JUTIS (Jurnal Teknik Informatika Unis) **p-ISSN: 2252-5351**
Vol. 14 No. 1 Bulan April 2026 **e-ISSN: 2656-0860**
Program Studi Teknik Informatika – Fakultas Teknik Universitas Islam Syekh-Yusuf Tangerang

IMPLEMENTASI REPLIKASI DATABASE MYSQL MASTER TO MASTER DENGAN SIMULASI KETAHANAN SISTEM TERHADAP GANGGUAN SERVER

Muhammad Ilham Al Fasya[1,*], Jemakmun[2]

[1,2]Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Bina Darma, Jl. Jenderal Ahmad Yani No. 3, Palembang, Sumatera Selatan 30111

**E-mail korespondensi: ilhamfasya13@gmail.com*

Abstrak

Ketersediaan dan keandalan data merupakan aspek yang sangat penting dalam mendukung kelangsungan operasional sistem informasi, termasuk pada lingkungan industri seperti PT Pupuk Sriwidjaja Palembang. Gangguan pada server basis data dapat menimbulkan waktu henti yang berdampak pada kerugian operasional dan reputasi perusahaan, baik yang disebabkan oleh kegagalan perangkat keras maupun insiden keamanan siber. Penelitian ini bertujuan untuk mengimplementasikan dan mengevaluasi efektivitas replikasi basis data MySQL dengan skema master to master dalam meningkatkan ketahanan sistem terhadap gangguan server, serta mengevaluasi strategi backup sebagai lapisan mitigasi tambahan terhadap risiko query destruktif yang turut tereplikasi ke seluruh master. Metode yang digunakan adalah eksperimen terapan yang meliputi konfigurasi dua server sebagai master ganda, pengujian fungsi replikasi berupa pembuatan basis data, penyisipan data, dan pembaruan data pada kedua arah, simulasi gangguan dengan mematikan salah satu server, serta skenario backup dan pemulihan yang mensimulasikan query penghapusan data tanpa filter pada salah satu master. Hasil pengujian menunjukkan bahwa replikasi dua arah berjalan stabil dan konsisten, sistem tetap mampu melayani permintaan basis data selama salah satu server tidak aktif, dan mampu melakukan sinkronisasi ulang berbasis binary log tanpa kehilangan data setelah server pulih. Namun demikian, pengujian juga membuktikan bahwa replikasi master to master tanpa filtering konten turut mereplikasi query destruktif ke seluruh master, sehingga seluruh data dapat hilang secara bersamaan; risiko ini berhasil dimitigasi melalui strategi backup terjadwal yang terbukti mampu memulihkan ketersediaan data secara penuh pada kedua master. Temuan ini mengindikasikan bahwa kombinasi replikasi master to master dan strategi backup merupakan solusi yang lebih menyeluruh untuk mendukung ketersediaan tinggi (high availability) dan pemulihan bencana (disaster recovery) pada sistem basis data berbasis MySQL di lingkungan operasional nyata.

Kata kunci: *backup dan pemulihan, high availability, ketahanan sistem, master to master, replikasi database.*

A. Pendahuluan

Dalam era digital saat ini, ketersediaan dan keandalan data menjadi faktor penentu keberlangsungan operasional berbagai sektor, mulai dari *e-commerce*, perbankan, hingga industri manufaktur seperti PT Pupuk Sriwidjaja Palembang yang mengandalkan sistem informasi berbasis basis data dalam menunjang proses bisnisnya. Gangguan pada server basis data dapat menyebabkan waktu henti (*downtime*) yang signifikan dan berdampak pada kerugian finansial maupun reputasi perusahaan. Oleh karena itu, strategi peningkatan ketahanan sistem terhadap gangguan menjadi fokus utama dalam pengelolaan infrastruktur teknologi informasi.

Salah satu pendekatan yang efektif untuk meningkatkan ketersediaan data adalah replikasi basis data. Replikasi master to master pada MySQL memungkinkan dua server saling mereplikasi data secara bersamaan, sehingga apabila salah satu server mengalami gangguan, server lainnya tetap dapat melanjutkan layanan tanpa interupsi berarti. Meskipun demikian, penerapan skema ini tidak lepas dari tantangan teknis, seperti kompleksitas konfigurasi dan potensi konflik data yang memerlukan penanganan khusus.

Sejumlah penelitian terdahulu telah mengkaji penerapan replikasi basis data untuk meningkatkan keandalan sistem. Shrestha (2017) membandingkan replikasi master-slave tradisional dengan solusi cluster modern seperti Galera Cluster, dan menemukan bahwa meskipun replikasi master-slave unggul dalam throughput dan waktu respons pada pengujian baca-tulis, solusi cluster lebih unggul dalam ketersediaan tinggi dan konsistensi data pada skenario kegagalan node. Akhlak Kamiswara dan Wijaya (2023) mengimplementasikan replikasi MySQL master to master untuk menjamin ketersediaan data pada server, namun belum menyertakan pengujian kuantitatif terhadap waktu pemulihan pasca gangguan. Kristian, Rochim, dan Widianto (2015) mengembangkan sistem replikasi dan redundansi untuk basis data MySQL, sedangkan Maulana (2016) menerapkan replikasi MySQL berbasis VMware untuk meningkatkan ketahanan sistem, keduanya berfokus pada aspek konfigurasi tanpa pengujian skenario failover yang realistis. Suhatman, Rida, Surya, dan Tasya (2024) menerapkan disaster *recovery* plan pada MySQL menggunakan metode *master to master* berbasis Docker dan melaporkan keberhasilan pemulihan otomatis, tetapi tidak mengukur secara kuantitatif waktu sinkronisasi maupun proses failover. Sholihah dan Darujati (2022) mengkaji replikasi basis data MySQL menggunakan container Docker dengan hasil yang stabil, sementara Kurniadi, Djusmin, dan Yasir (2021) menerapkan replikasi master-slave pada sistem informasi wisuda yang berhasil menjaga sinkronisasi data secara real-time namun belum mendukung skema dual-master maupun pengujian failover. Pada konteks yang lebih konseptual, Suprianto (2019) menyusun disaster recovery plan berbasis NIST SP 800-34 di lingkungan industri pupuk, namun kajiannya berfokus pada aspek manajerial tanpa menyentuh implementasi teknis replikasi basis data. Rangkuman kajian penelitian terdahulu tersebut disajikan pada Tabel 1.

Tabel 1. Ringkasan Penelitian Terdahulu

No	Peneliti	Tahun	Topik Penelitian	Metode	Temuan Utama	Kelemahan/Gap
1	Suhatman, dkk.	2024	Replikasi MySQL master-to-master berbasis Docker dalam DRP	Simulasi Docker, query test	Replikasi 100% berhasil, pemulihan otomatis	Tidak mengukur waktu sinkronisasi dan failover
2	Ajie & Sutabri	2025	Optimasi basis data terdistribusi dengan replikasi dan partisi (NoSQL)	Simulasi JMeter, MongoDB, Cassandra	Throughput naik 42%, latensi turun 30%	Bukan MySQL, tidak diuji failover
3	Maulana	2016	Replikasi MySQL dengan VMware untuk peningkatan ketahanan sistem	Konfigurasi virtualisasi	Sistem stabil, layanan tetap berjalan	Tidak diuji failover atau pemulihan otomatis

No	Peneliti	Tahun	Topik Penelitian	Metode	Temuan Utama	Kelemahan/Gap
4	Suprianto	2019	DRP berbasis NIST SP 800-34 di PT Pupuk Kaltim	Studi kasus manajerial	Skema DRP terstruktur dan lengkap	Tidak menyentuh aspek teknis replikasi basis data
5	Kurniadi, dkk.	2021	Replikasi master-slave pada sistem informasi wisuda	Implementasi MySQL	Sinkronisasi data real-time berhasil	Belum mendukung dual-master dan pengujian failover

Dari sisi teknis, platform MySQL turut mengalami sejumlah pembaruan yang relevan bagi konteks replikasi dan ketersediaan tinggi. MySQL 8.0 menghadirkan data dictionary terintegrasi, dukungan kunci enkripsi bawaan, serta peningkatan pada mekanisme replikasi multithread yang memungkinkan efisiensi lebih tinggi dalam proses sinkronisasi antar server, termasuk dukungan terhadap skema replikasi asinkron, semi-sinkron, maupun grup replikasi (Oracle Corporation, 2024). Ketersediaan fitur-fitur tersebut menjadikan MySQL sebagai basis yang relevan untuk dikaji lebih lanjut dalam konteks replikasi master to master, khususnya pada aspek ketahanannya terhadap gangguan operasional di lingkungan industri.

Berdasarkan kajian terhadap penelitian-penelitian tersebut, ditemukan bahwa mayoritas studi masih terbatas pada skenario replikasi satu arah, perencanaan disaster recovery secara konseptual, atau penerapan pada platform non-MySQL, serta belum banyak yang secara langsung menguji ketahanan sistem terhadap gangguan server dalam konteks replikasi master to master berbasis MySQL secara kuantitatif dan pada lingkungan operasional nyata. Kebaruan penelitian ini terletak pada integrasi antara implementasi skema replikasi dua arah (master to master) dengan pengujian failover secara langsung pada dua server aktif di lingkungan industri, disertai evaluasi terhadap tiga aspek operasional sekaligus, yaitu keberhasilan replikasi data, kontinuitas layanan selama gangguan, dan keberhasilan sinkronisasi ulang pascapemulihan.

Berdasarkan uraian tersebut, permasalahan yang dikaji dalam penelitian ini meliputi bagaimana proses konfigurasi replikasi master to master pada MySQL dilakukan secara teknis, bagaimana pengaruhnya terhadap ketahanan sistem ketika terjadi gangguan server, serta seberapa efektif skema ini dalam mempertahankan kontinuitas layanan saat salah satu server mengalami kegagalan. Dengan demikian, penelitian ini bertujuan untuk mengimplementasikan konfigurasi replikasi database MySQL master to master pada lingkungan server ganda, menguji ketahanan sistem terhadap berbagai skenario gangguan server, serta mengevaluasi efektivitas skema replikasi tersebut dalam mendukung ketersediaan tinggi dan pemulihan bencana pada sistem basis data operasional.

B. Metode

Penelitian ini menggunakan pendekatan eksperimen terapan (applied experiment) yang dilaksanakan di Departemen Teknologi Informasi PT Pupuk Sriwidjaja Palembang selama periode magang Merdeka Belajar Kampus Merdeka (MBKM) berlangsung. Objek penelitian adalah dua unit server berbasis sistem operasi Ubuntu yang dikonfigurasi sebagai

node master ganda (dual master) dalam skema replikasi MySQL. Selain MySQL sebagai sistem manajemen basis data utama, penelitian ini juga memanfaatkan phpMyAdmin untuk verifikasi struktur dan isi basis data secara visual pada masing-masing node, serta Nginx sebagai web server pendukung dalam simulasi lingkungan layanan berbasis basis data.

Tahapan penelitian dilaksanakan melalui beberapa langkah utama. Pertama, dilakukan instalasi dan konfigurasi MySQL pada kedua server, mencakup pengaturan berkas my.cnf dengan penetapan server-id yang unik pada masing-masing node, pengaktifan binary logging, serta penyesuaian parameter agar setiap server mampu berperan sebagai master sekaligus slave terhadap satu sama lain. Kedua, dilakukan pembuatan akun pengguna khusus untuk keperluan replikasi pada kedua server, dilanjutkan dengan konfigurasi arah replikasi menggunakan perintah change master to yang memuat alamat IP server tujuan, kredensial pengguna replikasi, nama berkas binary log, dan posisi log. Proses replikasi kemudian diaktifkan melalui perintah start slave pada kedua server secara bergantian. Ketiga, dilakukan verifikasi awal terhadap status replikasi menggunakan perintah show slave status untuk memastikan parameter *Slave_IO_Running* dan *Slave_SQL_Running* bernilai Yes serta *Last_Error* bernilai 0 pada kedua node.

Setelah replikasi dinyatakan berjalan normal, dilakukan pengujian fungsional replikasi meliputi operasi pembuatan basis data (create database), penyisipan data (insert), dan pembaruan data (update) yang dijalankan secara bergantian dari kedua arah server untuk memastikan konsistensi replikasi dua arah. Tahap berikutnya adalah simulasi gangguan server yang dilakukan dengan mematikan salah satu server secara sengaja, kemudian melakukan pengujian konektivitas antarserver menggunakan perintah ping, menjalankan operasi basis data pada server yang masih aktif, serta melakukan verifikasi terhadap ketersediaan dan integritas data. Pengujian diakhiri dengan menyalakan kembali server yang sebelumnya nonaktif untuk mengamati proses sinkronisasi ulang (catch-up replication) berbasis binary log secara otomatis.

Sebagai tindak lanjut atas masukan reviewer bahwa gangguan server pada implementasi aktual tidak hanya disebabkan oleh kegagalan perangkat keras, tetapi juga dapat dipicu oleh insiden keamanan siber, seperti kredensial yang disalahgunakan atau serangan yang menjalankan perintah penghapusan data secara masif, penelitian ini menambahkan satu skenario pengujian lanjutan berupa strategi backup dan pemulihan (backup and recovery). Pengujian ini bertujuan mengevaluasi risiko bahwa replikasi master to master mereplikasi seluruh perintah data manipulation language tanpa filtering konten, sehingga sebuah query destruktif yang dijalankan pada satu master, baik akibat kesalahan maupun tindakan berbahaya, akan turut tereplikasi dan menghapus data pada master pasangannya secara bersamaan. Skenario pengujian dirancang melalui lima tahap, yaitu: (1) pengambilan cadangan data logis (logical backup) secara terjadwal menggunakan mysqldump dengan opsi single-transaction pada salah satu master untuk merepresentasikan titik pemulihan (recovery point) terakhir; (2) simulasi query destruktif tanpa filter berupa penghapusan seluruh baris data (delete tanpa klausa where) pada Master 1 untuk merepresentasikan skenario insiden keamanan; (3) pengamatan terhadap propagasi otomatis operasi tersebut ke Master 2 melalui mekanisme replikasi; (4) pemulihan data pada kedua master menggunakan berkas backup yang telah diambil, dengan menghentikan sementara

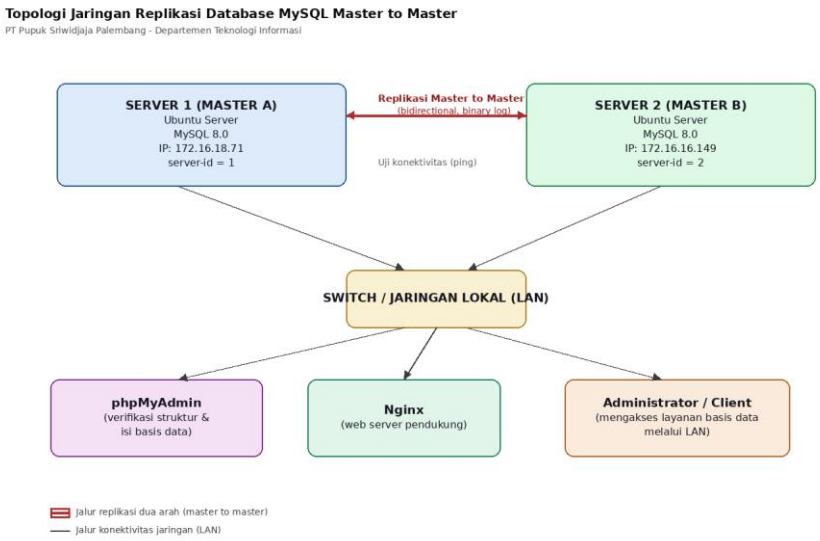
proses replikasi pada salah satu node untuk mencegah penerapan ulang perintah usang; dan (5) penyelarasan ulang posisi binary log serta pengujian penyisipan data baru untuk memastikan replikasi kembali berjalan normal pascapemulihan.

Analisis data dilakukan secara deskriptif kualitatif dengan mengamati luaran perintah verifikasi MySQL pada setiap tahap pengujian, meliputi status replikasi, konsistensi data antarnode, serta keberhasilan maupun kegagalan pada setiap skenario. Keberhasilan replikasi diukur berdasarkan kesesuaian data antara kedua server setelah setiap operasi manipulasi data, ketahanan sistem dievaluasi berdasarkan kemampuan server yang masih aktif untuk tetap melayani permintaan basis data selama server mitranya mengalami gangguan, dan efektivitas strategi backup dievaluasi berdasarkan kelengkapan serta konsistensi data yang berhasil dipulihkan pada kedua master setelah skenario penghapusan data disimulasikan.

Tabel 2. Spesifikasi Lingkungan Pengujian

No	Komponen	Spesifikasi
1	Sistem operasi server	Ubuntu Server (Server 1 dan Server 2)
2	Sistem manajemen basis data	MySQL 8.0, dikonfigurasi sebagai dual master
3	Antarmuka manajemen basis data	phpMyAdmin, untuk verifikasi struktur dan isi basis data
4	Web server pendukung	Nginx
5	Konektivitas jaringan	Jaringan lokal (LAN) Departemen Teknologi Informasi PT Pupuk Sriwidjaja Palembang
6	Skema replikasi	Master to master (bidirectional, dua node)

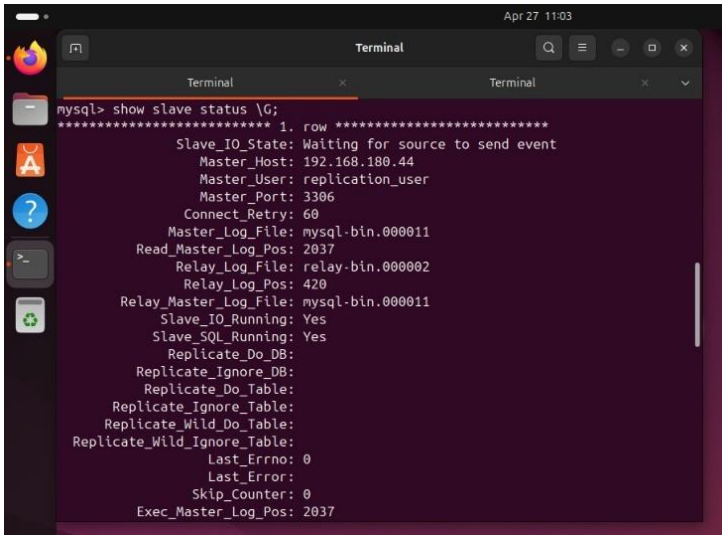
Untuk memudahkan pemahaman terhadap arsitektur penelitian, topologi jaringan yang menghubungkan Server 1 dan Server 2 beserta komponen pendukungnya ditampilkan pada Gambar 1. Topologi tersebut menggambarkan interkoneksi kedua server melalui replikasi dua arah (bidirectional), serta keterhubungan keduanya dengan phpMyAdmin, Nginx, dan jaringan lokal (LAN) pada Departemen Teknologi Informasi PT Pupuk Sriwidjaja Palembang.



Gambar 1. Topologi jaringan replikasi database MySQL master to master

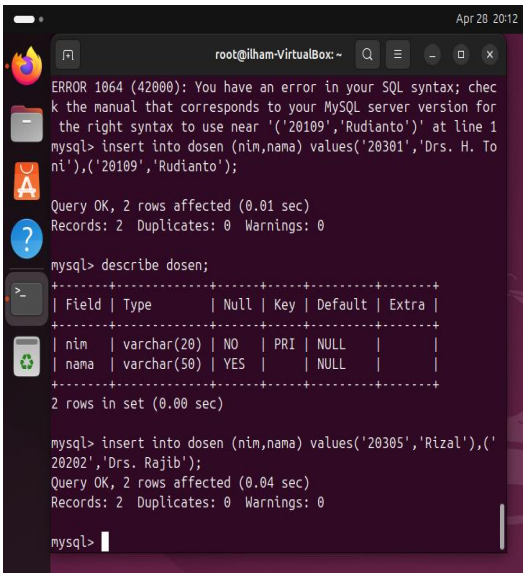
C. Hasil dan Pembahasan

Implementasi replikasi master to master**.** Konfigurasi replikasi diawali dengan penyesuaian berkas my.cnf pada server pertama, mencakup penetapan server-id, pengaktifan log_bin, dan penentuan basis data yang direplikasi. Konfigurasi serupa diterapkan pada server kedua dengan server-id yang berbeda untuk membedakan identitas kedua node dalam skema replikasi. Setelah kedua server direstart, dilakukan pembuatan akun pengguna replikasi pada masing-masing server, kemudian koneksi replikasi diatur secara bersilangan menggunakan perintah change master to yang menunjuk pada server pasangannya, dan proses replikasi diaktifkan melalui perintah start slave. Hasil verifikasi menggunakan perintah show slave status pada Gambar 2 menunjukkan bahwa parameter Slave_IO_Running dan Slave_SQL_Running bernilai Yes pada kedua server, dengan Last_Error tercatat 0, yang mengonfirmasi bahwa koneksi replikasi berjalan tanpa kendala dan proses sinkronisasi data berlangsung normal.



Gambar 2. Verifikasi status replikasi pada server master (show slave status)

Pengujian fungsi replikasi**.** Pengujian dilakukan terhadap tiga jenis operasi basis data, yaitu pembuatan basis data baru, penyisipan data, dan pembaruan data, yang masing-masing dijalankan dari kedua arah server untuk memastikan sifat bidirectional dari replikasi. Basis data baru yang dibuat pada Server 1 secara otomatis muncul pada Server 2 dalam hitungan detik setelah perintah create database dijalankan, demikian pula sebaliknya. Pengujian penyisipan data dilakukan dengan membuat tabel dosen serta mahasiswa dan menambahkan baris data baru pada masing-masing server; hasil verifikasi menunjukkan bahwa seluruh data yang disisipkan pada satu server tampil secara identik pada server pasangannya. Pengujian pembaruan data pada tabel dosen juga menunjukkan hasil yang konsisten, di mana perubahan nilai kolom pada satu server langsung tercermin pada server lainnya tanpa terjadi konflik maupun kehilangan data. Sebagai ilustrasi, Gambar 3 menunjukkan hasil penyisipan dua baris data ke tabel dosen beserta verifikasi struktur tabel menggunakan perintah describe, yang menegaskan bahwa struktur maupun isi tabel tetap konsisten setelah proses replikasi. Ringkasan hasil pengujian fungsional disajikan pada Tabel 3.



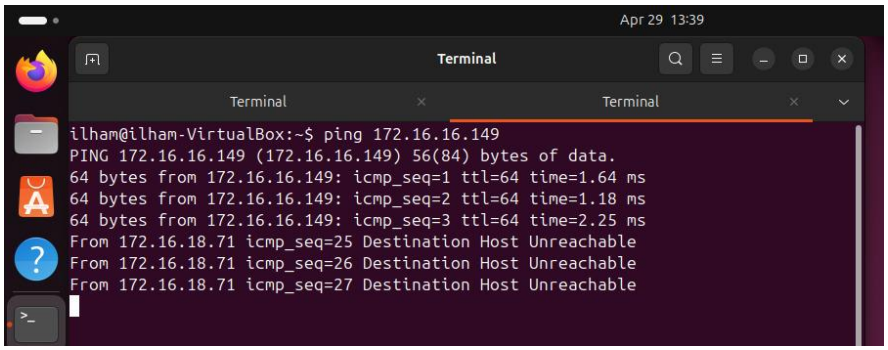
Gambar 3. Penyisipan data pada tabel dosen dan verifikasi struktur tabel

Tabel 3. Ringkasan Hasil Pengujian Fungsional Replikasi Master to Master

No	Jenis Pengujian	Arah Pengujian	Hasil
1	Pembuatan basis data (create database)	Server 1 → Server 2	Berhasil, tereplikasi otomatis
2	Pembuatan basis data (create database)	Server 2 → Server 1	Berhasil, tereplikasi otomatis
3	Penyisipan data (insert)	Server 1 → Server 2	Berhasil, data identik pada kedua server
4	Penyisipan data (insert)	Server 2 → Server 1	Berhasil, data identik pada kedua server
5	Pembaruan data (update)	Server 1 → Server 2	Berhasil, tanpa konflik data
6	Pembaruan data (update)	Server 2 → Server 1	Berhasil, tanpa konflik data
7	Operasi basis data saat server pasangan mati	Server aktif (mandiri)	Berhasil, data tersimpan lokal
8	Sinkronisasi ulang pascapemulihan server	Kedua arah	Berhasil, tanpa kehilangan data

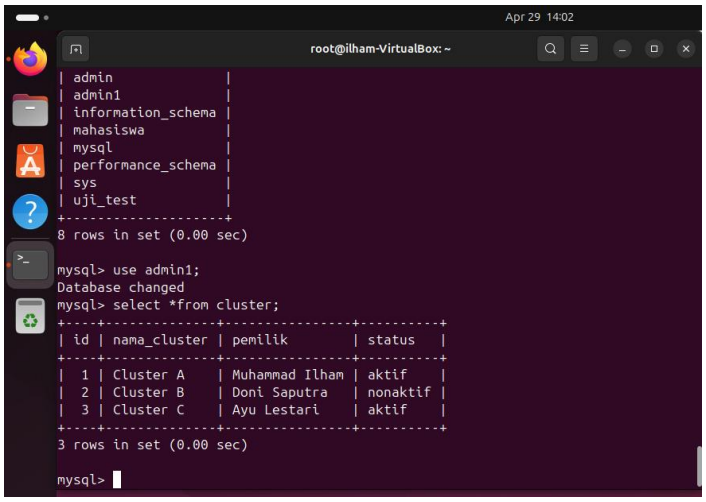
Seluruh skenario pengujian pada Tabel 3 menunjukkan hasil yang konsisten "berhasil" pada kedua arah replikasi, yang mengindikasikan bahwa mekanisme replikasi master to master pada MySQL mampu menjaga konsistensi data secara real-time untuk operasi data definition language (DDL) maupun data manipulation language (DML) tanpa memerlukan intervensi manual.

Simulasi gangguan server**.** Simulasi ketahanan sistem dilakukan dengan mematikan salah satu server secara sengaja untuk merepresentasikan skenario gangguan pada lingkungan nyata. Pengujian konektivitas menggunakan perintah ping dari Server 1 menunjukkan bahwa Server 2 semula dapat dijangkau, namun setelah dimatikan, permintaan ping menghasilkan pesan destination host unreachable sebagaimana ditunjukkan pada Gambar 4, yang mengonfirmasi bahwa server tersebut benar-benar tidak dapat diakses melalui jaringan. Pengujian serupa dari arah sebaliknya juga menunjukkan hasil yang konsisten ketika Server 1 dimatikan.



Gambar 4. Hasil pengujian konektivitas (ping) saat salah satu server dimatikan

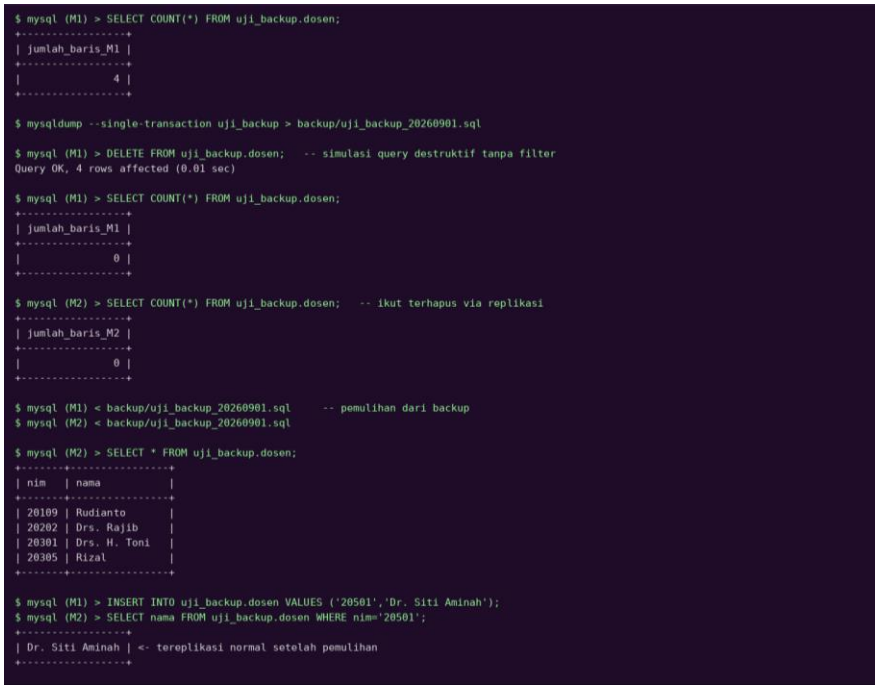
Selama salah satu server dalam kondisi nonaktif, operasi basis data tetap dijalankan pada server yang masih aktif, meliputi pembuatan tabel baru dan penyisipan data. Hasil verifikasi menunjukkan bahwa seluruh operasi tersebut berhasil dilakukan dan data tersimpan secara lokal tanpa error, membuktikan bahwa server yang tersisa mampu melanjutkan layanan basis data secara mandiri selama gangguan berlangsung. Setelah server yang sebelumnya nonaktif dinyalakan kembali, sistem replikasi secara otomatis membaca binary log dari server pasangannya dan menjalankan seluruh perintah yang tertunda selama periode downtime. Gambar 5 menunjukkan bahwa data yang disisipkan selama salah satu server tidak aktif berhasil tersinkronisasi sepenuhnya setelah server tersebut pulih, tanpa memerlukan intervensi manual maupun terjadinya kehilangan data.



Gambar 5. Verifikasi data hasil sinkronisasi otomatis setelah server pulih

Skenario backup dan pemulihan**.** Sesuai dengan masukan reviewer bahwa gangguan server pada implementasi aktual tidak hanya bersumber dari kegagalan perangkat keras, tetapi juga dapat dipicu oleh insiden keamanan siber, penelitian ini menambahkan pengujian skenario backup dan pemulihan. Pengujian diawali dengan pengambilan cadangan data logis (mysqldump) pada tabel dosen di Master 1, yang saat itu berisi empat baris data konsisten pada kedua master hasil pengujian fungsional sebelumnya. Selanjutnya disimulasikan sebuah query destruktif berupa perintah delete tanpa klausa where pada Master 1, yang merepresentasikan skenario insiden keamanan seperti kredensial yang disalahgunakan atau serangan siber yang menjalankan perintah penghapusan data secara masif. Hasil pengujian pada Gambar 6 menunjukkan bahwa operasi tersebut langsung tereplikasi ke Master 2 tanpa filtering, sehingga kedua master kehilangan seluruh data secara bersamaan (0 baris pada

kedua node). Temuan ini membuktikan kekhawatiran reviewer bahwa replikasi master to master tanpa mekanisme filtering konten tidak memberikan proteksi terhadap query destruktif yang berasal dari salah satu sisi, sehingga tidak dapat diandalkan sebagai satu-satunya mekanisme ketahanan sistem.



Gambar 6. Simulasi penghapusan data tanpa filter dan pemulihan menggunakan backup

Selanjutnya, data dipulihkan pada kedua master menggunakan berkas backup yang telah diambil sebelumnya, dengan terlebih dahulu menghentikan proses replikasi pada Master 2 guna mencegah penerapan ulang transaksi usang. Setelah proses pemulihan selesai, seluruh empat baris data awal berhasil dikembalikan secara identik pada kedua master. Posisi binary log kemudian diselaraskan ulang ke koordinat terkini, dan pengujian penyisipan data baru menunjukkan bahwa replikasi dua arah kembali berfungsi normal tanpa kendala pascapemulihan. Ringkasan tahapan dan hasil pengujian skenario ini disajikan pada Tabel 4.

Tabel 4. Tahapan dan Hasil Pengujian Skenario Backup dan Pemulihan

No	Tahapan Pengujian	Hasil
1	Pengambilan backup logis (mysqldump) pada Master 1 sebagai recovery point	Berhasil, backup memuat 4 baris data tabel dosen
2	Simulasi query destruktif (delete tanpa where) pada Master 1	Seluruh baris data terhapus pada Master 1 (0 baris)
3	Pengamatan propagasi otomatis ke Master 2 melalui replikasi	Data pada Master 2 turut terhapus (0 baris), tanpa filtering
4	Pemulihan data pada kedua master dari berkas backup	Berhasil, 4 baris data kembali identik pada kedua master
5	Penyelarasan ulang posisi binary log dan uji replikasi lanjutan	Berhasil, data baru tereplikasi normal ke kedua arah

Pembahasan**.** Secara praktis, hasil penelitian ini relevan bagi pengelolaan infrastruktur basis data di PT Pupuk Sriwidjaja Palembang maupun instansi lain dengan kebutuhan

serupa, mengingat gangguan pada satu server tidak lagi berarti terhentinya layanan basis data secara keseluruhan selama server pasangannya masih beroperasi. Skema ini memberikan alternatif yang relatif terjangkau dari sisi kompleksitas dan kebutuhan sumber daya dibandingkan solusi cluster penuh, sehingga cocok diterapkan pada lingkungan dengan jumlah node terbatas namun tetap membutuhkan tingkat ketersediaan yang tinggi.

Temuan ini memperkuat hasil penelitian Akhlak Kamiswara dan Wijaya (2023) maupun Suhatman, dkk. (2024) yang juga melaporkan keberhasilan replikasi master to master dalam menjaga ketersediaan data, namun penelitian ini memberikan kontribusi tambahan berupa pengujian langsung terhadap skenario failover pada dua server aktif secara berurutan, yang belum banyak dilakukan pada penelitian sejenis, termasuk pada studi Kurniadi, dkk. (2021) yang masih terbatas pada skema satu arah. Dibandingkan dengan temuan Shrestha (2017) yang menunjukkan keunggulan solusi cluster dalam skenario kegagalan node, hasil penelitian ini menegaskan bahwa replikasi master to master pada MySQL, meskipun bukan solusi cluster penuh, tetap mampu memberikan tingkat ketahanan yang memadai untuk skenario kegagalan satu node pada lingkungan dengan dua server, dengan kompleksitas implementasi yang relatif lebih sederhana. Mekanisme catch-up replication berbasis binary log yang bekerja otomatis setelah server pulih juga menunjukkan keselarasan dengan prinsip disaster recovery yang dikemukakan oleh Suprianto (2019), meskipun penelitian tersebut hanya membahasnya pada tataran konseptual manajerial tanpa pengujian teknis.

Sebagaimana ditunjukkan pada skenario backup dan pemulihan, gangguan server pada implementasi nyata tidak hanya bersumber dari kegagalan perangkat keras sebagaimana disimulasikan pada bagian sebelumnya, tetapi juga dapat berasal dari insiden keamanan siber, termasuk penyalahgunaan kredensial maupun serangan yang menjalankan perintah destruktif pada basis data. Karena replikasi master to master mereplikasi seluruh perubahan data tanpa mekanisme filtering konten, satu query berbahaya yang dijalankan pada salah satu master berpotensi menghapus data pada seluruh master secara bersamaan, sehingga replikasi semata tidak dapat diandalkan sebagai satu-satunya mekanisme ketahanan sistem. Penambahan strategi backup terjadwal, sebagaimana diuji pada penelitian ini, terbukti mampu memulihkan ketersediaan data secara penuh setelah insiden semacam ini terjadi, sehingga memperkuat kontribusi penelitian dalam mendukung high availability secara lebih menyeluruh dan menjawab keterbatasan pendekatan replikasi murni yang diangkat oleh reviewer. Untuk implementasi produksi, strategi ini idealnya dilengkapi dengan mekanisme tambahan seperti delayed replica atau retensi binary log yang lebih panjang guna mendukung pemulihan pada titik waktu tertentu (point-in-time recovery), pembatasan hak akses eksekusi perintah destruktif skala besar, serta sistem pemantauan yang mampu mendeteksi anomali pola query secara dini sebelum tereplikasi ke seluruh node.

Meskipun demikian, penelitian ini memiliki keterbatasan, yaitu pengujian dilakukan pada skala dua server dalam kondisi simulasi laboratorium sehingga belum sepenuhnya merepresentasikan kompleksitas trafik data pada lingkungan produksi berskala besar, serta belum dilakukan pengukuran kuantitatif terhadap durasi pemulihan (recovery time), frekuensi ideal penjadwalan backup, maupun potensi konflik data pada skenario penulisan bersamaan (concurrent write) ke baris data yang identik pada kedua server. Aspek ini menjadi peluang pengembangan penelitian selanjutnya, khususnya dalam mengukur metrik

performa replikasi dan pemulihan secara kuantitatif, menguji mekanisme resolusi konflik data pada skenario beban tinggi, serta mengevaluasi mekanisme deteksi dini terhadap query destruktif sebelum tereplikasi ke seluruh master.

D. Kesimpulan

Berdasarkan implementasi, pengujian fungsional, simulasi gangguan, dan skenario backup yang telah dilakukan, dapat disimpulkan bahwa replikasi database MySQL dengan skema master to master berhasil diimplementasikan pada dua server berbasis Ubuntu dengan konfigurasi yang tepat, mencakup pengaturan berkas my.cnf, pembuatan akun replikasi, serta pengaturan change master to pada kedua arah, sehingga kedua server mampu berfungsi secara bersamaan sebagai sumber dan penerima replikasi (bidirectional replication). Fungsi replikasi dua arah terbukti stabil dan konsisten, di mana operasi create database, insert, dan update pada salah satu server tereplikasi secara otomatis dan real-time ke server pasangannya tanpa menimbulkan konflik maupun kehilangan data. Pengujian ketahanan melalui simulasi gangguan perangkat keras menunjukkan bahwa sistem mampu mempertahankan kontinuitas layanan basis data pada server yang masih aktif selama server pasangannya nonaktif, dan mampu melakukan sinkronisasi ulang secara otomatis berbasis binary log setelah server tersebut kembali aktif. Di sisi lain, skenario backup dan pemulihan membuktikan bahwa replikasi master to master tanpa filtering konten turut mereplikasi query destruktif ke seluruh master, sehingga seluruh data dapat hilang secara bersamaan apabila terjadi insiden keamanan siber atau kesalahan operasional; risiko ini berhasil dimitigasi melalui strategi backup terjadwal yang terbukti mampu memulihkan ketersediaan data secara penuh dan mengembalikan replikasi ke kondisi normal. Temuan ini menunjukkan bahwa kombinasi replikasi master to master dan strategi backup merupakan pendekatan yang lebih menyeluruh untuk mendukung ketersediaan tinggi dan pemulihan bencana pada sistem basis data operasional, dibandingkan hanya mengandalkan replikasi semata. Untuk pengembangan lebih lanjut, disarankan penerapan sistem monitoring replikasi secara real-time, mekanisme delayed replica atau retensi binary log yang lebih panjang untuk mendukung pemulihan pada titik waktu tertentu, pembatasan hak akses terhadap perintah destruktif skala besar, serta pengukuran kuantitatif terhadap waktu pemulihan dan mekanisme resolusi konflik data pada skenario penulisan bersamaan, guna memperkuat keandalan implementasi pada lingkungan produksi berskala lebih besar.

Daftar Pustaka

Ajie, R., & Sutabri, T. (2025). Optimasi kinerja basis data terdistribusi menggunakan algoritma replication dan partitioning. Jurnal Sains Student Research, 3(2), 371–378. https://doi.org/10.61722/jssr.v3i2.4318

Akhlak Kamiswara, & Wijaya, A. (2023). Implementasi replikasi database MySQL master to master untuk menjamin ketersediaan data pada server. Jurnal Teknika, 21(1), 10–17.

Kristian, D. S., Rochim, A. F., & Widianto, E. D. (2015). Pengembangan sistem replikasi dan redundansi untuk meningkatkan kehandalan basis data MySQL. Jurnal Teknologi dan Sistem Komputer, 3(4), 523–529. https://doi.org/10.14710/jtsiskom.3.4.2015.523-529

Kurniadi, W., Djusmin, V. B., & Yasir, F. N. (2021). Replikasi database terdistribusi di universitas. Prosiding Semantik, 3(1), 258–262.

Maulana, H. (2016). Analisis dan perancangan sistem replikasi database MySQL dengan menggunakan VMware pada sistem operasi open source. InfoTekJar.

Oracle Corporation. (2024). MySQL 8.0 reference manual. https://dev.mysql.com/doc/refman/8.0/en/replication.html

PT Pupuk Sriwidjaja Palembang. (2025, 12 Maret). Profil perusahaan – Tentang kami. https://www.pusri.co.id/id/about/profile

Sholihah, I., & Darujati, C. (2022). Sistem replikasi basis data berdasarkan MySQL menggunakan container Docker. Majalah Ilmiah Teknologi Elektro, 21(2), 209–216. https://doi.org/10.24843/MITE.2022.v21i02.P08

Shrestha, R. (2017). High availability and performance of database in the cloud – Traditional master-slave replication versus modern cluster-based solutions. Proceedings of the 7th International Conference on Cloud Computing and Services Science.

Suhatman, R., Rida, M. A. F., Surya, I., & Tasya, R. (2024). Penerapan disaster recovery plan pada database MySQL menggunakan metode master to master replication berbasis Docker. Jurnal Komputer Terapan, 10(1), 78–85. https://doi.org/10.35143/jkt.v10i1.6226

Suprianto, A. (2019). Penyusunan DRP berbasis NIST SP 800-34 pada PT Pupuk Kaltim [Skripsi, Universitas Brawijaya].