

SIMULATION OF POPULATION DATABASE DISASTER RECOVERY USING POSTGRESQL AND VMWARE

SIMULASI *DISASTER RECOVERY* BASIS DATA KEPENDUDUKAN MENGGUNAKAN POSTGRESQL DAN VMWARE

Hanifah Syahidah¹, Fauzan Arif Tricahya², Brilyan Pancar Komara³, Muhammad Farhadh², and Shelvie Nidya Neyman^{5†}

^{1,2,3,4,5}Computer Science Study Program, IPB University, Indonesia

[†]corresponding author: shelvie@apps.ipb.ac.id

Copyright © 2026 Hanifah Syahidah, Fauzan Arif Tricahya, Brilyan Pancar Komara, Muhammad Farhadh, and Shelvie Nidya Neyman. This is an open-access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Population databases are critical assets for government institutions that underpin various public services, making system failures potentially disruptive to the continuity of public service delivery. Despite extensive studies on database backup and recovery strategies, their implementation and testing in the context of local government population databases in Indonesia using a chaos engineering approach remains insufficiently documented. This study simulates backup, recovery, and data integrity mechanisms on a synthetic Indonesian population dataset using a two-site virtual infrastructure based on VMware and Dockerized PostgreSQL 15. Three backup strategies were implemented (full, incremental, and differential backup) supported by AES-256-CBC encryption and real-time WAL-based streaming replication. Nine test scenarios were executed, encompassing chaos engineering with fault injection, DROP TABLE CASCADE, restoration testing, Point-in-Time Recovery, and encryption security testing. Results indicate that all scenarios were successfully executed with a Recovery Point Objective approaching zero and full recovery of 2,199,747 rows without data loss under the simulated conditions tested. Full backup achieved the fastest Recovery Time Objective of approximately 49 seconds, while differential backup required the longest duration of approximately 413 seconds due to cumulative dump size accumulation. These findings are expected to serve as a

technical reference for local government institutions in designing reliable and secure population database backup and recovery systems.

Keywords: Backup, Chaos Engineering, Data Integrity, PostgreSQL, Recovery.

Abstrak

Basis data kependudukan merupakan aset kritis bagi instansi pemerintah yang menopang berbagai layanan publik, sehingga kegagalan sistem dapat berdampak langsung pada kontinuitas pelayanan masyarakat. Meskipun strategi *backup* dan *recovery* basis data telah banyak dikaji secara umum, implementasi dan pengujiannya pada konteks basis data kependudukan pemerintah daerah di Indonesia dengan pendekatan *chaos engineering* masih belum banyak didokumentasikan. Penelitian ini mensimulasikan mekanisme *backup*, *recovery*, dan uji integritas pada dataset sintetis kependudukan Indonesia menggunakan infrastruktur *virtual* dua *site* berbasis VMware dan PostgreSQL 15 yang dikontainerisasi dengan Docker. Tiga strategi *backup* diimplementasikan yaitu *full*, *incremental*, dan *differential backup*, dilengkapi enkripsi AES-256-CBC dan *streaming replication* berbasis WAL secara *real-time*. Sebanyak sembilan skenario pengujian dieksekusi mencakup *chaos engineering* dengan metode *fault injection*, *DROP TABLE CASCADE*, *restoration test*, *Point-in-Time Recovery*, dan uji keamanan enkripsi. Hasil pengujian menunjukkan seluruh skenario berhasil dieksekusi dengan nilai *Recovery Point Objective* mendekati nol dan pemulihan data sebanyak 2.199.747 baris tanpa kehilangan data pada kondisi simulasi yang diujikan. Strategi *full backup* menghasilkan *Recovery Time Objective* tercepat sekitar 49 detik, sedangkan strategi *differential* memerlukan waktu terlama sekitar 413 detik akibat akumulasi ukuran *dump*. Temuan ini diharapkan dapat menjadi referensi teknis bagi instansi pemerintah daerah dalam merancang sistem *backup* dan *recovery* basis data kependudukan yang andal dan aman.

Kata Kunci: Backup, Chaos Engineering, Integritas Data, PostgreSQL, Recovery.

1. Pendahuluan

Basis data kependudukan warga negara Indonesia merupakan aset penting bagi instansi pemerintah Indonesia. Kependudukan Indonesia berdasarkan Badan Pusat Statistik Indonesia (2026) tercatat, jumlah seluruh warga penduduk Indonesia sekitar 287-288 juta jiwa. Basis data ini menjadi standar penting untuk pengembangan maupun pelayanan negara terhadap warganya. Berbagai struktur layanan publik bergantung pada basis data kependudukan ini, mulai dari kesehatan, administrasi, bantuan sosial, dan layanan publik lainnya. Masing-masing memiliki

peran tersendiri dalam kehidupan warga negara Indonesia, dan jika terjadi bencana yang menyebabkan kehilangan maupun kerusakan data pada sistem, hal tersebut akan berdampak langsung pada pelayanan masyarakat. Oleh karena itu diperlukannya *Data Recovery Center* (DRC) sebagai tindakan preventif menjaga kelangsungan pelayanan terhadap warga negara Indonesia. Hal ini juga disampaikan dalam Peraturan Presiden Nomor 95 (Indonesia, 2018) tentang Sistem Pemerintahan Berbasis Elektronik (SPBE) Pasal 40 ayat (4), yang menyatakan bahwa diperlukannya penjaminan ketersediaan sistem melalui penyediaan cadangan dan pemulihan data. Oleh karena itu pengujian secara berkala diperlukan untuk mengidentifikasi kekurangan dalam rencana pemulihan, mencakup peralatan, prosedur *backup*, hingga kinerja personel (Wallace & Webber, 2004).

Urgensi pengelolaan dan perlindungan basis data pemerintah semakin diperkuat oleh meningkatnya insiden keamanan siber di Indonesia. Rinaldo & Puspitasari (2024) mencatat lonjakan insiden siber berkembang hingga 1.65 miliar kasus pada 2021. Sejalan dengan itu, Hamid & Huda (2025) menunjukkan bahwa tren penelitian keamanan siber sektor pemerintah kini bergeser ke arah manajemen risiko siber, regulasi perlindungan data, dan tantangan adopsi *cloud computing*, yang mengindikasikan meningkatnya kesadaran global akan pentingnya kesiapan infrastruktur digital institusi publik. Kondisi ini menegaskan bahwa penguatan sistem informasi pemerintah tidak cukup hanya pada aspek regulasi, tetapi juga memerlukan implementasi teknis yang terukur dan teruji, termasuk mekanisme *backup* dan *recovery* basis data yang andal.

Strategi *backup* dan *recovery* basis data telah banyak dikaji dalam beberapa tahun terakhir. Alhamidi & Hamoudi (2024) membuktikan bahwa *incremental backup* otomatis berbasis *cron job* lebih efisien dalam penggunaan penyimpanan dibandingkan *full backup*, meskipun memiliki kompleksitas lebih tinggi saat pemulihan. Sutar *et al.* (2023) mengusulkan solusi terintegrasi yang menggabungkan *backup*, *recovery*, dan enkripsi dalam satu *platform* dengan fitur *point-in-time recovery* untuk basis data SQL dan MongoDB. Rajendiran (2022) mendemonstrasikan pemanfaatan mekanisme *Write-Ahead Log* (WAL) PostgreSQL untuk replikasi data real-time yang mampu menjaga konsistensi data pada lingkungan terdistribusi. Rahman & Soewito (2025) membuktikan bahwa kombinasi *Failover Cluster Instance* dan *Availability Groups* menghasilkan ketersediaan basis data sebesar 99,97% dengan *downtime* rata-rata hanya 12,3 menit. Dunak (2025) menyimpulkan bahwa pencapaian ketersediaan sistem yang tinggi membutuhkan kombinasi redundansi berlapis, otomasi, dan validasi konsistensi replika dengan RTO dan RPO sebagai indikator utamanya. Meskipun demikian, implementasi dan pengujian strategi tersebut pada konteks basis data kependudukan

pemerintah daerah di Indonesia dengan pendekatan *chaos engineering* masih belum banyak didokumentasikan.

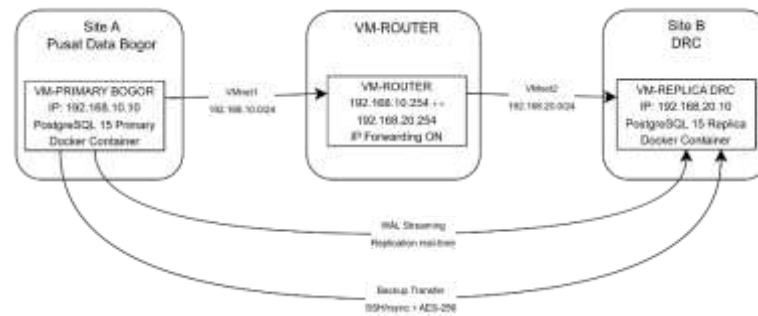
Penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan mengkombinasikan konteks basis data kependudukan Indonesia, pendekatan *chaos engineering*, dan kelola ~2,2 juta baris sintetis melalui simulasi berbasis infrastruktur *virtual* dua *site*. Berdasarkan permasalahan yang terjadi, penelitian ini bertujuan untuk membangun infrastruktur *virtual* dua *site* dengan PostgreSQL 15 yang dikontainerisasi menggunakan Docker, mengimplementasikan strategi *backup full*, *incremental*, dan *differential* dengan enkripsi AES-256-CBC, mengkonfigurasi streaming *replication* berbasis WAL, mengeksekusi skenario *chaos engineering*, serta mengukur metrik RTO dan RPO pasca-simulasi bencana.

2. Metodologi

2.1 Bahan dan Data

Penelitian ini menggunakan dataset sintetis yang merepresentasikan tiga sektor layanan publik yang dipilih berdasarkan tingkat ketergantungannya terhadap NIK dan nomor kartu keluarga sebagai identitas utama. Sektor kesehatan dipilih karena BPJS Kesehatan merupakan salah satu pengguna data NIK terbesar, dengan cakupan peserta JKN mencapai 283.096.765 jiwa atau 98,75 persen dari seluruh penduduk Indonesia hingga 30 November 2025 (ANTARA News, 2025). Sektor layanan publik dipilih karena mencakup layanan administrasi dan data pemerintah yang verifikasi identitasnya bergantung pada NIK. Sektor bantuan sosial dipilih karena program bansos pemerintah menjangkau 16,5 juta keluarga penerima manfaat pada triwulan II 2025 yang seluruh proses identifikasi penerimanya berbasis NIK dan nomor kartu keluarga (Kementerian Sekretariat Negara Republik Indonesia, 2025). Ketiga sektor ini dipilih untuk merepresentasikan spektrum dampak yang komprehensif apabila terjadi kegagalan sistem basis data kependudukan.

Dataset yang digunakan pada penelitian ini merupakan data sintesis kependudukan Indonesia dari berbagai sektor, yaitu sektor kesehatan, layanan publik, dan rumah tangga. Datasetnya mencakup enam entitas utama yang saling berkorelasi, yaitu instansi, kartu_keluarga, penduduk, bpjs_kesehatan, layanan_publik, dan bansos_umkm, dengan rincian jumlah baris data yang disajikan pada Tabel 3. Setiap entitas penduduk memiliki atribut NIK, nama lengkap, tanggal lahir, jenis kelamin, dan alamat yang dipetakan berdasarkan nomor kartu keluarga. Infrastruktur simulasi dua *site* yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.

Gambar 1: Arsitektur infrastruktur simulasi dua *site*.

Infrastruktur yang digunakan terdiri atas tiga *virtual machine* (VM). VM pertama berperan sebagai *data center* utama yaitu VM-PRIMARY-BOGOR (IP 192.168.10.10), VM kedua berperan sebagai *server standby* yaitu VM-REPLICA-DRC (IP 192.168.20.10), dan VM ketiga berperan sebagai *virtual router* yaitu VM-ROUTER (IP 192.168.10.254 / 192.168.20.254). VM-PRIMARY dan VM-REPLICA berada pada subnet yang berbeda dan dihubungkan melalui VM-ROUTER menggunakan dua jaringan *host-only* VMware, VMnet1 (192.168.10.0/24) dan VMnet2 (192.168.20.0/24). Basis data pada simulasi ini dijalankan dengan menggunakan Docker *container* dengan *image* PostgreSQL 15, dan enkripsi *backup*-nya menggunakan OpenSSL AES-256-CBC. Seluruh proses simulasi mulai dari *backup*, *recovery*, dan uji integritas basis data ini dilakukan pada perangkat laptop lokal peneliti.

2.2 Metode Penelitian

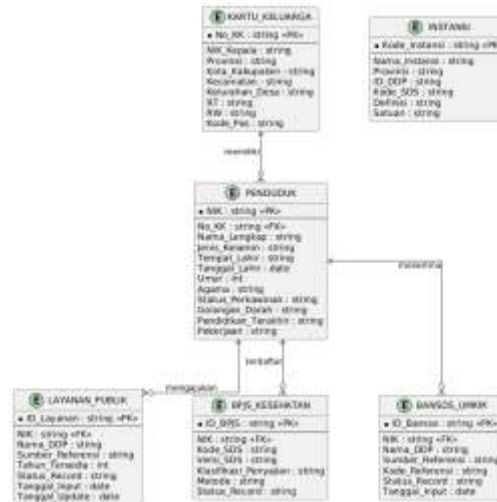
2.2.1. Pembangkitan Data Sintesis

Data sintetis dibangkitkan menggunakan fungsi `generate_series` PostgreSQL dan `pandas` untuk menghasilkan atribut kependudukan seperti NIK, nama lengkap, tanggal lahir, jenis kelamin, dan alamat sesuai format data Indonesia. Struktur data mengacu pada portal Open Data Jawa Barat (opendata.jabarprov.go.id) dengan dukungan Claude Sonnet 4.6 dalam perancangan skema data. Pendekatan sintetis dipilih karena akses terhadap data kependudukan asli memerlukan otorisasi khusus dari instansi pemerintah.

2.2.2. Setup Infrastruktur & Lingkungan

Tahap awal dari simulasi ini adalah membangun 2 lingkungan simulasi dua *site data center* yaitu *site A* : Pusat data bogor (*Primary Database*) dan *site B* : DRC (*Replica Database*). Untuk pengerjaannya terdiri dari enam langkah, yaitu: pembuatan *virtual machine* menggunakan VMware Workstation dengan Ubuntu Server 24.04 LTS, konfigurasi jaringan *host-only* dua subnet, instalasi docker dan docker compose, setup postgresQL dengan docker compose, perancangan basis data (ERD dan DDL), dan terakhir

adalah skrip inisialisasi basis data otomatis ([init-db.sh](https://github.com/indonesiabank/init-db.sh)). Struktur ERD untuk dataset sintetis kependudukan Indonesia tersebut ditunjukkan pada Gambar 2.



Gambar 2: ERD dataset sintesis kependudukan Indonesia.

2.2.3. Implementasi *Backup & Replikasi*

Tahapan selanjutnya merupakan sistem replikasi data *real-time* dan *backup* otomatis yang dibangun antara node primer (VM-PRIMARY-BOGOR) dan node standby (VM-REPLICA-DRC). Replikasi antar node dikonfigurasi menggunakan mekanisme *streaming replication* bawaan PostgreSQL berbasis *Write-Ahead Log* (WAL) (The PostgreSQL Global Development Group, 2026b). Pendekatan ini dipilih karena tidak memerlukan perangkat lunak tambahan dan secara efektif mampu meneruskan setiap perubahan data ke node standby secara *real-time* (Rajendiran, 2022).

Sebagai pelengkap replikasi, strategi pencadangan data (*backup*) juga diimplementasikan dalam tiga pendekatan menggunakan utilitas bawaan (*pg_dump*), yakni *full backup*, *differential backup*, dan *incremental backup* (de Guise, 2020). Guna memastikan integritas dan kerahasiaan data kependudukan yang sensitif, diterapkan keamanan berlapis. Lapis pertama adalah enkripsi *at-rest* menggunakan standar AES-256-CBC yang direkomendasikan NIST (2023), di mana berkas *plaintext* langsung dihapus setelah dienkripsi. Lapis kedua adalah enkripsi *in-transit* menggunakan terowongan SSH/TLS (*rsync* via SSH) untuk mengamankan proses transfer berkas *backup* antar *site* agar tidak dapat disadap. Seluruh rangkaian proses *backup* tersebut dieksekusi secara otomatis tanpa intervensi administrator menggunakan penjadwalan *cron job* di server primer. Adapun rincian strategi, frekuensi, dan beban dari masing-masing metode *backup*

diuraikan pada Tabel 1.

Tabel 1: Strategi *backup* otomatis.

Tipe	Frekuensi	Keterangan Prosedur	Ukuran Relatif
<i>Full</i>	Setiap Minggu 01:00	<i>Dump</i> lengkap seluruh basis data	Besar
<i>Incremental</i>	Setiap Senin-Sabtu 02:00	<i>Dump</i> lengkap dengan <i>checkpoint</i> harian	Kecil
<i>Differential</i>	Setiap Senin-Sabtu 03:00	<i>Dump</i> lengkap dengan titik pemulihan antara dua <i>full backup</i>	Sedang

Konfigurasi dilakukan melalui skrip [setup-cron.sh](#) yang menambahkan entri ke *crontab* secara otomatis sehingga sistem *backup* berjalan tanpa intervensi administrator.

2.2.4. Recovery Drill (Simulasi Bencana Data), Validasi Integritas, dan Dokumentasi SOP

Tahapan ini merupakan proses akhir dari proses *backup*, *recovery*, dan uji integritas basis data kependudukan. Fase ini bertujuan menguji ketahanan sistem melalui *chaos engineering* (skenario negatif) sekaligus memvalidasi integritas data pasca pemulihan. Seluruh proses pengujian diotomatisasi menggunakan [recovery-drill.sh](#) yang menjalankan sembilan skenario simulasi bencana secara terstruktur pada dataset kependudukan dengan total lebih dari 2,1 juta baris data.

Pengujian ketahanan sistem dilakukan menggunakan pendekatan *chaos engineering* dengan metode *fault injection*, yaitu penyuntikan kegagalan secara terencana untuk mengevaluasi *fault tolerance* sistem terhadap kondisi abnormal sekaligus memverifikasi efektivitas mekanisme deteksi dan penanganan kegagalan yang telah diimplementasikan (Nandigam, 2024). Terdapat tiga skenario *negative testing* yang dieksekusi, sebagaimana diuraikan pada Tabel 2.

Tabel 2: Skenario *chaos engineering*.

No	Skenario	Mekanisme	Tujuan
1	File <i>backup</i> dikorupsi	injeksi byte statis di tengah file .enc	Menguji deteksi kerusakan file sebelum <i>restore</i> .
2	Interupsi jaringan saat	rsync dihentikan secara paksa pada	Menguji validitas file <i>backup</i> saat interupsi.

	<i>backup</i>	progres 50%	
3	Disk penuh di site primary	Reduksi kapastias disk hingga <30 Mb	Memastikan sistem memberikan peringatan yang tepat.

Selain skenario kegagalan operasional, simulasi bencana utama juga dieksekusi berupa kehilangan data masif dengan menggunakan DROP TABLE CASCADE yang merepresentasikan insiden *human error* atau SQL Injection. Tabel layanan_public, bpjs_kesehatan, bansos_umkm, dan penduduk dihapus secara paksa dari sistem Primarynya. Skenario ini ditujukan untuk mengamati perambatan efek replikasi asinkron (*replication lag*) dari node Primary ke node Replica. Proses ini dilanjutkan dengan pengujian pemulihan data untuk mengukur metrik *Recovery Time Objective* (RTO) dari tiga strategi *backup* yang telah diimplementasikan, yaitu *full backup*, *differential*, juga *incremental*. Proses pemulihan ini diuji dengan mengembalikan seluruh data ke masing-masing basis data target (db_full, db_diff, db_inc). Selain itu, dilakukan juga pengujian *Point-in-Time Recovery* (PITR) untuk mengembalikan status basis data ke kondisi tepat sebelum gelombang (*batch*) injeksi data masuk (*Checkpoint T0*) (The PostgreSQL Global Development Group, 2026a).

Sebagai validasi keamanan akhir, sistem diuji dengan mencoba memulihkan data *at-rest* terenkripsi menggunakan kunci AES-256 (*passphrase*) yang salah, untuk membuktikan bahwa data tidak dapat diakses tanpa kunci dekripsi yang sah. Selain pengujian tersebut, simulasi ini juga terdapat beberapa limitasi teknis yang perlu dicatat. Karena sistem ini disimulasikan menggunakan limitasi bawaan (*native*) dari pg_dump dan pg_basebackup, maka *differential backup* yang tidak didukung langsung oleh *tools native* PostgreSQL, sehingga hanya direpresentasikan melalui modifikasi *dump*. Selanjutnya, restorasi parsial untuk *incremental backup* perlu disimulasikan pada level artefak fisik yaitu (.tar.gz.enc). Terakhir untuk aplikasi dunia asli, peneliti merekomendasikan agar menggunakan peranti *3rd-party* khusus seperti pgBackRest untuk mendukung strategi *backup* yang lebih komprehensif (The PostgreSQL Global Development Group, 2026c).

3. Hasil dan Pembahasan

Berdasarkan percobaan, hasil simulasi *backup*, *recovery*, dan uji integritas basis data simulasi ini berhasil dilaksanakan dan memberikan hasil sesuai ekspektasi awal. Berikut merupakan hasil dan pembahasan dari setiap tahapan yang telah dilakukan.

3.1 Hasil Setup Infrastruktur dan Lingkungan

Penelitian Pada tahapan ini dataset yang diinisialisasi sesuai dengan dataset asli sebagaimana hasilnya dapat dilihat pada Tabel 3.

Tabel 3: Hasil verifikasi jumlah data per tabel.

Tabel	Jumlah Baris
instansi	21
kartu_keluarga	2.201.551
penduduk	2.199.747
bpjs_kesehatan	2.701.000
layanan_publik	2.701.000
bansos_umkm	2.701.000

3.2 Hasil Implementasi Replikasi dan *Backup*

Konfigurasi *streaming replication* WAL antara VM-PRIMARY-BOGOR dan VM-REPLICA-DRC berhasil diverifikasi dengan status streaming pada kolom status di `pg_stat_wal_receiver`. Sinkronisasi antara data real-time primary dan replika berjalan dengan baik, dibuktikan dengan data yang dimasukkan di Primary langsung muncul di Replica dalam beberapa detik. Sebanyak empat basis data (`db_kependudukan`, `db_full`, `db_inc`, `db_diff`) berhasil tersinkronisasi. Kemudian untuk hasil *backup* dari empat pengujian memberikan hasil tiga berhasil dan satu ditolak, untuk detailnya tertera pada Tabel 4.

Tabel 4: Hasil verifikasi *backup*.

No	Pengujian	Hasil
1	<i>Full backup</i> empat basis data	Berhasil
2	Enkripsi AES-256	Berhasil
3	Transfer via SSH ke replica	Berhasil
4	Test <i>passphrase</i> salah	Ditolak

3.3 Hasil *Chaos Engineering* (*Negative Testing*) dan Hasil *Restoration Test*

Seluruh tiga skenario *chaos engineering* dinyatakan [PASS], membuktikan sistem memiliki mekanisme *fault tolerance* yang memadai. Korupsi file terdeteksi via *checksum* MD5, interupsi jaringan tidak menghasilkan file *backup* yang tidak valid, dan *disk* penuh berhasil ditangani tanpa menyebabkan korupsi data. Keberhasilan ini sejalan dengan tujuan utama *chaos engineering*, yaitu memverifikasi efektivitas penanganan kegagalan dan resiliensi sistem sebelum insiden sungguhan terjadi di lingkungan produksi nyata (Nandigam, 2024). Untuk hasil pengukuran RTO detailnya tertera pada Tabel 5.

Tabel 5: Hasil pengukuran RTO pada setiap skenario *Recovery*.

Skenario	Target <i>restore</i>	Data terpulihkan	RTO	Integritas (<i>data loss</i>)
<i>Full Backup</i>	db_full	2.199.747 baris	~49 detik	0 baris (100% valid)
<i>Differential</i>	db_diff	2.199.747 baris	~413 detik	0 baris (100% valid)
<i>Incremental</i>	db_inc	2.199.747 baris	~257 detik	0 baris (100% valid)
PITR	-	2.199.747 baris	~219 detik	0 baris (100% valid)

Berdasarkan hasil pengukuran RTO, strategi *full backup* menghasilkan waktu pemulihan tercepat yaitu ~49 detik karena hanya memuat satu file tunggal (de Guise, 2020). *Incremental* memerlukan waktu ~257 detik akibat *dependency chain* pemilihan berurutan (de Guise, 2020). Meskipun secara konseptual strategi *differential* seharusnya lebih cepat dari *incremental* karena hanya memerlukan dua komponen, namun dalam simulasi ini *differential* justru menghasilkan RTO tertinggi yaitu ~413 detik karena akumulasi *dump*-nya paling besar yang meningkatkan *parsing overhead* saat *restore* menjadi lebih tinggi dibandingkan strategi lainnya (de Guise, 2020). Sementara itu, PITR memerlukan waktu ~219 detik untuk merestorasi data ke kondisi *Checkpoint TO* melalui mekanisme *replay WAL* bertahap (The PostgreSQL Global Development Group, 2026a).

3.4 Hasil Skenario Bencana Utama

Pada skenario DROP TABLE CASCADE (*data loss* secara masif), tabel layanan_public, bpjs_kesehatan, bansos_umkm, dan penduduk berhasil dihapus secara paksa dari *node Primary*. Efek replikasi ini kemudian merambat ke node Replica secara *real-time* melalui *streaming replication WAL* (The PostgreSQL Global Development Group, 2026b). Skenario ini memvalidasi pentingnya keberadaan sistem *backup* independen yang tidak bergantung pada replikasi semata, karena data yang terhapus di *Primary* akan ikut terhapus di *Replica*.

3.5 Hasil Uji Keamanan Enkripsi

Pengujian keamanan enkripsi *data at-rest* dengan menggunakan kunci AES-256 (*passphrase*) yang salah menghasilkan penolakan akses secara total oleh sistem, sesuai dengan standar enkripsi AES-256-CBC yang direkomendasikan untuk perlindungan data sensitif (NIST, 2023). Penerapan enkripsi berlapis, yaitu *at-rest* menggunakan AES-256-CBC dan *in-transit* menggunakan SSH/TLS, memastikan bahwa file *backup* tidak dapat dibaca meskipun berhasil disadap selama proses pengiriman antar *site*.

4. Simpulan dan Saran

Simulasi *backup*, *recovery*, dan uji integritas basis data kependudukan berhasil dilakukan pada infrastruktur lokal berbasis VMware dan Docker PostgreSQL. Seluruh pengujian sembilan skenario dinyatakan berhasil, dengan ketiga strategi *backup* berhasil memulihkan 2.199.747 baris data tanpa kehilangan data. *Full backup* menghasilkan RTO tercepat (~49 detik), *differential* menjadi terlama (~413 detik), dan RPO mendekati nol pada seluruh skenario. Selain itu *streaming replication* WAL berhasil berjalan stabil dengan sinkronisasi *real-time* yang terverifikasi, dan enkripsi berlapis *at-rest* (AES-256-CBC) serta *in-transit* (SSH/TLS) berfungsi sebagaimana seharusnya, dibuktikan dengan penolakan akses total terhadap kunci dekripsi yang salah.

Keterbatasan penelitian ini mencakup tidak adanya beban pengguna konkuren, latensi WAN nyata, dan *failover* otomatis, sehingga hasil RTO dan RPO belum sepenuhnya merepresentasikan kondisi produksi. Strategi *incremental* dan *differential* berbasis *pg_dump* juga tidak merepresentasikan *delta backup* sesungguhnya, karena PostgreSQL tidak mendukung mekanisme *delta* secara *native* (The PostgreSQL Global Development Group, 2026c). Pada penelitian selanjutnya direkomendasikan untuk menggunakan *pgBackRest* guna mendukung implementasi yang lebih komprehensif.

Ucapan Terima Kasih. Penulis mengucapkan terima kasih kepada Allah SWT, Ibu Shelve selaku dosen pembimbing, seluruh rekan kelompok: Hanifah Syahidah, Fauzan Arif Tricahya, Brilyan Pancar Komara, dan Muhammad Farhadh, dan orang tua masing-masing atas doa dan dukungan selama penelitian ini berlangsung.

Daftar Pustaka

- Alhamidi, R. S., & Hamoudi, R. (2024). ChronoBak: Automated cost-effective incremental backup software for large-scale data. *Advances in Biomedical and Health Sciences*, 3(4), 170–176. https://doi.org/10.4103/abhs.abhs_22_24.
- ANTARA News. (2025). *BPJS Kesehatan: 2026 momen kuatkan integrasi JKN lewat 4 prioritas*. <https://m.antaranews.com/amp/berita/5328631/bpjs-kesehatan-2026-momen-kuatkan-integrasi-jkn-lewat-4-prioritas>.
- Badan Pusat Statistik Indonesia. (2026). *Statistik Indonesia 2026*, Katalog 1101001, Jakarta: Badan Pusat Statistik Indonesia.
- de Guise, P. (2020). *Data Protection: Ensuring Data Availability* (2nd ed.). Auerbach Publications. <https://doi.org/10.1201/9780367463496>.
- Dunak, J. (2025). Database backup and recovery for near-perfect availability. *American Scientific Research Journal for Engineering, Technology, and Sciences*, 103(1), 544-554.

- https://asrjetsjournal.org/American_Scientific_Journal/article/view/12133.
- Hamid, S., & Huda, M. N. (2025). Mapping the landscape of government data breaches: A bibliometric analysis of literature from 2006 to 2023. *Social Sciences and Humanities Open*, 11, 101234. <https://doi.org/10.1016/j.ssaho.2024.101234>.
- Indonesia. (2018). *Peraturan Presiden Nomor 95 Tahun 2018 tentang Sistem Pemerintahan Berbasis Elektronik*, Jakarta: Sekretariat Negara.
- Kementerian Sekretariat Negara Republik Indonesia. (2025). *Pemerintah salurkan bansos PKH dan sembako bagi 16,5 juta keluarga*. https://setneg.go.id/baca/index/pemerintah_salurkan_bansos_pkh_dan_sembako_bagi_165_juta_keluarga.
- Nandigam, V. D. G. (2024). Chaos engineering: Stress-testing microservices for resilience. *International Journal of Research in Computer Applications and Information Technology*, 7(2), 1993–2004. <https://doi.org/10.5281/zenodo.14282950>.
- National Institute of Standards and Technology. (2023). *Advanced Encryption Standard (AES)*. <https://doi.org/10.6028/NIST.FIPS.197-upd1>.
- Rahman, F. & Soewito, B. (2025). Enhancing database availability: A combined approach using SQL Always On failover cluster instance and availability groups. *Journal of Computer Science*, 21(6), 1332-1342. <https://doi.org/10.3844/jcssp.2025.1332.1342>.
- Rajendiran, G. R. (2022). Real-time data replication for Postgres using WAL event capture and its usage in e-commerce inventory management. *Journal of Economics & Management Research*, 3(2), 1–5. [https://doi.org/10.47363/JESMR/2022\(3\)E108](https://doi.org/10.47363/JESMR/2022(3)E108).
- Rinaldo & Puspitasari, D. (2024). Digital risks in emerging economies: Cyber threat escalation in Indonesia (2020–2023). *Data: Journal of Information Systems and Management*, 2(3), 152–163. <https://doi.org/10.61978/data.v2i3.697>.
- Sutar, S., Jose, K., Gaikwad, V., Mishra, V., Wankhede, D., & Karnik, M. (2023). Enhancing data management: An integrated solution for database backup, recovery, conversion, and encryption capabilities. *International Journal of Intelligent Systems and Applications in Engineering*, 12(6s), 720–734. <https://ijisae.org/index.php/IJISAE/article/view/4011>.
- The PostgreSQL Global Development Group. (2026a). *Backup and restore: Continuous Archiving and PITR*. <https://www.postgresql.org/docs/15/continuous-archiving.html>.
- The PostgreSQL Global Development Group. (2026b). *Log-Shipping*

standby servers: Streaming replication.
<https://www.postgresql.org/docs/current/warm-standby.html>.

The PostgreSQL Global Development Group. (2026c). *pgBackRest: Reliable PostgreSQL backup & restore*. <https://pgbackrest.org>.

Wallace, M. & Webber, L. (2004). *The disaster recovery handbook: A step-by-step plan to ensure business continuity and protect vital operations, facilities, and assets*, New York: AMACOM.