STRATEGI MIGRASI WORKLOAD KRITIS KE CLOUD DENGAN MEKANISME FALLBACK OTONOM BERBASIS KONTEKS

M. Daris Dzimar Darussalam, Muhammad Faizal Abymanyu, Suharno, Ramdan Nugraha, Ibnu Muakhori
Institut Teknologi dan Bisnis Visi Nusantara Bogor, Bogor, Indonesia
email: daris.dzimar@itbvinusbogor.ac.id , faizal.abimanyu@itbvinusbogor.ac.id, suharno@itbvinusbogor.ac.id,ramdan.nugraha@itbvinusbogor.ac.id, ibnu@itbvinusbogor.ac.id

Migrating critical workloads to cloud environments introduces significant operational risks due to network volatility, provider outages, and strict latency/availability requirements. Traditional migration strategies rely on static thresholds or manual intervention, resulting in unacceptable downtime during emergency or mission-critical scenarios. This paper proposes an autonomous context-aware fallback mechanism that dynamically orchestrates workload migration based on real-time contextual scoring of network conditions, compute availability, security posture, and service-level agreement (SLA) compliance. The framework integrates a lightweight decision engine with stateful migration orchestration, enabling proactive fallback to edge or hybrid cloud nodes before degradation crosses critical thresholds. Experimental evaluation across simulated cloud-edge environments demonstrates a 73% reduction in migration-induced downtime, 99.2% packet delivery ratio under network degradation, and fallback activation within 1.8 seconds. The proposed strategy outperforms reactive and threshold-based baselines while maintaining computational overhead below 6.4% of node resources. These results validate the feasibility of context-driven autonomous migration for critical infrastructure, offering a resilient, low-latency alternative for disaster response, healthcare IoT, and industrial control systems.
Keywords: Autonomous Fallback; Cloud Migration; Context-Aware Computing; Critical Workloads; Edge-Cloud Continuum

Abstrak
Migrasi workload kritis ke lingkungan cloud menghadirkan risiko operasional signifikan akibat volatilitas jaringan, outage provider, dan persyaratan latensi/ketersediaan yang ketat. Strategi migrasi tradisional mengandalkan ambang batas statis atau intervensi manual, mengakibatkan downtime yang tidak dapat diterima pada skenario darurat atau misi kritis. Penelitian ini mengusulkan mekanisme fallback otonom berbasis konteks yang mengorkestrasi migrasi workload secara dinamis berdasarkan penilaian kontekstual real-time terhadap kondisi jaringan, ketersediaan komputasi, postur keamanan, dan kepatuhan service-level agreement (SLA). Kerangka kerja ini mengintegrasikan decision engine ringan dengan orkestrasi migrasi stateful, memungkinkan fallback proaktif ke edge atau node hybrid cloud sebelum degradasi melampaui ambang kritis. Evaluasi eksperimental pada lingkungan cloud-edge tersimulasi menunjukkan reduksi 73% pada downtime akibat migrasi, packet delivery ratio 99,2% saat degradasi jaringan, dan aktivasi fallback dalam 1,8 detik. Strategi usulan mengungguli pendekatan reaktif dan berbasis

ambang batas sambil mempertahankan overhead komputasi di bawah 6,4% dari sumber daya node. Hasil ini memvalidasi kelayakan migrasi otonom berbasis konteks untuk infrastruktur kritis, menawarkan alternatif tangguh dan berlatensi rendah untuk respons bencana, IoT kesehatan, dan sistem kontrol industri.

Kata kunci: Fallback Otonom; Komputasi Berbasis Konteks; Kontinuitas Edge-Cloud; Migrasi Cloud; Workload Kritis

1. PENDAHULUAN

Migrasi workload kritis ke infrastruktur cloud telah menjadi pilar utama transformasi digital modern, memungkinkan komputasi terukur, manajemen data terpusat, dan alokasi sumber daya yang efisien secara biaya. Namun, aplikasi kritis seperti jaringan komunikasi darurat, pemantauan kesehatan real-time, dan sistem kontrol industri menuntut latensi deterministik, ketersediaan berkelanjutan, dan jaminan integritas data yang ketat. Lingkungan cloud, meskipun skalabel, tetap rentan terhadap partisi jaringan, kegagalan sisi provider, pelanggaran SLA, dan insiden keamanan yang dapat mengganggu operasi misi kritis. Strategi migrasi cloud konvensional umumnya menggunakan penjadwalan statis, alur kerja persetujuan manual, atau mekanisme failover reaktif yang hanya terpicu setelah degradasi layanan terjadi. Pendekatan ini tidak memadai untuk workload sensitif waktu di mana bahkan beberapa detik downtime dapat mengakibatkan kegagalan operasional atau hilangnya nyawa.

Kemajuan terkini dalam komputasi edge dan arsitektur kontinuitas cloud-edge telah memperkenalkan kapabilitas fallback terdistribusi, namun sebagian besar implementasi kekurangan pengambilan keputusan cerdas berbasis konteks. Orkestrator migrasi umumnya bergantung pada utilisasi CPU/memori atau ambang batas latensi sederhana, mengabaikan sinyal kontekstual multi-dimensi seperti jitter jaringan, kedekatan geografis, intelijen ancaman, dan kepatuhan SLA historis. Selain itu, migrasi stateful dari workload kritis sering kali menimbulkan overhead tinggi akibat sinkronisasi data sinkron dan protokol checkpointing yang kaku, yang memperburuk latensi selama transisi fallback. Tidak adanya mekanisme fallback otonom berbasis konteks merupakan celah riset kritis, khususnya untuk domain darurat dan keselamatan-kritis di mana prediktabilitas dan ketahanan harus dijamin dalam kondisi volatil.

Penelitian ini mengatasi celah tersebut dengan memperkenalkan strategi fallback otonom berbasis konteks untuk migrasi workload kritis. Kerangka kerja yang diusulkan secara berkelanjutan memantau metrik kontekstual multi-sumber, menghitung skor ketahanan dinamis, dan memicu migrasi preventif ke node fallback optimal sebelum degradasi layanan terjadi. Decision engine menggabungkan machine learning ringan dengan prioritisasi berbasis aturan untuk menyeimbangkan kinerja, keamanan, dan kendala sumber daya. Orkestrator migrasi stateful memastikan kehilangan data minimal melalui checkpointing asinkron dan sinkronisasi diferensial.

Kontribusi penelitian ini tiga aspek: (1) model penilaian berbasis konteks yang dikhususkan untuk migrasi workload kritis, (2) pipa orkestrasi fallback otonom dengan latensi aktivasi di bawah 2 detik, dan (3) validasi empiris komprehensif yang menunjukkan ketahanan superior, downtime lebih rendah, dan overhead sumber daya yang dapat diterima dibandingkan strategi migrasi tradisional.

2. METODOLOGI

Penelitian ini menggunakan pendekatan eksperimental berbasis prototipe fisik dan simulasi serangan terkontrol, mengacu pada kerangka evaluasi keamanan IoT dan ZTA yang direkomendasikan dalam literatur terkini (NIST, 2020; Al-Garadi et al., 2021). Tahapan metodologis dirancang secara sistematis untuk memastikan validitas teknis dan relevansi operasional.

2.1 Desain Arsitektur Sistem

Kerangka kerja yang diusulkan terdiri dari tiga lapisan saling terhubung: Pemantauan Konteks, Decision Engine, dan Orkestrator Migrasi. Setiap lapisan dirancang beroperasi dengan latensi minimal dan perilaku deterministik, memastikan kesesuaian untuk workload kritis.

Lapisan Pemantauan Konteks mengumpulkan metrik kontekstual multi-dimensi secara berkelanjutan dari node sumber, node target, dan infrastruktur jaringan. Metrik dikategorikan ke dalam empat domain:

a. Konteks Jaringan: Latensi (RTT), jitter, packet loss, ketersediaan bandwidth, dan stabilitas resolusi DNS.
b. Konteks Komputasi: Utilisasi CPU, tekanan memori, throughput I/O, dan status kesehatan kontainer.
c. Konteks Keamanan: Alert deteksi intrusi, integritas hash firmware, validitas sertifikat TLS, dan skor anomali dari model perilaku ringan.
d. Konteks SLA & Operasional: Uptime historis, tingkat kepatuhan provider, kedekatan geografis, dan status kesiapan node fallback.

Data diagregasi melalui pipa telemetri ringan menggunakan packet capture berbasis eBPF dan exporter kompatibel Prometheus. Interval pengambilan sampel bersifat adaptif: 500 ms dalam kondisi stabil, meningkat menjadi 100 ms saat volatilitas kontekstual untuk memastikan deteksi tepat waktu tanpa membebani sumber daya control plane.

Lapisan Decision Engine menghitung Skor Ketahanan dinamis (Rs) untuk setiap workload aktif menggunakan model evaluasi multi-kriteria tertimbang:

$$R_s = \sum_{i=1}^{n} w_i \cdot \frac{C_i - C_{i,\min}}{C_{i,\max} - C_{i,\min}}$$

Dengan Ci merepresentasikan metrik kontekstual ternormalisasi, Wi menunjukkan bobot spesifik domain (dituning untuk workload kritis: Jaringan 0,35, Komputasi 0,25, Keamanan 0,25, SLA 0,15), dan $Rs \in [0,1]$. Pemicu fallback diaktifkan ketika Rs turun di bawah ambang dinamis θ_t, dihitung sebagai:

$$\theta_t = \theta_{base} - \alpha \cdot \Delta R_s^{(t-1)}$$

θ

dengan θbase =0,65 (dioptimalkan secara empiris), α = 0,15 (faktor peluruhan adaptif), dan Δ Rs merepresentasikan laju degradasi ketahanan pada jendela evaluasi sebelumnya. Ambang adaptif

ini mencegah migrasi prematur selama fluktuasi sementara sambil memastikan intervensi proaktif selama degradasi berkelanjutan.

Lapisan Orkestrator Migrasi mengeksekusi pipa migrasi tiga fase saat aktivasi fallback:

a. *Checkpointing State*: Memanfaatkan CRIU (Checkpoint/Restore In Userspace) untuk menangkap state aplikasi secara asinkron, memprioritaskan halaman memori kritis dan file descriptor terbuka.
b. Sinkronisasi Diferensial: Mentransfer hanya region memori yang dimodifikasi dan log delta-state ke node target, mengurangi konsumsi bandwidth hingga 68% dibandingkan migrasi full-state.
c. Redireksi Trafik: Memperbarui tabel routing DNS dan service mesh secara atomik menggunakan konsensus etcd, memastikan pengiriman paket tanpa duplikasi selama transisi.

Seluruh pipa beroperasi di bawah SLA ketat 2 detik, ditegakkan melalui pemantauan deadline real-time dan rollback fallback jika kesehatan node target menurun selama migrasi.

2.2 Konfigurasi Lingkungan Pengujian

Lingkungan eksperimental terdiri dari pengaturan hybrid cloud-edge:

a. Cloud Primer: AWS EC2 (t3.large, us-east-1) menjalankan Kubernetes v1.28
b. Node Fallback Edge: 3× Raspberry Pi 4 (4GB RAM) dan 2× Intel NUC menjalankan K3s
c. Workload: Simulasi pipa data IoT darurat (MQTT broker, kontainer analitik real-time, instance PostgreSQL stateful)
d. Stack Pemantauan: Prometheus, Grafana, exporter eBPF, agen telemetri kustom
e. Injeksi Chaos: Pumba (delay/loss jaringan), Chaos Mesh (kegagalan pod/jaringan)

2.3 Skenario Evaluasi

Empat skenario operasional disimulasikan:

a. Baseline: Jaringan stabil, tanpa degradasi
b. Degradasi Jaringan: 15% packet loss, peningkatan latensi 120 ms, reduksi bandwidth 30%
c. Cloud Outage: Ketidaktersediaan provider primer (disimulasikan via aturan firewall)
d. Anomali Keamanan: Kedaluwarsa sertifikat TLS + injeksi pola trafik anomali

Setiap skenario dieksekusi selama 50 siklus migrasi, dengan metrik dicatat per transisi.

2.4 Perbandingan Baseline dan Metrik

a. Strategi usulan dibandingkan terhadap:
b. Migrasi Ambang Batas Statis: Terpicu pada ambang CPU/latensi tetap
c. Fallback Reaktif: Teraktivasi hanya setelah pelanggaran SLA
d. Orkestrasi Manual: Migrasi diinisiasi operator via CLI

Metrik evaluasi utama mencakup: Downtime Migrasi (ms), Packet Delivery Ratio (PDR, %), Latensi Aktivasi Fallback (s), Overhead Sumber Daya (CPU/Memori %), dan Tingkat Error Konsistensi Data (%).

3. HASIL DAN PEMBAHASAN

3.1 Downtime Migrasi dan Latensi Aktivasi

Kerangka kerja usulan mencapai downtime migrasi rata-rata 1.420 ms, reduksi 73% dibandingkan migrasi ambang batas statis (5.280 ms) dan peningkatan 81% dibandingkan fallback reaktif (7.510 ms). Latensi aktivasi fallback rata-rata 1,8 detik, secara konsisten memenuhi persyaratan SLA di bawah 2 detik. Orkestrasi manual menunjukkan variabilitas tertinggi (3,2–8,9 s), bergantung pada waktu respons operator.

Tabel 1. Perbandingan Kinerja Migrasi Antar Skenario

Strategi	Rata-rata Downtime (ms)	Latensi Aktivasi (s)	PDR (%)	Overhead (%)
Ambang Batas Statis	5.280 ± 310	4,1 ± 0,6	91,3	9,2
Fallback Reaktif	7.510 ± 480	6,8 ± 0,9	87,5	11,7
Orkestrasi Manual	6.320 ± 520	5,4 ± 1,1	89,1	8,5
Usulan (Berbasis Konteks)	1.420 ± 95	1,8 ± 0,2	99,2	6,4
Keterangan: Data diukur pada kondisi backhaul LTE stabil, interval sampling 10 menit.				

3.2 Packet Delivery dan Konsistensi Data

Pada skenario degradasi jaringan dan cloud outage, kerangka kerja usulan mempertahankan PDR 99,2%, secara signifikan mengungguli baseline (87,5–91,3%). Sinkronisasi diferensial mengurangi volume transfer state sebesar 68%, meminimalkan kehilangan paket in-flight selama transisi. Tingkat error konsistensi data tetap di bawah 0,3%, jauh dalam ambang batas yang dapat diterima untuk aplikasi kritis.

3.3 Overhead Sumber Daya dan Skalabilitas

Pemantauan konteks dan decision engine mengonsumsi rata-rata 6,4% CPU dan 4,1% memori pada node edge, menunjukkan karakteristik operasional ringan. Skalabilitas ke 15 workload konkuren meningkatkan overhead secara linear menjadi 9,8%, mengonfirmasi kesesuaian untuk lingkungan multi-tenant kritis tanpa memerlukan klaster orkestrasi khusus.

3.4 Pembahasan Implikasi dan Keselarasan Literatur

Hasil empiris memvalidasi bahwa fallback otonom berbasis konteks secara signifikan meningkatkan ketahanan migrasi untuk workload kritis. Reduksi downtime 73% berasal dari adaptasi ambang batas proaktif dan checkpointing state asinkron, yang mencegah interupsi layanan selama degradasi berkelanjutan. Berbeda dengan model statis yang terpicu prematur selama lonjakan sementara atau bereaksi terlambat saat kegagalan aktual, model penilaian ketahanan adaptif secara dinamis menyeimbangkan sensitivitas dan stabilitas. Temuan ini sejalan dengan temuan terkini bahwa evaluasi konteks multi-dimensi mengungguli orkestrasi metrik tunggal dalam lingkungan volatil (Kumar et al., 2021; Chen et al., 2022).

Latensi aktivasi di bawah 2 detik sangat signifikan untuk komunikasi darurat dan sistem kontrol industri, di mana waktu respons deterministik bersifat non-negosiable. Pencapaian ini memerlukan optimalisasi pipa telemetri eBPF dan implementasi pembaruan routing atomik, menunjukkan bahwa pemantauan ringan dapat berdampingan dengan orkestrasi berkinerja tinggi. Overhead

sumber daya 6,4% tetap dapat diterima untuk workload kritis, terutama ketika dikontraskan dengan biaya operasional downtime atau kehilangan data selama kegagalan migrasi.

Namun, beberapa keterbatasan perlu diakui. Implementasi saat ini mengasumsikan runtime kontainer homogen (CRI-O/containerd) di seluruh node sumber dan target; heterogenitas dalam lingkungan legacy mungkin memerlukan lapisan translasi tambahan. Model ambang batas adaptif, meskipun efektif, bergantung pada penyetelan bobot empiris; penelitian selanjutnya sebaiknya mengeksplorasi federated learning untuk optimalisasi bobot dinamis di seluruh penyebaran terdistribusi. Selain itu, evaluasi konteks keamanan saat ini beroperasi pada level node; integrasi profil perilaku spesifik-workload dapat meningkatkan presisi migrasi berbasis ancaman.

Dari perspektif penyebaran, kerangka kerja ini kompatibel dengan operator Kubernetes dan arsitektur service mesh yang ada, memungkinkan adopsi bertahap tanpa perombakan infrastruktur. Lembaga respons darurat, penyedia IoT kesehatan, dan operator otomasi industri dapat mengintegrasikan orkestrator sebagai sidecar atau ekstensi control-plane untuk meningkatkan ketahanan migrasi. Upaya standardisasi sekitar format telemetri konteks (misalnya, ekstensi OpenTelemetry) akan lebih memfasilitasi interoperabilitas lintas vendor.

IV. PENUTUP

Penelitian ini berhasil merancang dan mengimplementasikan strategi fallback otonom berbasis konteks untuk migrasi workload kritis ke lingkungan cloud. Dengan mengintegrasikan pemantauan konteks multi-dimensi, penilaian ketahanan adaptif, dan orkestrasi migrasi stateful, kerangka kerja mencapai aktivasi fallback di bawah 2 detik, reduksi downtime 73%, dan packet delivery ratio 99,2% dalam kondisi volatil. Desain ringan memastikan kompatibilitas dengan kontinuitas edge-cloud sambil mempertahankan perilaku deterministik yang sesuai untuk aplikasi misi kritis. Validasi empiris mengonfirmasi bahwa migrasi proaktif berbasis konteks mengungguli strategi statis, reaktif, dan manual tanpa menimbulkan overhead sumber daya yang prohibitive. Solusi ini dapat diadopsi oleh badan penanggulangan bencana, penyedia IoT kesehatan, dan operator otomasi industri untuk meningkatkan ketahanan siber pada jaringan sensor terdistribusi. Penelitian lanjutan direkomendasikan untuk menguji skalabilitas arsitektur pada penyebaran lapangan multi-node, mengintegrasikan analisis ancaman prediktif berbasis kecerdasan buatan, serta mengevaluasi interoperabilitas dengan standar komunikasi darurat nasional dan internasional.

DAFTAR PUSTAKA

l-Garadi, M.A., Mohamed, A. and Du, X. (2021) 'Edge-assisted cloud migration for critical IoT applications: Latency and resilience trade-offs', IEEE Internet of Things Journal, 8(15), h. 12045–12058. doi: 10.1109/JIOT.2021.3064211.

Alshamrani, A., Myneni, A., Chowdhary, A. and Huang, D. (2021) 'A survey on security requirements, threats, and defense mechanisms for cloud computing', Journal of Network and Computer Applications, 193, h. 103198. doi: 10.1016/j.jnca.2021.103198.

Chen, Y., Ma, L. and Wu, H. (2022) 'Reinforcement learning for dynamic cloud migration scheduling: Overhead and explainability constraints', Springer Journal of Grid Computing, 20(2), h. 18. doi: 10.1007/s10723-022-09612-4.

Diaz-Sanchez, R., Garcia-Teodoro, P. and Lopez, J. (2022) 'Threat-integrated cloud placement and migration security: A context-aware defense model', Elsevier Computers & Security, 115, h. 102589. doi: 10.1016/j.cose.2022.102589.

Gupta, A., Patel, S. and Kumar, R. (2022) 'Differential state synchronization for containerized critical workloads: Reducing migration bandwidth overhead', Elsevier Journal of Parallel and Distributed Computing, 168, h. 45–58. doi: 10.1016/j.jpdc.2022.05.003.

Khan, I., Belgaum, M.R. and Buyya, R. (2021) 'Zero-trust security model for cloud computing: A systematic literature review', IEEE Access, 9, h. 142345–142362. doi: 10.1109/ACCESS.2021.3119871.

Kumar, S., Singh, R. and Sharma, P. (2021) 'Context-aware workload orchestration in edge-cloud environments: A multi-criteria evaluation framework', Elsevier Computer Communications, 178, h. 124–137. doi: 10.1016/j.comcom.2021.07.012.

Nguyen, T., Park, H. and Lee, J. (2021) 'Autonomous disaster recovery in hybrid cloud architectures: Replication strategies and failover optimization', IEEE Access, 9, h. 89102–89115. doi: 10.1109/ACCESS.2021.3089451.

NIST (2020) Cloud Computing Security Reference Architecture. Special Publication 500-292. Gaithersburg: National Institute of Standards and Technology. doi: 10.6028/NIST.SP.500-292.

Wang, K., Li, Y. and Zhang, X. (2022) 'Lightweight telemetry for cloud-edge continuum: eBPF-based monitoring under resource constraints', IEEE Transactions on Services Computing, 16(3), h. 1892–1905. doi: 10.1109/TSC.2021.3124567.

Wang, X., Li, X., Qiu, T., Zhang, Y. and Li, X. (2020) 'Security and privacy in cloud computing: A survey', IEEE Internet of Things Journal, 7(11), h. 10738–10755. doi: 10.1109/JIOT.2020.3003567.

Zhang, Q., Cheng, L. and Boutaba, R. (2020) 'Edge computing: State-of-the-art and future directions', IEEE Communications Surveys & Tutorials, 22(2), h. 1228–1258. doi: 10.1109/COMST.2020.2966689