IPM., Asean Eng

NIDN : 0402037401

ABSTRACT

Postgresql is a familiar database and is often used by application developers, because of the advantages of the Postgre database, namely because of the very large record capacity so that when carrying out large amounts of data CRUD activities, this database is prioritized. Postgre is usually used in the e-commerce industry and the banking world, however, due to the high transaction traffic from clients, it sometimes causes the main server to crash resulting in data being lost and transactions being delayed. The method used is Replication Hot Standby on the Postgresql cluster. In this method there are two servers, namely the master server as the main server, and the slave server as data backup / replica. In its design, the master server is designed as read / write, while the slave server is read-only. so that when the master node crashes or goes down, the standby server can be promoted to a new master using Manual Failover. But until now there are still many companies that have not used this method because it is considered data redundancy, even though by using the pg_basebackup method, data on the old server can be removed and new data added so that the data can be updated. The results of this study are that by implementing a replication cluster on the Postgresql database using the hot standby method there is a decrease in database performance when replicating data from master to slave because there is no division of workload on the main server, but the advantage is that data is still available even though the master server experience a crash or down, because there is a standby server that is used as a replica that can back up data from the master server in real time.

Keyword: Postgresql, Cluster, Replication, Hot Standby, Failover.



ABSTRAK

Postgresql merupakan *database* yang familiar dan sering di gunakan oleh para *developer* aplikasi , karena kelebihan dari *database postgre* yaitu karena daya tampung *record* yang sangat besar sehingga pada saat melakukan aktivitas CRUD data dalam jumlah banyak, *database* ini lebih di unggulkan. *Postgre* biasanya digunakan dalam industri *e-commerce* maupun dunia perbankan, namun karena *traffic* transaksi yang tinggi dari *client*, terkadang membuat *server* utama mengalami *crash* sehingga mengakibatkan ada data yang hilang dan transaksi pun menjadi tertunda. Metode yang digunakan yaitu *Replication Hot Standby* pada *cluster Postgresql*, Pada metode ini terdapat dua buah *server* yaitu *server master* sebagai *server* utama, dan *server slave* sebagai *backup* data / *replica*. Dalam perancangan nya, *server master* di rancang sebagai *read / write*, sedangkan *server slave* bersifat *read-only*. sehingga disaat *master node* mengalami *crash* atau *down*, maka *server standby* dapat di *promote* menjadi *master* yang baru menggunakan *Manual Failover*. Namun hingga sekarang masih banyak perusahaan yang belum menggunakan metode tersebut karena di anggap redudansi data, padahal dengan menggunakan metode *pg_basebackup*, data di *server* yang lama dapat di *remove* dan beri data baru sehingga data tersebut dapat terbaharui. Hasil dari Penelitian ini adalah dengan menerapkan *cluster* replikasi pada *database Postgresql* dengan menggunakan metode *hot standby* terdapat penurunan performa pada *database* saat melakukan *replikasi* data dari *master* ke *slave* karena tidak ada nya pembagian beban kerja pada *server* utama, namun kelebihannya yaitu data tetap tersedia walaupun *server master* mengalami *crash* atau *down*, karena ada nya sebuah *server standby* yang digunakan sebagai *replica* yang dapat melakukan *backup data* dari *server master* secara *real time*.

Kata Kunci: *Postgresql, Cluster, Replication, Hot Standby, Failover.*



KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Allah SWT, berkat rahmat dan karunia-Nya akhirnya penulis dapat menyelesaikan skripsi “ANALISIS PERANCANGAN CLUSTER REPLICATION PADA DATABASE POSTGRESQL MENGGUNAKAN METODE HOT STANDBY”. Tujuan penulisan skripsi ini diajukan untuk memenuhi salah satu syarat dalam menempuh gelar sarjana komputer.

Sehubungan dengan itu penulis menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada :

1. Bapak DR . Kurniawan, ST.,M.Si.MM selaku Rektor Universitas Nusa Putra Sukabumi.
2. Bapak Ir. Paikun, S.T., M.T., IPM., Asean Eng Selaku plt. Dekan Fakultas Teknik Komputer dan Desain Nusa Putra Sukabumi.
3. Bapak Adhitia Erfina, ST., M.Kom selaku Ketua Program Studi Sistem Informasi yang senantiasa mendukung dan memberikan arahan – arahan yang diperlukan dalam pelaksanaan dalam pengerjaan skripsi.
4. Dosen Pembimbing I Falentino Sembiring M.Kom yang telah meluangkan waktu, tenaga dan pikiran untuk memberikan arahan, dan bimbingan bagi penulis dalam penyusunan skripsi ini.
5. Dosen Pembimbing II Cecep Warman M.Kom yang telah meluangkan waktu,tenaga dan pikiran untuk memberikan arahan, dan bimbingan bagi penulis dalam penyusunan skripsi ini.
6. Seluruh Jajaran Dosen Universitas Nusa Putra yang tidak dapat disebutkan satu persatu, yang telah berjasa memberikan ilmu pengetahuannya.
7. Keluarga besar yang senantiasa membantu memberikan doa, dan memenuhi kebutuhan – kebutuhan finansial yang dibutuhkan.
8. Rekan – rekan seperjuangan di Jurusan Sistem Informasi angkatan 2019.
9. Serta masih banyak lagi pihak-pihak yang sangat berpengaruh dalam proses penyusunan skripsi yang tidak bisa penulis sebutkan satu persatu.

Dengan ini semua semoga atas izin Allah SWT semua pihak yang telah membantu proses penyelesaian skripsi ini kebaikannya Allah balas dengan pahala yang berlipat ganda.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, oleh karena itu kritik dan saran yang membangun dari berbagai pihak sangat kami harapkan demi perbaikan. Amin Yaa Rabbal 'Alamiin

Sukabumi, 4 Juli 2023

DIDIT SUPRIYADI



**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik UNIVERSITAS NUSA PUTRA , saya yang bertanda tangan di bawah ini:

Nama : Didit Supriyadi
NIM : 20210050088
Program Studi : Sistem Informasi
Jenis karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Nusa Putra Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty-Free Right) atas karya ilmiah saya yang berjudul :

**Analisis Perancangan Cluster Replication Pada Database
Postgresql Menggunakan Metode Hot Standby**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Nusa Putra berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Sukabumi

Pada tanggal : 4 Juli 2023



Didit Supriyadi

DAFTAR ISI

<u>HALAMAN SAMPUL</u>	i
<u>HALAMAN JUDUL</u>	ii
PERNYATAAN PENULIS	iii
LEMBAR PENGESAHAN PROPOSAL SKRIPSI	iv
ABSTRACT	v
ABSTRAK	vi
LEMBAR PERSETUJUAN PUBLIKASI	vii
DAFTAR ISI	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan dan Manfaat	4
1.5 Metode	4
1.6 Sistematika Penulisan	6
BAB II TINJAUAN PUSTAKA	7
2.1 Penelitian Terkait	7
2.2 Basis Data	8
2.2.1 Definisi Basis Data.....	8
2.2.2 Basis Data Operasional	9
2.2.3 Basis Data Relasional.....	10
2.3 Clustering	13
2.3.1 Data Base Cluster	15

2.3 PostgreSQL	15
2.3.1 Definisi PostgreSQL	15
2.3.2 Kelebihan PostgreSQL.....	16
2.3.3 Kekurangan PostgreSQL.....	17
2.4 Replikasi Basis Data	17
2.5 Manfaat Replikasi <i>Database</i>	18
2.6 Replikasi <i>Synchronous</i>	19
2.7 Replikasi <i>Asynchronous</i>	20
2.8 Master – Slave Replikasi.....	21
2.9 <i>Write-Ahead Logging</i>	21
2.10 DL dan DML.....	22
2.11 <i>Server</i>	22
2.12 <i>Tools</i> Pendukung.....	23
2.3.1 Pgbench.....	23
2.3.2 PgAdmin III	23
2.3.3 <i>Wireshark</i>	24
BAB III METODOLOGI PENELITIAN	26
3.1 Tahapan Penelitian	26
3.1.1 Analisis Masalah	26
3.1.2 Pemecahan Masalah	26
3.1.3 Rancangan Topologi Jaringan.....	27
3.1.4 Analisis Perangkat Lunak	27
3.1.5 Analisis Perangkat Keras	28
3.2 Rancangan Infrastruktur Jaringan	28
3.3 Alur Logical	30
3.4 Konfigurasi.....	31

3.4.1 Konfigurasi Pada <i>Node Master</i>	32
3.4.2 Konfigurasi Pada <i>Node Slave</i>	38
3.4.3 Aktivasi Replikasi	42
3.4.4 Konfigurasi <i>Failover</i>	47
BAB IV HASIL DAN PEMBAHASAN	48
4.1. Server Master	49
4.2. <i>Server Slave</i>	48
4.3. Hasil Replikasi Server	49
BAB V PENUTUP	56
5.1. Kesimpulan	56
5.2. Saran	58



DAFTAR TABEL

Tabel 1.1. Penelitian Terkait.....	3
Tabel 2.1. <i>Operational Database</i>	10
Tabel 2.2. <i>Relational Database</i>	11
Tabel 3.1. Sistem Operasi Pada Perangkat.....	30
Tabel 3.2. Analisis Perangkat Keras	30
Tabel 4.1. Hasil Pengujian Replikasi	52
Tabel 4.2. Hasil <i>Benchmark</i> dengan Replikasi	54
Tabel 4.3. Hasil <i>Benchmark Server Stand Alone</i>	55



DAFTAR GAMBAR

Gambar 1.1. Metode Penulisan Tugas Akhir	5
Gambar 2.1. Replikasi Data	19
Gambar 2.2. Replikasi <i>Synchronous</i>	21
Gambar 2.3. Replikasi <i>Asynchronous</i>	21
Gambar 3.1. Rancangan Topologi Jaringan.....	29
Gambar 3.2. Rancangan Infrastruktur Jaringan	31
Gambar 3.3. Diagram Alir Replikasi	32
Gambar 3.4. Diagram Alir <i>Failover</i>	33
Gambar 3.5. <i>Flowchart</i> Konfigurasi <i>Master</i>	34
Gambar 3.6. Instalasi Postgresql Pada <i>Node Master</i>	35
Gambar 3.7. Mengatur <i>Password</i> Postgres di <i>Node Master</i>	35
Gambar 3.8. <i>Psql Console</i>	36
Gambar 3.9. Membuat <i>User</i> untuk <i>Replication Set</i>	36
Gambar 3.10. Cek <i>User</i>	36
Gambar 3.11. Masuk ke Direktori Postgresql	37
Gambar 3.12. <i>Edit</i> Isi File Pg Hba Conf Pada <i>Master Node</i>	37
Gambar 3.13. <i>Edit Listen Address</i>	38
Gambar 3.14. <i>Edit Wal Level</i> Pada <i>Master Node</i>	38
Gambar 3.15. <i>Edit Archiving</i> Pada <i>Master Node</i>	39
Gambar 3.16. <i>Edit Replication</i> Pada <i>Master Node</i>	39
Gambar 3.17. <i>Edit Hot Standby</i> Pada <i>Master Node</i>	40
Gambar 3.18. <i>Restart Service</i> Postgresql Pada <i>Master Node</i>	40
Gambar 3.19. <i>Flowchart</i> Konfigurasi <i>Slave</i>	41
Gambar 3.20. Melakukan <i>Update</i> Pada <i>Slave</i>	41

Gambar 3.21. <i>Install Postgresql Pada Slave</i>	41
Gambar 3.22. Memberikan Password Postgresql Pada <i>Slave</i>	42
Gambar 3.23. Memberhentikan <i>Service</i> Postgresql Pada <i>Slave</i>	42
Gambar 3.24. Masuk Ke Direktori Postgresql <i>Slave</i>	42
Gambar 3.25. <i>Edit Pg Hba Conf</i> Pada <i>Slave</i>	43
Gambar 3.26. <i>Edit Postgre Conf</i> Pada <i>Slave</i>	44
Gambar 3.27. <i>Flowchart</i> Aktivasi Replikasi	44
Gambar 3.28. Masuk Ke Postgres.....	45
Gambar 3.29. Sinkronisasi <i>Master Node</i> Dengan <i>Slave1 Node</i>	45
Gambar 3.30. Masuk Ke Folder Postgresql	45
Gambar 3.31. Membuat <i>File Recovery</i>	46
Gambar 3.32. Menambahkan Konfigurasi di <i>File Recovery</i>	46
Gambar 3.33. Menyalakan <i>Service</i> Postgesql	46
Gambar 3.34. Pengecekan <i>Error</i> Pada <i>Slave</i>	47
Gambar 3.35. Pengujian Koneksi Pada <i>Master Node</i>	48
Gambar 3.36. Cek Status Pada <i>Slave</i>	48
Gambar 3.37. Konfigurasi <i>Manual Failover</i>	49
Gambar 4.1. Pembuatan <i>Database</i> Pada <i>Server Master</i>	50
Gambar 4.2. Cek Replikasi Pada <i>Slave</i>	51
Gambar 4.3. Pengujian <i>Query DDL</i> pada <i>Slave</i>	51
Gambar 4.4. Pembuatan <i>Database</i> Untuk DML	51
Gambar 4.5. <i>Insert Record</i> Pada <i>Database Master</i>	52
Gambar 4.6. Pengujian <i>Query DML</i>	52
Gambar 4.7. <i>Benchmark</i> Pada <i>Server Master</i>	53
Gambar 4.8. <i>Monitoring Server Slave</i> menggunakan <i>Htop</i>	54
Gambar 4.9. <i>Benchmark</i> Pada <i>Database</i> Replikasi	54

Gambar 4.10. Pengujian <i>Failover</i> Pada <i>Slave</i>	55
Gambar 4.11. Pengujian <i>Wireshark</i>	56



BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam dunia bisnis tentu nya membutuhkan teknologi yang dapat membantu memudahkan perusahaan guna meningkatkan produktifitas kinerja bisnis di perusahaan tersebut, namun tidak sedikit perusahaan yang mengalami kerugian yang di akibatkan karena *server* yang bermasalah karena tinggi nya trafik yang mengakses data di *server* perusahaan tersebut. Kurangnya kesiapan mengatasi data yang rusak atau hilang mengakibatkan perusahaan menjadi merugi, dan bahkan hingga saat ini hal tersebut masih banyak terjadi di seluruh dunia, hal ini karena perusahaan yang hanya menerapkan *single server* yang menjadi tumpuan bisnis mereka, sehingga tidak adanya *server* cadangan yang dapat melakukan *backup server* utama karena sedang mengalami gangguan.

Kebutuhan akan teknologi yang semakin masif dan semakin besarnya kebutuhan masyarakat akan informasi, mendorong pengembang aplikasi untuk selalu berinovasi dalam menciptakan aplikasi dengan fitur - fitur yang lebih efektif dan efisien, Media penyimpanan data pun kini sudah menjadi terstruktur dengan hadir nya *database*, sehingga informasi maupun data - data perusahaan akan tersimpan dengan rapi di dalam sistem tersebut. Namun di sisi lain dalam perkembangan teknologi masih memiliki keterbatasan apabila diintergrasikan dengan aplikasi yang berbeda.

Semakin banyaknya data yang di *input* kedalam *database*, jumlah pengakses dalam data terkadang mengakibatkan penuh nya *traffic* pada *database*, apalagi jika dibuatkan kedalam sebuah aplikasi. Jika aplikasi diakses secara terus menerus dalam sehari, maka dapat mengakibatkan data menjadi *crash* pada *database* maupun pada *server*.

Database Management System (DBMS) yaitu software yang bisa menangani kumpulan data besar dan melakukan utilitas. DBMS juga dimaksudkan untuk memfasilitasi manipulasi data. *MySQL, DB2, Oracle, SQL Server, PostgreSQL*, dan

lainnya adalah contoh DBMS. *PostgreSQL* adalah salah satu DBMS yang memberikan karakteristik yang dapat diandalkan; Meskipun demikian, ini adalah sistem data usang yang didistribusikan secara luas. Perbedaan utama antara *PostgreSQL* dan sistem manajemen basis data lainnya adalah arsitektur *postgreSQL*, yang memungkinkan pengguna untuk membangun SQL, terutama ketika mendefinisikan fungsi atau prosedur standar [1].

Dihadapkan dengan meningkatnya permintaan informasi dalam konteks global saat ini, sangat penting untuk memeriksa tuntutan dan ketersediaan tinggi (*high availability*) yang dimiliki untuk mengelola sejumlah besar permintaan. Perusahaan tidak akan dapat kehilangan klien sebagai konsekuensi dari layanan yang dibatasi, khususnya karena *downtime server*, yang hampir pasti akan terjadi dan menyebabkan kerugian finansial yang signifikan.

Pada kenyataannya, server tidak beroperasi selamanya. Kerusakan komponen perangkat keras (*hardware*) pada komputer server, software yang mengakibatkan crash, atau sistem operasi pada komputer server yang harus dimatikan untuk memperbarui dan memelihara proses mungkin menimbulkan masalah. Sebab itu, diperlukan teknologi untuk mengelola backup data sehingga data bisa dipakai kembali jika *server* utama memiliki masalah atau disusupi [2]. Di komputer, sistem replikasi *database* dikembangkan untuk menjaga ketersediaan dan kesesuaian data.

Prosedur replikasi database sering dilaksanakan pada sistem database terdistribusi. Sistem basis data terdistribusi tersusun dari banyak situs, yang masing-masing bisa berpartisipasi dalam pemrosesan transaksi yang mengakses data pada satu atau lebih lokasi. Berbagi, ketergantungan, ketersediaan, dan kecepatan pemrosesan kueri adalah beberapa alasan untuk membuat database terdistribusi. Manfaat utama dari database terdistribusi yaitu kapasitas untuk bertukar dan mengakses data secara efisien dan andal. Replikasi *master-slave* adalah salah satu cara replikasi data. Replikasi master-slave adalah teknik duplikasi *database* yang memungkinkan data disimpan di beberapa situs/node dan diperbarui oleh server database master dan *slave standby*. Dalam konfigurasi *master-slave*, master berfungsi sebagai server aktif dan meminta pembaruan dan penghapusan,

sementara *slave* berfungsi sebagai server siaga dan duplikat perubahan dari server master. [3].

Replikasi dirancang menjadi sebuah sistem yang memiliki kemampuan untuk menjamin ketersediaan layanan sehingga mengurangi resiko terjadinya *downtime* yang dapat mengakibatkan terhambatnya proses pelayanan yang dilakukan oleh *server*. Skenario sederhana dalam penerapannya adalah mesin *virtual* dikonfigurasi sebagai *Cluster* dan Replikasi data dari *master* ke *slave* bersifat *real time*. Ketika *node master* mengalami *crash* ataupun sedang melakukan *maintenance* sehingga tidak bisa diakses, maka data akan tetap tersedia di *server slave*

Berdasarkan permasalahan dan penjelasan diatas maka penulis mengambil judul **“ANALISIS PERANCANGAN *CLUSTER REPLICATION* PADA *DATABASE POSTGRESQL* MENGGUNAKAN METODE *HOT STANDBY*”**. Sebagai mitigasi atau pencegahan dari kerusakan perangkat (*hardware failure*) dan *single point of failure* pada perangkat *server database* dengan menggunakan metode *Replication Hot Standby* pada *database PostgreSQL*, sehingga ketika *server master* mengalami permasalahan yang mengakibatkan *down* atau *crash*, data tetap terjaga karena *master* melakukan *backup* data terhadap *server slave* dan *server slave* yang sebelumnya hanya bersifat *read-only* dapat di *promote* menjadi *master* baru dengan menggunakan teknik *failover*.

1.2. Rumusan Masalah

Setelah penjelasan tentang konteks masalah, rumusan masalah diperoleh yaitu::

1. Bagaimana merancang dan merancang sistem replikasi *master – slave* dengan teknik *manual failover* pada *postgresSQL* ?.
2. Bagaimana cara mengetahui kinerja pada *database Postgresql* yang di *Cluster* ?
3. Bagaimana merubah *server slave* menjadi *master* baru ketika *master* yang lama mengalami *down* atau *crash* ?

1.3. Batasan Masalah

Dari perumusan masalah yang disajikan di atas, batasan masalah berikut ditentukan:

1. *Database* yang digunakan yaitu PostgreSQL versi 9.5
2. Sistem operasi yang digunakan yaitu Ubuntu server 16.04.
3. Replikasi pada PostgreSQL menggunakan metode *Hot Standby*.
4. Pengujian dilakukan pada salah satu *node cluster* dengan mengimplementasikan SQL DDL & DML.

1.4. Tujuan dan Manfaat

1.4.1. Tujuan

Adapun tujuan dari penulisan Penelitian ini yaitu:

1. Merancang dan membangun *clustering* replikasi *master - slave* pada *database* PostgreSQL dengan menggunakan teknik *manual failover*.
2. Menerapkan Replikasi pada PostgreSQL *cluster* dengan metode *Hot Standby* guna menjaga ketersediaan data ketika *server master* sedang *down*.
3. Membuat *server slave* menjadi *master* baru ketika *master* yang lama mengalami *crash*.

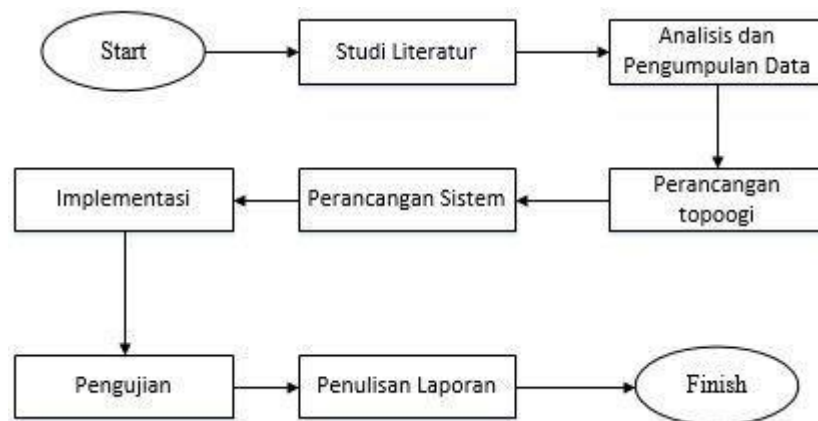
1.4.2. Manfaat

Manfaat dari penulisan penelitian ini yaitu :

1. Ketersediaan data akan tetap terjaga meskipun salah satu *server* nya mengalami *overload* atau *crash*.
2. Data yang di *cluster* di *database* tidak akan mengalami *crash* dan tidak perlu khawatir apabila *server* sedang *down*, karena data sudah di *backup* di *node* lainnya.

1.5. Metode

Metode yang dipakai dalam penulisan Penelitian ini ialah sebagai berikut:



Gambar 1.1. Metode Penulisan

1. Studi Literatur

Informasi terkait penelitian dari buku, jurnal, dan skripsi atau tesis harus dicari. Selain itu, publikasi dan berita yang berkaitan dengan penelitian tentang kinerja *Hot Standby Replication* pada database *Postgresql* memberikan informasi lebih lanjut.

2. Analisis dan Pengumpulan Data

Struktur dan arsitektur akan ditentukan dengan menganalisis data penelitian sebelumnya. Setelah melakukan analisis, data akan diolah dan dipakai sesuai dengan format standar, sesuai kebutuhan.

3. Perancangan Topologi

Perancangan terhadap infrastruktur topologi jaringan yang akan digunakan yaitu *server master* dan *server slave*.

4. Perancangan Sistem

Setelah melakukan analisa kebutuhan dan data, berikutnya dilakukan perancangan sistem dengan melakukan instalasi sistem operasi dan konfigurasi yang akan di gunakan untuk menguji terhadap sistem Replikasi *database*.

5. Implementasi

Beberapa komputer dan perangkat jaringan dibangun menjadi sebuah sistem yang menerapkan *Cluster* replikasi dengan metode *hot standby* pada *database PostgreSQL*.

6. Pengujian

Pada penelitian ini, pengujian dilaksanakan dengan melaksanakan beberapa perlakuan pada komputer *server* untuk mendapatkan nilai dari beberapa parameter yang sudah ditentukan.

7. Penulisan Laporan

Setelah melakukan pengujian ,metode terakhir yaitu penulisan laporan dimana disini akan di muat seluruh praktikum dari penelitian yang sudah dibuat

1.6. Sistematika Penulisan

Penelitian ini akan disusun menurut sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Dalam bab ini, sejarah, rumusan masalah, batasan masalah, tujuan, kelebihan, teknik, dan sistematika penulisan dibahas secara singkat.

BAB II TINJAUAN PUSTAKA

Bab ini menyoroti studi literatur teoritis yang digunakan untuk memecahkan kesulitan yang disebutkan, serta pemahaman kata-kata seperti *database*, *clustering*, *PostgreSQL*, replikasi data, dan *server*.

BAB III METODOLOGI PENELITIAN

Bab ini berisi perencanaan simulasi *Cluster* pada *database* PostgreSQL.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi mengenai hasil pengujian *Cluster* pada *database* PostgreSQL.

BAB V PENUTUP

Berisi kesimpulan temuan studi dan ide-ide untuk pengembangan lebih lanjut.

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil uraian bab-bab sebelumnya dapat dikemukakan beberapa kesimpulan dari hasil penelitian ini, yaitu :

1. Perancangan replikasi *master-slave* dengan menggunakan metode *hot – standby* bersifat *real-time*, sehingga data yang di replikasi tanpa *delay*. Dengan kondisi *server master* sebagai *server* utama untuk melakukan transaksi, dan *server slave* sebagai *server backup*, hal ini dapat membantu perusahaan Ketika adanya *high traffic* pada *server* utama maka data tetap tersedia walaupun *server master* mengalami *down* atau *crash* karena data sudah ter replikasi ke *server standby*.
2. Terdapat perbedaan performa pada saat menerapkan sistem *cluster* replikasi. Terjadi penurunan performa dibandingkan dengan *server* yang bersifat *stand alone*.
3. Saat *server master* mengalami *down* atau *crash*, *server slave* dapat di *promote* menjadi *master* yang baru. Sehingga aktivitas transaksi dari *client* pun dapat dilakukan, karena *server slave* yang sebelumnya hanya bersifat *read-only*, setelah di *promote* ,dapat menjadi *read / write*.

5.2. Saran

Dengan melihat hasil dari penelitian ini, diperoleh kesimpulan lain yaitu untuk penelitian selanjutnya dapat menjadi lebih baik dengan melakukan beberapa hal sebagai berikut :

1. Penelitian dikembangkan menjadi *High Availability*, sehingga *failover* pada *server* dilakukan secara otomatis.
2. Menggunakan sistem operasi yang lainnya karena *repository* dan *service* yang tersedia banyak yang tidak mendukung dan berbayar.
3. Metode *cluster* replikasi diimplementasikan pada jaringan yang lebih luas dan menggunakan *server* yang asli.

4. Menerapkan sistem *load balancing* sebagai pembagi beban pada *server*, sehingga tidak memberatkan *server master* yang dapat mengakibatkan *master* menjadi *crash*.



DAFTAR PUSTAKA

- [1] Nahrin Hartono, Keuntungan Penggunaan External Function Pada Database Postgresql, CSRID Journal, Vol.7 No.1 Februari 2015.
- [2] Zaenal Arifin , Joko Triyono, Rr Yuliana Rachmawati , Membangun Server Dan Analisis Backup Database Postgresql Menggunakan Teknik Replication Master/Slave, Jurnal SCRIPT Vol. 7 No. 1 Juni 2019.
- [3] Erwin Asriyar, Teten Sutendi , Implementasi Sistem Replikasi Data Basepostgresql Master-Slave Repmgr Dengan Auto Promote Masterdb, JI-Tech Vol.15 No.2 Juli 2019
- [4] Halim Maulana, “Analisis Dan Perancangan Sistem Replikasi Database Mysql Dengan Menggunakan Vmware Pada Sistem Operasi Open Source”, InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan) Vol 1, No 1, September 2016.
- [5] Tersedia : <https://dosenpintar.com/pengertian-database/> , Diakses Tanggal : 10 Februari 2023.
- [6] Halim setya Mulyantoro, “Penerapan metode load balancing cluster pada database serve guna peningkatan kinerja pengaksesan data” , Vol. IX No.1, Maret 2013.
- [7] Tersedia : <https://dosenit.com/kuliah-it/database/macam-macam-database> , Diakses Tanggal : 10 Februari 2023.
- [8] Tersedia : <https://www.tutorialpedia.net/apa-itu-postgresql-dan-sejarah-perkembangannya/> , Diakses Tanggal : 10 Februari 2023.
- [9] Tersedia : <https://www.nesabamedia.com/pengertian-postgresql/> , Diakses Tanggal : 10 Februari 2023
- [10] Eddy Purwanto "Perbandingan Strategi replikasi pada sistem basis data terdistribusi".
- [11] Tersedia : <https://www.enterprisedb.com/postgres-tutorials/postgresql-replication-and-automatic-failover-tutorial/> , Diakses Tanggal : 24 Februari 2023.
- [12] Mohammad Hindam Adli, "Restrukturisasi Replikasi Database Menggunakan Mapping Rules Untuk Pengiriman Data Penyakit di Dinkes Kota Semarang".
- [13] Royan Chadzami, “Implementasi Sistem Penjualan Alat Olah Raga Menggunakan Basis Data Terdistribusi Dengan Metode Replikasi Asynchronous”.

- [14] Tersedia : <https://www.nesabamedia.com/pengertian-ddl-dan-dml/> ,
Diakses Tanggal : 24 Febuari 2023.
- [15] Tersedia : <https://idwebhost.com/blog/apa-yang-dimaksud-dengan-server/> ,
Diakses Tanggal : 24 Febuari 2023.
- [16] Tersedia : <https://medium.com/@FranckPachot/do-you-know-what-you-are-measuring-with-pgbench-d8692a33e3d6> , Diakses Tanggal : 24 Febuari 2023.
- [17] Tersedia : <http://www.kerjanya.net/faq/12075-unduh-pgadmin-iii-gratis-download-pgadmin-iii.html> , Diakses Tanggal : 24 Febuari 2023

