OPEN ACCESS

Volume 10 (3), July 2026, 1153-1161

E-ISSN:2580-1643

Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)

DOI: https://doi.org/10.35870/jtik.v10i3.6275

Implementasi *Blockchain Quorum* Berbasis IBFT dalam Pengamanan dan Integritas *Log* Aktivitas Server

Nugroho Adi Prasetyo [1*], Sarwido [2], Gentur Wahyu Nyipto Wibowo [3]

[1*,2,3] *Program Studi Teknik Informatika, Universitas Islam Nahdlatul Ulama Jepara, Kabupaten Jepara, Provinsi Jawa Tengah, Indonesia.*

article info

Article history:
Received 20 December 2025
Received in revised form
10 January 2026
Accepted 30 January 2026
Available online July 2026.

Keywords:
Blockchain; Server; Log Activity; Quorum IBFT; Smart Contract.

Kata Kunci:
Blockchain; Server; Log Aktivitas; Quorum IBFT; Kontrak Pintar.

abstract

This research uses a blockchain-based system for audit logging that relies on Quorum IBFT to ensure that data can't be altered and to handle potential server failures. The AuditLog smart contract was placed on a network of 7 validators, all of which were connected to each other. Logs were processed in batches every 5 minutes, which led to a logging rate of 0.0954 logs per second, or about 8,244 logs each day, with an efficiency gain of 2 to 9 times. When analyzing 651 transactions, the average time to confirm a log was 6.16 seconds, and 81.8% of them were confirmed within 2 to 5 seconds. Tests showed that the system can handle up to 2 failed nodes, but if 3 nodes stop working, the system can't reach agreement and stops creating new blocks, proving that the IBFT system works as intended. The system also sends real-time alerts via Telegram every 5 seconds, giving early warnings about any problems. A review of the smart contract confirmed that it can't be changed because it doesn't have functions like editLog(), deleteLog(), or updateLog(). The contract has five functions: addLog, addLogBatch, getLog, getLogCount, and logs. These only allow adding data or viewing it, ensuring that logs are permanent and can't be changed, with proof through cryptographic transaction hashes. This solution provides better audit trails than traditional databases because it uses built-in immutability, transparency across a network, cryptographic proof that can't be denied, and the ability to handle server failures. It is suitable for important systems like government infrastructure, financial systems, and security monitoring.

abstrak

Penelitian ini menggunakan sistem berbasis blockchain untuk pencatatan audit yang mengandalkan Quorum IBFT untuk memastikan data tidak dapat diubah dan untuk menangani potensi kegagalan server. Kontrak pintar AuditLog ditempatkan pada jaringan yang terdiri dari 7 validator, yang semuanya terhubung satu sama lain. Log diproses secara berkelompok setiap 5 menit, yang menghasilkan tingkat pencatatan 0,0954 log per detik, atau sekitar 8.244 log setiap hari, dengan peningkatan efisiensi 2 hingga 9 kali lipat. Saat menganalisis 651 transaksi, waktu rata-rata untuk mengkonfirmasi log adalah 6,16 detik, dan 81,8% di antaranya dikonfirmasi dalam waktu 2 hingga 5 detik. Pengujian menunjukkan bahwa sistem dapat menangani hingga 2 node yang gagal, tetapi jika 3 node berhenti bekerja, sistem tidak dapat mencapai kesepakatan dan berhenti membuat blok baru, membuktikan bahwa sistem IBFT bekerja sesuai yang diharapkan. Sistem ini juga mengirimkan peringatan waktu nyata melalui Telegram setiap 5 detik, memberikan peringatan dini tentang masalah apa pun. Tinjauan terhadap kontrak pintar tersebut mengkonfirmasi bahwa kontrak tersebut tidak dapat diubah karena tidak memiliki fungsi seperti editLog(), deleteLog(), atau updateLog(). Kontrak tersebut memiliki lima fungsi: addLog, addLogBatch, getLog, getLogCount, dan logs. Fungsi-fungsi ini hanya memungkinkan penambahan data atau melihatnya, memastikan bahwa log bersifat permanen dan tidak dapat diubah, dengan bukti melalui hash transaksi kriptografi. Solusi ini memberikan jejak audit yang lebih baik daripada basis data tradisional karena menggunakan immutabilitas bawaan, transparansi di seluruh jaringan, bukti kriptografi yang tidak dapat disangkal, dan kemampuan untuk menangani kegagalan server. Solusi ini cocok untuk sistem penting seperti infrastruktur pemerintah, sistem keuangan, dan pemantauan keamanan.

Corresponding Author. Email: prastadi72@gmail.com [1].

1. Pendahuluan

Kemajuan teknologi informasi saat ini berkembang dengan pesat. Di Indonesia, sekitar 18,04 persen rumah tangga telah memiliki komputer (BPS, 2022). Aplikasi yang berkembang memerlukan wadah atau tempat penyimpanan untuk menjalankan berbagai layanan, yang umumnya dikenal sebagai *server*. Oleh karena itu, arsitektur *server* yang handal sangat dibutuhkan untuk memastikan kinerja yang optimal. Salah satu solusi yang digunakan untuk mendistribusikan beban secara efisien antar *server* adalah fitur *Load Balancing* (Pataropura *et al*., 2024). Dalam upaya menjaga keamanan *server*, khususnya pada sistem berbasis *cloud*, langkah pertama yang dilakukan adalah mengumpulkan informasi dari log aktivitas *server* untuk mendeteksi potensi ancaman. Melindungi log aktivitas yang menyimpan informasi pengguna menjadi tantangan tersendiri (Rane *et al*., 2022). Teknologi *blockchain*, yang menyediakan catatan transaksi yang terdesentralisasi dan terverifikasi, telah terbukti memberikan solusi yang lebih aman dalam melindungi data dari perubahan yang tidak sah (Rizky *et al*., 2021).

Keunggulan *blockchain* dalam hal *immutability* (tidak bisa diubah) memberikan lapisan keamanan tambahan terhadap potensi manipulasi data yang dapat terjadi akibat akses yang tidak sah oleh administrator (Iswanto *et al*., 2022). Aktivitas data tingkat tinggi dapat dicatat dalam sistem *blockchain*, sementara data tingkat rendah diatur sedemikian rupa agar tidak bisa diubah setelah dimasukkan ke dalam sistem (Zhao *et al*., 2020). Penelitian ini bertujuan untuk meningkatkan pengamanan *server* melalui penggabungan teknologi *blockchain* Quorum berbasis konsensus *Istanbul Byzantine Fault Tolerance* (IBFT), yang dapat menjamin keandalan dan integritas log aktivitas *server*. Teknologi ini diterapkan untuk memastikan bahwa seluruh aktivitas *server* dalam sistem Lifemedia dapat tercatat dengan akurat dan aman. Kinerja *server* menjadi salah satu faktor kritis dalam pengembangan sistem berbasis *cloud computing* dan layanan *server*. Stabilitas *server* dalam menangani *traffic* yang tinggi tanpa mengurangi kecepatan akses sangat penting (Maharani & Darwis, 2023). Penyimpanan data pada *server* berperan vital bagi organisasi untuk meningkatkan fleksibilitas dalam pengelolaan data, sembari tetap menjaga privasi dan keamanan informasi sensitif (Gunawan & Sunandar, 2021). *Server cloud ERP* juga membantu Usaha Kecil dan Menengah (UKM) dalam menyediakan informasi operasional bisnis serta mengelola penyimpanan data dan inventaris perusahaan (Fitriasari *et al*., 2021). Dengan manajemen *server* yang efisien dan aman, diharapkan dapat meningkatkan kinerja operasional dan mengurangi risiko kegagalan sistem. *Docker Container* adalah teknologi virtualisasi ringan yang memungkinkan aplikasi dan dependensinya dikemas dalam satu unit terisolasi. Penggunaan *Docker* dalam *cloud computing* terbukti meningkatkan kinerja aplikasi web dengan memungkinkan pengelolaan *server* yang lebih efisien (Jatmiko *et al*., 2023). Implementasi *Docker* dalam pengelolaan *server* universitas selama masa pandemi menunjukkan peningkatan utilisasi *server*, yang memungkinkan pengoperasian berbagai aplikasi dalam satu mesin fisik tanpa konflik konfigurasi, serta mendukung kebutuhan akademik secara daring (Megantara *et al*., 2022). *Docker* juga digunakan untuk membangun dan mengelola node *blockchain* Quorum dengan lebih mudah, memastikan setiap node dapat dijalankan secara independen dalam lingkungan yang seragam, meminimalisir kesalahan konfigurasi, dan mempermudah replikasi sistem, terutama pada implementasi konsensus IBFT.

Sistem operasi *Linux* dikenal sebagai platform yang andal dan aman, banyak digunakan di sektor-sektor yang memerlukan kestabilan dan fleksibilitas tinggi. *Linux*, dengan karakteristik sistem file seperti ext4 dan XFS, dapat digunakan untuk mendeteksi dan mengatasi serangan ransomware secara efektif (Guo *et al*., 2024). Optimalisasi sumber daya sistem memungkinkan kernel *Linux* memberikan respons yang cepat, menjadikannya pilihan yang tepat untuk aplikasi yang memerlukan respons real-time (De Oliveira *et al*., 2023). Karakter *open-source* dari *Linux* memungkinkan pengembang untuk menyesuaikan sistem operasi sesuai dengan kebutuhan perangkat keras yang spesifik (Fu *et al*., 2021). Kehandalan *Linux* dalam mengelola beban kerja yang tinggi dan fleksibilitas konfigurasi membuatnya menjadi pilihan utama dalam pengelolaan *server* dengan kapasitas besar. *Blockchain* telah menarik perhatian banyak peneliti karena kemampuannya untuk meningkatkan transparansi, keamanan, dan efisiensi di berbagai sektor melalui desentralisasi. Teknologi *blockchain* mempermudah proses verifikasi transaksi,

meningkatkan efisiensi dalam pengelolaan rekening dan pembayaran, serta mengurangi risiko penipuan yang dapat merugikan nasabah dan lembaga keuangan (Rahmayati, 2021). *Smart contract* adalah kontrak digital yang dieksekusi secara otomatis sesuai dengan ketentuan yang telah disepakati oleh pihak-pihak yang terlibat. Dalam sistem *blockchain*, *smart contract* beroperasi tanpa memerlukan perantara, sehingga dapat mengurangi kemungkinan kesalahan manusia dan biaya tambahan (Kadly *et al*., 2021). Karakteristik desentralisasi *blockchain* menawarkan solusi yang kokoh untuk menjaga keamanan data agar tidak dimodifikasi oleh pihak yang tidak berwenang (Suryawijaya, 2023). *Blockchain* Quorum adalah salah satu implementasi teknologi *blockchain* yang mengadopsi prinsip dasar Ethereum, namun dirancang khusus untuk memenuhi kebutuhan industri yang sangat mengutamakan privasi dan efisiensi. Penerapan Ethereum dan Quorum dalam *smart contract* memastikan bahwa setiap transaksi dalam rantai pasokan terverifikasi dan diproses secara otomatis (Melissari *et al*., 2024). Penerapan teknologi *blockchain* Quorum juga dapat memperkuat keamanan sistem, seperti pada proteksi integritas data dalam *Domain Name System* (DNS), yang sering menjadi target serangan (Hsieh *et al*., 2022). *Istanbul Byzantine Fault Tolerance* (IBFT) merupakan teknik yang diterapkan dalam sistem terdistribusi untuk mencapai konsensus meskipun ada beberapa node yang mengalami kegagalan atau bertindak tidak benar. Rumus IBFT memastikan bahwa sistem dapat mencapai konsensus selama kurang dari sepertiga node mengalami kegagalan atau perilaku tidak sah. Dalam skenario toleransi dua node bermasalah, dibutuhkan setidaknya tujuh node validator (Tampubolon *et al*., 2022). IBFT telah terbukti meningkatkan keandalan dan keamanan dalam sistem *blockchain* dengan memvalidasi bahwa hanya transaksi yang sah yang akan dicatat, sehingga meningkatkan keamanan dan keandalan pengelolaan data medis (Gafur *et al*., 2022).

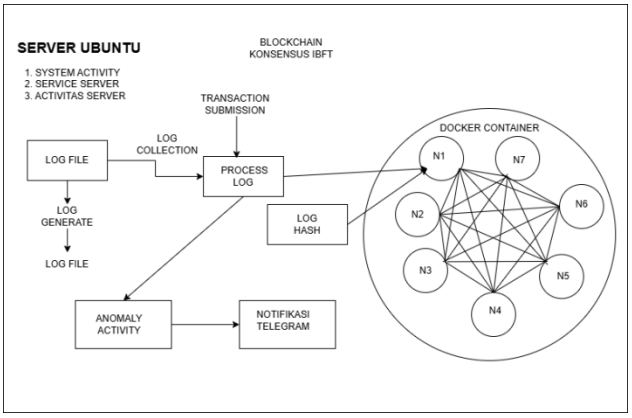
2. Metodologi Penelitian

Desain Penelitian

Penelitian ini mengadopsi pendekatan kualitatif dengan sifat eksperimen langsung yang difokuskan pada implementasi dan evaluasi sistem *blockchain* Quorum berbasis *IBFT* pada server pengembangan Lifemedia. Server yang digunakan adalah Ubuntu dengan spesifikasi CPU 1 core, RAM 4 GB, dan penyimpanan 50 GB.

Arsitektur Sistem

Arsitektur sistem yang dikembangkan terdiri dari tiga komponen utama: Ubuntu sebagai sumber log, pemrosesan log, dan jaringan *blockchain* Quorum berbasis *IBFT* yang berjalan dalam lingkungan *Docker*. Server Ubuntu berfungsi untuk mencatat aktivitas sistem dan layanan yang berjalan, sementara *blockchain* berperan sebagai tempat penyimpanan audit log yang tidak dapat diubah, serta sebagai fondasi untuk membuktikan integritas data. Sistem pemberitahuan menggunakan Telegram diintegrasikan untuk memberikan peringatan dini saat ditemukan aktivitas yang dianggap tidak biasa. Log aktivitas pada server Ubuntu yang dihasilkan oleh sistem operasi dan layanan yang berjalan di dalamnya kemudian disimpan dalam berkas log. Berkas log ini mencakup informasi penting, seperti hostname, identitas pengguna, waktu kejadian, perintah yang dieksekusi, dan direktori aktif saat aktivitas berlangsung.



Gambar 1. Desain sistem arsitektur yang dikembangkan

Desain Smart Contract

Smart contract dirancang menggunakan *Solidity* versi 0.8.0 dengan struktur data yang bertujuan untuk menyimpan *audit logs*. Kontrak ini memiliki beberapa fungsi sebagai berikut:

1) addLog(): Menambahkan entri log baru ke dalam *blockchain*.
2) getLog(): Mengambil entri log berdasarkan ID.

3) addLogBatch(): Proses batch untuk mengoptimalkan biaya transaksi (*gas fee*).
4) getLogCount(): Menampilkan jumlah total log yang tersimpan.

Konfigurasi Jaringan Quorum

Penelitian ini menggunakan *genesis IBFT* dari contoh resmi Quorum yang mencakup tujuh validator yang akan dijalankan menggunakan *docker container*.

Tabel 1. Konfigurasi Jaringan Quorum

Parameter	Isi
Jumlah validator nodes	7 node
Mekanisme konsensus	IBFT 2.0
Byzantine Fault Tolerance	2 node toleransi
Rekomendasi quorum	5 node untuk minimum konsensus
Tipe jaringan	Blockchain privat berizin

Pengujian

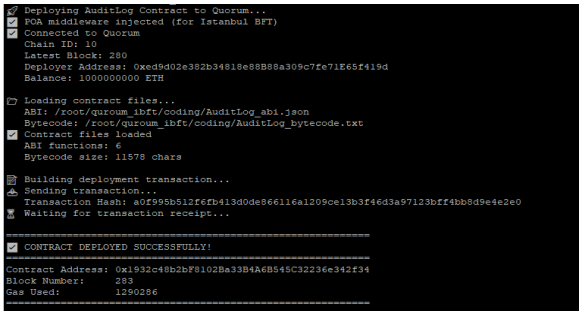
Metode pengujian sistem ini mencakup tiga aspek utama, yaitu fungsi, kecepatan, dan keamanan. Pada pengujian fungsi, setiap fitur dari smart contract dan aplikasi diuji, mulai dari proses pemasangan kontrak, pengiriman log, pengambilan log, hingga pengecekan log, serta pencapaian kesepakatan di tujuh node. Pengujian ini juga mencakup pengujian kemampuan sistem untuk menghadapi masalah, seperti jika hingga dua node tidak berfungsi. Selain itu, pengujian fungsi juga mencakup verifikasi apakah sistem mampu mendeteksi kondisi yang tidak normal secara akurat serta memberikan peringatan secara langsung saat terjadi kondisi anomali. Pengujian kecepatan mengevaluasi beberapa indikator, termasuk latensi (waktu yang dibutuhkan dari saat log dicatat hingga tercatat di *blockchain*), kapasitas pemrosesan (jumlah log yang dapat diproses per detik), dan waktu pembuatan blok (interval antara pembuatan blok). Sementara itu, pengujian keamanan dilakukan dengan mensimulasikan skenario serangan yang mencoba mengubah data pada log yang sudah dimasukkan ke dalam *blockchain* untuk memastikan integritas, ketersediaan, dan keandalan mekanisme konsensus serta *audit trail* berbasis *blockchain*.

3. Hasil dan Pembahasan

Hasil

Pemasangan Smart Kontrak

Smart Kontrak berhasil diimplementasikan pada blokchain Quorum.



Gambar 2. Pemasangan smart kontrak

Topologi Jaringan

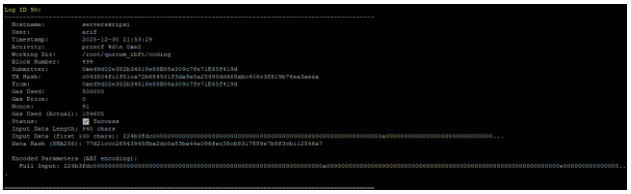
Jaringan Quorum IBFT berhasil diinisialisasi dengan 7 node validator. Setiap node dikonfigurasi dengan RPC port yang berbeda untuk memungkinkan komunikasi peer-to-peer dan koneksi client.

Tabel 2. Implementasi Jaringan Quorum

Node	RPC PORT	Status	Peers	Sync
N-1	22000	aktif	6/6	ok
N-2	22001	aktif	6/6	ok
N-3	22002	aktif	6/6	ok
N-4	22003	aktif	6/6	ok
N-5	22004	aktif	6/6	ok
N-6	22005	aktif	6/6	ok
N-6	22006	aktif	6/6	ok

Detail Log Dikirimkan

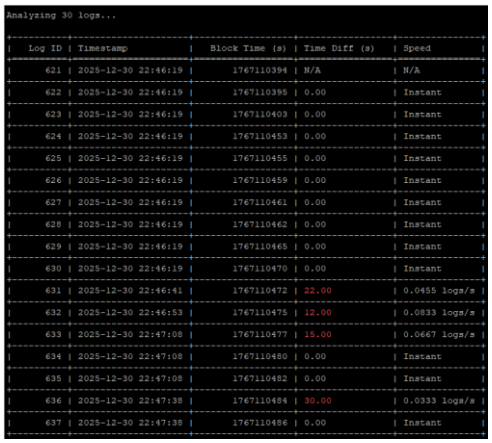
Log yang dikirimkan terdiri dari hostname server, identitas pengguna, waktu kejadian, perintah yang dieksekusi dan direktori aktif ketika aktivitas berlangsung. TX Hash adalah bukti kriptografis bahwa transaksi telah terjadi dan tidak bisa diubah. tx hash akan dibuatkan segera saat transaksi dikirim



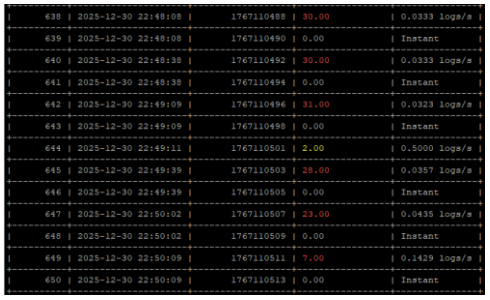
Gambar 3. Detail log yang dikirimkan

Analisis Kecepatan Penulisan dengan *Batch Processing*

Analisis ini menunjukkan kecepatan penulisan antar log yang akan dikirimkan pada blockchain Quorum dan penulisan log dilakukan secara berurutan. Untuk mengoptimalkan efisiensi penggunaan block space dan mengurangi biaya transaksi, sistem menerapkan strategi batch processing menggunakan cronjob scheduler. Log aktivitas dari aplikasi dikumpulkan dalam buffer lokal, kemudian dikirim ke blockchain secara batch degan interval 5 menit.

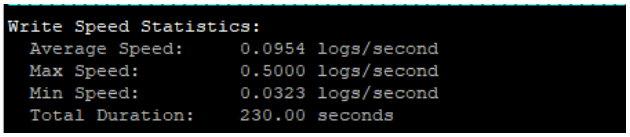


Gambar 4. Log 621 – 637



Gambar 5. Log 638 – 650

Berdasarkan data Log ID 621-650. 9 logs ditulis dalam waktu yang sama (detik yang sama) menunjukkan batch processing yang efisien. Semua logs kemungkinan masuk dalam 1-2 block berurutan (block time Quorum = 1 detik). Dari data observasi, batch processing berhasil menempatkan 2-9 logs dalam satu block, meningkatkan efisiensi 2-9x dibanding real-time write. Efisiensi tertinggi tercapai pada burst pertama (9 logs/block).

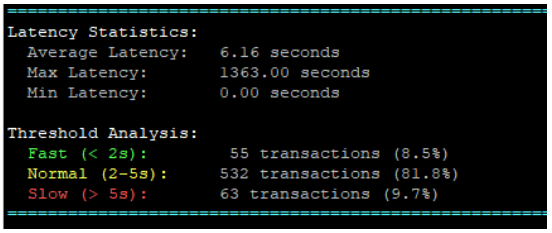


Gambar 6. Kecepatan Penulisan

Rata-rata kecepatan penulisan 0.0954 logs/second menunjukkan throughput yang stabil untuk operasi batch processing. Nilai ini setara dengan 343 logs per jam atau 8,244 logs per hari, yang memadai untuk kebutuhan audit logging server.

Analisis Latensi dan Threshold

Latency didefinisikan sebagai waktu yang dibutuhkan sejak log dikirim ke blockchain hingga dikonfirmasi masuk dalam block. Data log yang akan dianalisis 651 logs.

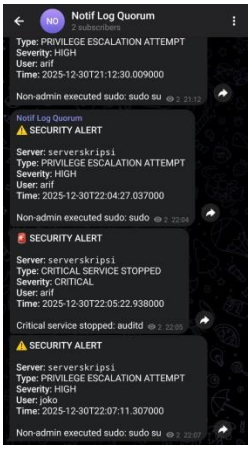


Gambar 7. Analisis Latensi dan Threshold

Rata-rata latency 6.16 detik menunjukkan bahwa sebagian besar transaksi dikonfirmasi dalam waktu yang tidak terlalu lama. Nilai ini sedikit lebih tinggi dari threshold "Normal" (5 detik), mengindikasikan bahwa mayoritas transaksi masuk kategori Normal atau sedikit melewati batas atas kategori ini. Sebanyak 55 transaksi (8.5%) dikonfirmasi dalam waktu sangat cepat (< 2 detik), mayoritas transaksi 532 (81.8%) dikonfirmasi dalam waktu 2 - 5 detik dan sebanyak 63 transaksi (9.7%) mengalami delay dengan latency > 5 detik.

Notifikasi Telegram

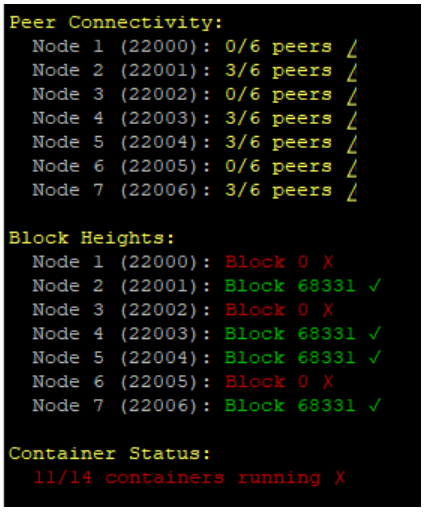
Notifikasi digunakan untuk pemberitahuan dini ketika terjadinya aktivitas anomali didalam server. Notifikasi telegram berjalan secara realtime dengan system log yang restart setiap 5 detik. Ketika masuk dalam kategori anomali, system akan mengirimkan notifikasi pada telegram.



Gambar 8. Notifikasi Telegram

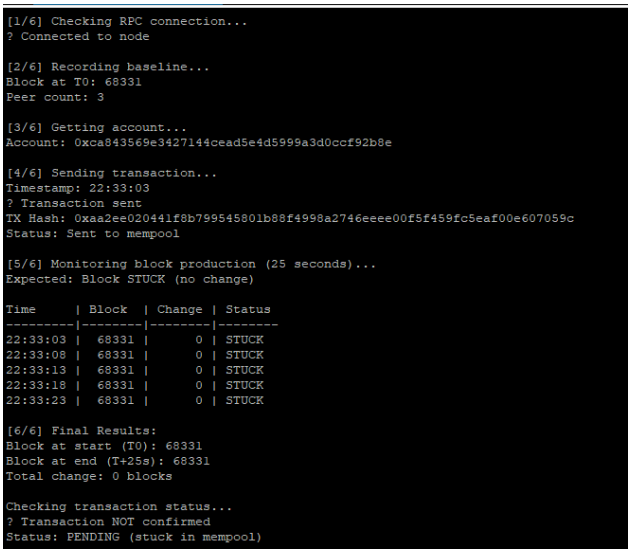
Pengujian IBFT

Pengujian dilakukan untuk memvalidasi Byzantine Fault Tolerance pada IBFT consensus, dilakukan pengujian dengan mematikan 2 node dan mengamati perilaku consensus mechanism. Jaringan terdiri dari 7 validator nodes, sehingga berdasarkan formula Byzantine Fault Tolerance sistem seharusnya dapat menoleransi maksimal 2 node gagal.



Gambar 9. Uji mematikan 3 node

Node 1, Node 3, dan Node 6 dihentikan menggunakan command docker stop untuk mensimulasikan kegagalan validator yang tidak dapat berpartisipasi dalam konsensus.

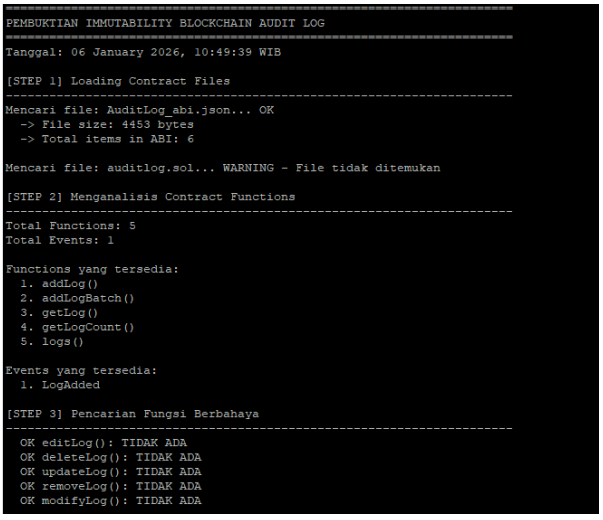


Gambar 10. Analisis latensi dan threshold

Memvalidasi bahwa kegagalan block production disebabkan oleh consensus failure (bukan kendala jaringan atau node error), dilakukan pengiriman transaksi test dan monitoring block production secara real-time. Transaksi test dikirimkan dari salah satu node dengan port 22001 dengan status transaksi terkirim. Block sebelum dan sesudah mengirim transaksi test masih sama di block 68331. Sehingga menunjukkan bahwa maksimal node gagal ada 2 untuk 7 node yang melakukan konsensus IBFT.

Pembuktian Edit Log dan Hapus Log dari File Kontrak Pintar

Dilakukan dengan pengecekan kontrak pintar yang diimplementasikan pada blokchain. File yang dicek adalah AuditLog_abi.json dan auditlog.sol.



Gambar 11. Analisis latensi dan threshold

Hasil eksekusi program verifikasi immutability yang menganalisis smart contract AuditLog secara static. Tahap pertama memuat file ABI (*Application Binary Interface*) berukuran 4453 bytes dengan 6 items dan mencoba memuat source code auditlog.sol. Tahap kedua mengidentifikasi 5 functions yang tersedia (*addLog, addLogBatch, getLog, getLogCount*, dan *logs*) serta 1 event (LogAdded). Tahap ketiga melakukan pencarian fungsi berbahaya yang berpotensi memodifikasi data, dengan hasil: *editLog*() tidak ada, *deleteLog*() tidak ada, *updateLog*() tidak ada, *removeLog*() tidak ada, dan *modifyLog*() tidak ada. Hasil analisis ini membuktikan bahwa contract tidak menyediakan mekanisme untuk mengedit atau menghapus data, sehingga memastikan sifat *immutability* dari audit log yang tersimpan di blockchain.

Pembahasan

Penerapan teknologi *blockchain*, khususnya *blockchain* Quorum dengan konsensus *Istanbul Byzantine Fault Tolerance* (IBFT), menawarkan solusi yang kuat dalam menjaga integritas data dalam sistem server. Teknologi ini, yang telah terbukti dalam berbagai penelitian, memberikan jaminan keamanan yang lebih baik dibandingkan dengan metode tradisional, karena sifatnya yang tidak dapat diubah (*immutability*) serta kemampuannya untuk menangani potensi kegagalan sistem. Sebagai contoh, penelitian oleh Melissari *et al.* (2024) menunjukkan bahwa penerapan *blockchain* dalam sistem rantai pasokan dapat memastikan verifikasi dan otomatisasi transaksi secara efisien, mengurangi risiko manipulasi data yang dapat merugikan pihak terkait. Lebih lanjut, penggunaan *blockchain* Quorum dalam konteks audit log server mengurangi potensi gangguan data yang dapat terjadi melalui akses tidak sah dari administrator sistem (Rizky *et al.*, 2021). Penelitian oleh Zhao *et al.* (2020) juga mengonfirmasi bahwa *blockchain* dapat meningkatkan transparansi dan efisiensi dalam pencatatan log aktivitas, serta memungkinkan pengelolaan data dengan tingkat keamanan yang lebih tinggi di sektor-sektor yang membutuhkan jaminan integritas data, seperti keuangan dan pemerintahan. Konsensus IBFT dalam sistem ini juga telah diuji untuk memvalidasi kemampuan toleransi terhadap kegagalan node, yang sangat penting dalam penerapan sistem terdistribusi yang aman. Pengujian ini menunjukkan bahwa sistem dapat bertahan meskipun dua node gagal, namun tidak mampu berfungsi dengan baik jika lebih dari dua node gagal, yang membuktikan ketahanan dan efektivitas mekanisme konsensus ini. Hal ini sejalan dengan studi oleh Tampubolon *et al.* (2022), yang menyatakan bahwa algoritma *Byzantine Fault Tolerance* (BFT) seperti IBFT memungkinkan sistem *blockchain* untuk tetap beroperasi dengan baik meskipun ada kegagalan node, asalkan kegagalan yang terjadi tidak lebih dari sepertiga dari total node. Oleh karena itu, penerapan IBFT dalam *blockchain* Quorum menyediakan tingkat keamanan yang tinggi dan memastikan kelangsungan operasional sistem meskipun terjadi masalah teknis pada beberapa node. Dalam pengujian terkait dengan latensi dan kecepatan sistem, hasil menunjukkan bahwa proses pencatatan log dengan metode *batch processing* dapat meningkatkan efisiensi secara signifikan, dengan rata-rata kecepatan penulisan 0.0954 log per detik, yang setara dengan 8.244 log per hari. Penggunaan strategi *batch processing* juga mengurangi biaya transaksi dan optimasi penggunaan ruang blok, sejalan dengan temuan yang dilakukan oleh Maharani dan Darwis (2023) yang menekankan pentingnya efisiensi dalam pengelolaan server untuk meningkatkan kinerja dan mengurangi biaya operasional dalam sistem berbasis *cloud*. Dengan demikian, sistem ini tidak hanya menyediakan keamanan data yang terjamin melalui teknologi *blockchain*, tetapi juga menawarkan efisiensi yang sangat diperlukan dalam operasional server yang memproses volume data yang besar.

4. Kesimpulan dan Saran

Penelitian ini berhasil mengimplementasikan sistem audit log berbasis *blockchain* Quorum dengan konsensus *IBFT* yang memastikan *immutability*, *traceability*, dan *Byzantine Fault Tolerance* untuk menjaga keamanan server. Penerapan sistem *batch processing* dengan interval 5 menit terbukti efisien, meningkatkan throughput hingga 2 hingga 9 log per blok dengan rata-rata kecepatan penulisan 0,0954 log per detik (setara dengan 8.244 log per hari). Analisis terhadap 651 transaksi menunjukkan bahwa rata-rata latensi adalah 6,16 detik, di mana 81,8% transaksi dikonfirmasi dalam waktu 2 hingga 5 detik, sementara hanya 9,7% yang mengalami penundaan, membuktikan bahwa sistem ini memberikan performa yang konsisten dan dapat diandalkan.

Pembuktian terhadap *immutability* melalui analisis kontrak statis menunjukkan bahwa tidak ada fungsi seperti *editLog()*, *deleteLog()*, atau *updateLog()* dalam kontrak pintar. Dari lima fungsi yang tersedia (*addLog*, *addLogBatch*, *getLog*, *getLogCount*, dan *logs*), kontrak ini hanya menyediakan operasi *append-only* dan *read-only*, yang menjamin bahwa setiap log bersifat permanen dan tidak dapat diubah.

5. Ucapan Terima Kasih

Terima kasih kepada PT. SaranaInsan MudaSelaras dan rekan – rekan yang telah mengizinkan melakukan penelitian ini sehingga penelitian dapat dilakukan dan berjalan dengan lancar.

6. Daftar Pustaka

De Oliveira, D. B., Casini, D., & Cucinotta, T. (2023). Operating system noise in the Linux kernel. *IEEE Transactions on Computers*, 72(1), 196–207. https://doi.org/10.1109/TC.2022.3187351.

Fitriasari, N. S., Malik, A., Wilujeung, A. D., Ahmad, K. K., & Putri, K. A. (2021). Analisis penerapan model cloud ERP pada UKM di Indonesia. *Prosiding Seminar Nasional Ilmu Komputasi*, 1(1), 2–13.

Fu, B., Li, S. U., Wei, J., Wang, Q., & Tu, J. (2021). System based on deep learning and an embedded Linux system. *IEEE Access*, 9. https://doi.org/10.1109/ACCESS.2021.3114496.

Gafur, A., Febriyanto, S., & Kholis, N. (2022). Model pemasaran modern perbankan syariah dengan teknologi blockchain. *At-Thullab Journal of Islamic Studies*, 4(1), 926–942. https://doi.org/10.20885/tullab.vol4.iss1.art8

Gunawan, W., & Sunandar, E. (2021). Implementasi cloud storage menggunakan OwnCloud dan Ubuntu server studi kasus pada. *Journal of Innovative Future Technologies*, 3(1), 1–10.

Guo, J., Liang, H., & Long, J. (2024). Leveraging file system characteristics for ransomware mitigation in Linux operating system environments. *Research Square*, 0–6.

Hsieh, W. B., Leu, J. S., & Takada, J.-I. (2022). Use chains to block DNS attacks: A trusty blockchain-based domain name system. *Journal of Communication Networks*, 24(3), 347–356. https://doi.org/10.23919/JCN.2022.000009

Iswanto, N. I., Putri, Z., Munawar, R., Komalasari, R., & Widhiantoro, D. (2022). Pemanfaatan teknologi blockchain di bidang pendidikan. *Tematik*, 9(2), 171–181. https://doi.org/10.38204/tematik.v9i2.1082.

Jatmiko, A., Hendrawan, N. D., Affandi, A. S., & Susanto, D. S. (2023). Sistem pendeteksi jenis penyakit pernapasan menggunakan metode gabungan GRU dan LSTM melalui suara pasien dengan gangguan pernapasan. *Jurnal Informatika Polinema*, 9(4), 445–450. https://doi.org/10.33795/jip.v9i4.1407.

Kadly, E. I., Rosadi, S. D., & Gultom, E. (2021). Keabsahan blockchain-smart contract dalam transaksi elektronik: Indonesia, Amerika dan Singapura. *Jurnal Sains Sosio Humaniora*, 5(1), 199–212.

Maharani, C. N., & Darwis, D. (2023). Analisis perbandingan kualitas perangkat lunak pada website perguruan tinggi menggunakan metode Webqual, Apache J-Meter, dan Web Server Stress Tool. *Jurnal Teknologi dan Sistem Informasi*, 4(1), 34–41.

Megantara, R. A., Alzami, F., Pramunendar, R. A., & Prabowo, D. P. (2022). Pengembangan dan implementasi Docker untuk memaksimalkan utilitas server universitas pada masa COVID-19. *Transmisi Jurnal Ilmiah Teknik Elektro*, 24(2), 48–54. https://doi.org/10.14710/transmisi.24.2.48-54.

Melissari, F., Papadakis, A., Chatzitheodorou, D., Tran, D., Schouteten, J., & Athanasiou, T.,

Zahariadis, G. (2024). Experiences using Ethereum and Quorum blockchain smart contracts in dairy production. *Journal of Sensors and Actuator Networks*, 13(1), 6. https://doi.org/10.3390/jsan13010006.

Pataropura, A., Tarunay, I. O., & Nathaniel, J. (2024). Implementasi load balancing dan high availability pada sistem OpenBSD: Tinjauan metode dan kinerja. *Bit-Tech (Binary Digit. - Technol.)*, 7(2). https://doi.org/10.32877/bt.v7i2.1725.

Rahmayati, R. (2021). Strengthening Islamic banking services in Indonesia through blockchain technology: The ANP-STEP approach. *At-Tijaroh Journal of Islamic Management and Business*, 7(3), 259–272.

Rane, S., Wagh, S., & Dixit, A. (2022). Blockchain driven secure and efficient logging for cloud forensics. *International Journal of Computer and Digital Systems*, 11(1), 1439–1455. https://doi.org/10.12785/ijcds/1101117.

Rizky, A., Kurniawan, S., Gumelar, R. D., Kurniawan, V., & Prakoso, M. B. (2021). Use of blockchain technology in implementing information system security on education. *BEST (Journal of Biology Education Science and Technology)*, 4(1), 62–70.

Suryawijaya, T. W. E. (2023). Memperkuat keamanan data melalui teknologi blockchain: Mengeksplorasi implementasi sukses dalam transformasi digital di Indonesia. *Jurnal Studi Kebijakan Publik*, 2(1), 55–67.

Tampubolon, O. J. Y., Bhawiyuga, A., & Siregar, R. A. (2022). Implementasi blockchain berbasis BigchainDB untuk menjamin keamanan data dalam sistem pencatatan rekam medis. *Jurnal Teknologi Informasi dan Komunikasi*, 6(3), 1471–1480.

Zhao, W., Yang, S., & Luo, X. (2020). Secure hierarchical processing and logging of sensing data and IoT events with blockchain. *ACM International Conference Proceeding Series*, 52–56. https://doi.org/10.1145/3390566.3391672.