Polygon: Jurnal Ilmu Komputer dan Ilmu Pengetahuan Alam
Volume. 3 Nomor. 6 November 2025
e-ISSN: 3046-5419; p-ISSN: 3032-6249, Hal 50-59
DOI: https://doi.org/10.62383/polygon.v3i6.817
Tersedia: https://journal.arimsi.or.id/index.php/Polygon

Penerapan Hubungan *Hardware-Software* pada Sistem Operasi Android dalam Pengelolaan Sensor *Fingerprint*

Annisa Laili Tanzila[1*], Aldi Muhammad Reski[2]
[1-2]Program Studi Manajemen Informatika, Akademi Manajemen Informatika dan Komputer (AMIK) Bukittinggi, Indonesia
**Penulis korespondensi: ichaaa405@gmail.com*

Abstract. *Android continues to innovate, and biometric systems particularly fingerprint sensors must operate with high efficiency. However, the main challenge lies in the communication between hardware and software, which often causes high latency, excessive power consumption, and protocol incompatibility between components. As a result, authentication speed decreases and system stability is compromised, especially on mid- to low-range devices. This study explores various implementations of Android-based fingerprint systems, focusing on how sensor modules interact with the microcontroller or Trusted Execution Environment (TEE) and the operating system through a hardware-software co-design approach to evaluate integration efficiency across all layers. The analysis reveals that conventional protocols such as Bluetooth or serial connections still cause delays, while improvements in drivers and the Hardware Abstraction Layer (HAL) can significantly reduce latency. As a solution, the researchers propose a co-design optimization approach that utilizes data flow normalization within the HAL and adopts lightweight communication protocols to accelerate the verification process. Based on the test results, this approach successfully improves efficiency—authentication time is reduced by up to 35% and power consumption decreases by approximately 15%. Therefore, the efficiency of communication between hardware and software becomes a key factor in enhancing the performance and reliability of fingerprint systems on Android devices.*

Keywords: *Android; Biometric Authentication; Fingerprint Sensor; Hardware Software; System Efficiency.*

Abstrak. Android terus berinovasi, sistem biometrik khususnya sensor sidik jari (fingerprint) harus beroperasi dengan efisiensi yang tinggi. Namun, tantangan utama masih terletak pada komunikasi antara perangkat keras dan perangkat lunak yang sering kali menimbulkan latency tinggi, konsumsi daya berlebih, serta ketidaksesuaian protokol antar komponen. Akibatnya, kecepatan autentikasi menurun dan stabilitas sistem terancam, terutama pada perangkat kelas menengah ke bawah. Studi ini menelusuri beragam penerapan sistem sidik jari berbasis Android dengan menitikberatkan pada cara modul sensor berinteraksi dengan mikrokontroler atau Trusted Execution Environment (TEE) serta sistem operasi melalui pendekatan hardware-software co-design, guna menilai efisiensi integrasi di setiap lapisan. Analisis mengungkap bahwa protokol konvensional seperti Bluetooth atau koneksi serial masih menimbulkan penundaan, sementara penyempurnaan driver dan lapisan abstraksi hardware (HAL) dapat secara signifikan memangkas latensi. Sebagai solusi, peneliti mengusulkan penerapan co-design optimization yang memanfaatkan normalisasi alur data pada HAL serta mengadopsi protokol komunikasi ringan untuk mempercepat proses verifikasi. Berdasarkan hasil pengujian, pendekatan ini berhasil meningkatkan efisiensi secara signifikan—waktu autentikasi berkurang hingga 35 % dan konsumsi daya turun sekitar 15 %. Dengan demikian, efisiensi komunikasi antara hardware dan software menjadi faktor utama dalam mendorong peningkatan performa serta keandalan sistem fingerprint pada perangkat Android..

Kata kunci: Android; Autentikasi Biometrik; Efisiensi Sistem; Perangkat Keras dan Perangkat Lunak; Sensor Sidik Jari.

1. LATAR BELAKANG

Dengan meningkatnya kebutuhan akan sistem autentikasi biometrik yang cepat dan efisien, penelitian tentang kecepatan komunikasi perangkat lunak dan perangkat keras pada sistem operasi Android terus berlanjut. Sebuah studi sebelumnya menunjukkan bahwa faktor utama yang menyebabkan proses autentikasi sidik jari tertunda adalah kompleksitas lapisan komunikasi antara sensor, Hardware Abstraction Layer (HAL), dan sistem operasi (Fitriani & Pratama, 2020). Penelitian lainnya menunjukkan bahwa pendekatan co-design hardware-

Naskah Masuk: 25 September 2025; Revisi: 09 Oktober 2025; Diterima: 26 Oktober 2025; Tersedia: 29 Oktober 2025

software yang mengoptimalkan jalur komunikasi internal dapat meningkatkan efisiensi dan stabilitas sistem biometrik.oleh (Chen et al, 2022)

Permasalahan utama dalam penelitian ini berfokus pada ketidakefisienan interaksi antara perangkat keras dan perangkat lunak dalam sistem operasi Android, khususnya pada proses autentikasi biometrik menggunakan sensor sidik jari. Pada sistem Android, komunikasi antara Fingerprint Sensor Module, Hardware Abstraction Layer (HAL), Trusted Execution Environment (TEE), dan Android Biometric Framework berjalan secara berlapis. Namun, dalam praktiknya, proses pertukaran data antar lapisan tersebut sering kali mengalami hambatan akibat latensi tinggi, konsumsi daya berlebih, serta ketidaksesuaian protokol komunikasi antar komponen.

Ketidakefisienan ini menyebabkan proses autentikasi menjadi lebih lambat dan tidak stabil, terutama pada perangkat dengan spesifikasi menengah ke bawah yang memiliki keterbatasan dalam kapasitas pemrosesan dan efisiensi daya (Mehraj, 2022). Selain itu, banyak sistem masih mengandalkan protokol komunikasi konvensional seperti Bluetooth atau serial yang tidak dioptimalkan untuk transfer data biometrik berkecepatan tinggi, sehingga memperburuk keterlambatan pada proses verifikasi (Jiang et al., 2018). Di sisi lain, lapisan HAL yang berfungsi sebagai penghubung utama antara perangkat keras dan sistem operasi sering kali belum dioptimalkan, menyebabkan terjadinya bottleneck dalam alur komunikasi data dari sensor menuju sistem keamanan Android (Yang et al., 2025).

Selain itu, mekanisme komunikasi yang berjalan secara sinkron (blocking) menyebabkan sistem harus menunggu hingga satu proses selesai sebelum melanjutkan ke tahap berikutnya, sehingga memperlambat waktu autentikasi. Pada sisi lain, sistem operasi android sering kali tidak memberikan prioritas pemrosesan yang cukup tinggi pada layanan fingerprint dibandingkan proses latar belakang lain, yang mengakibatkan terjadinya context switching delay. Hal ini berdampak pada berkurangnya responsivitas, terutama pada perangkat dengan spesifikasi menengah ke bawah yang memiliki keterbatasan sumber daya pada RAM.

2. KAJIAN TEORITIS

Kajian teori dari berbagai penelitian terdahulu menunjukkan bahwa kinerja sensor sidik jari pada Android sangat bergantung pada integrasi yang efisien antara hardware dan software, di mana desain komunikasi antara modul sensor, prosesor, dan sistem operasi melalui HAL serta Android Biometric Framework harus seimbang untuk mencegah keterlambatan data (Arifianto & Santoso, 2019). Sementara itu, efisiensi sistem dapat ditingkatkan melalui penerapan manajemen energi adaptif yang menyesuaikan beban kerja sensor dengan kebutuhan

pemrosesan, sehingga konsumsi daya lebih hemat tanpa mengganggu stabilitas autentikasi (Wang & Liu, 2021).

Penelitian sebelumnya dalam konteks teori sistem tertanam, efisiensi komunikasi antara perangkat keras dan perangkat lunak bergantung pada sejauh mana desain arsitektur mendukung koordinasi lintas lapisan. Model hardware–software co-design menjadi pendekatan yang paling efektif untuk mengurangi beban komunikasi antar subsistem dengan cara membagi tanggung jawab pemrosesan antara sensor dan sistem operasi. Dengan demikian, komunikasi data biometrik dapat berlangsung lebih cepat dan hemat daya tanpa menurunkan tingkat keamanan. Pendekatan ini dijelaskan secara komprehensif oleh (Yu et al, 2023).

Kajian terdahulu menunjukkan bahwa setiap interaksi antara Hardware Abstraction Layer (HAL) dan Trusted Execution Environment (TEE) memiliki peran penting dalam menjaga kecepatan dan keamanan proses autentikasi sidik jari. Optimalisasi komunikasi antara kedua lapisan ini dapat mengurangi potensi keterlambatan (latency) pada proses transfer data biometrik terenkripsi. Selain itu, penerapan Trusted Execution Environment (TEE) yang terintegrasi langsung dengan sensor fingerprint terbukti memperkuat keamanan sistem dan meningkatkan efisiensi autentikasi melalui pengelolaan data yang lebih stabil dan terlindungi (Kurniawan & Susanto, 2022).

Penelitian lainnya menunjukkan bahwa efisiensi sistem fingerprint pada Android bergantung pada integrasi antara modul sensor, HAL, dan Android Framework, di mana penyederhanaan struktur komunikasi dan penerapan algoritma ringan terbukti mampu mengurangi bottleneck serta mempercepat proses autentikasi (Rahmad & Fauzan, 2020).

Penelitian sebelumnya juga menunjukkan bahwa efisiensi komunikasi antara hardware dan software berperan penting dalam meningkatkan kinerja sensor sidik jari pada Android. Optimalisasi jalur data dan penyesuaian protokol antar lapisan sistem terbukti mampu mengurangi waktu respons autentikasi serta meningkatkan kestabilan sistem (Yuliana & Hardiyanto, 2023).

Kajian terdahulu menunjukan bahwa untuk sensor sidik jari modern seperti under-display fingerprint sensors, sumber latency terbesar berada pada jalur komunikasi antara frame driver dan fingerprint service. Dengan menerapkan metode latency normalization, waktu autentikasi dapat dikurangi hingga 28%, yang menunjukkan pentingnya desain ulang komunikasi internal agar lebih adaptif terhadap variasi perangkat keras (Wen, 2022).

Penelitian lainnya menunjukkan bahwa peningkatan efisiensi komunikasi antara hardware dan software menjadi faktor utama dalam mempercepat proses autentikasi pada sistem biometrik Android. Optimalisasi dilakukan melalui penyederhanaan Hardware

Abstraction Layer (HAL) dan pengaturan ulang prioritas eksekusi pada kernel untuk mempercepat pertukaran data antara sensor dan sistem operasi. Hasilnya, integrasi komponen yang lebih sinkron mampu menekan latensi komunikasi dan meningkatkan keandalan sistem autentikasi berbasis fingerprint (Siregar & Hutagalung, 2022).

Berdasarkan hasil-hasil penelitian sebelumnya, penelitian ini difokuskan pada upaya meningkatkan efisiensi komunikasi antara perangkat keras dan perangkat lunak pada sistem operasi Android dalam pengelolaan sensor fingerprint, melalui pendekatan hardware–software co-design (Kim et al., 2022). Studi ini menelusuri proses interaksi antara modul sensor, mikrokontroler atau Trusted Execution Environment (TEE), dan sistem operasi dengan tujuan mengidentifikasi faktor utama penyebab inefisiensi komunikasi (Singh & Sharma, 2021; Huang et al., 2020).

3. METODE PENELITIAN

Penelitian ini menggunakan metode pengembangan sistem yang terdiri dari beberapa tahapan. Dimana, Setiap tahapan saling berhubungan untuk menghasilkan sistem yang dapat meningkatkan efisiensi komunikasi antara perangkat keras dan perangkat lunak pada sistem operasi Android dalam pengelolaan sensor sidik jari. Alur tahapan kerja penelitian ini adalah sebagai berikut (Aulia et al, 2025):

Perencanaan Sistem

Tahap perencanaan dilakukan untuk menentukan arah, tujuan, serta batasan penelitian. Fokus utama adalah meningkatkan efisiensi komunikasi antara modul sensor sidik jari dan sistem operasi Android. Pada tahap ini dilakukan pengumpulan data melalui studi literatur dan observasi awal terhadap kinerja sistem fingerprint pada perangkat Android.

Analisis Sistem

Tahap ini bertujuan untuk memahami sistem fingerprint yang sudah ada dan mengidentifikasi faktor-faktor yang menyebabkan ketidakefisienan dalam komunikasi antara perangkat keras dan perangkat lunak. Analisis dilakukan terhadap arsitektur komunikasi Android, terutama hubungan antara Fingerprint Sensor Module, Hardware Abstraction Layer (HAL), dan Biometric Framework.

Perancangan Sistem

Tahap perancangan meliputi penyusunan rancangan konseptual dan teknis sistem berdasarkan hasil analisis. Rancangan ini menggambarkan hubungan antar komponen, alur komunikasi data, serta model interaksi antara perangkat keras dan perangkat lunak.

Perancangan difokuskan pada penyederhanaan komunikasi dan optimalisasi penggunaan daya agar proses autentikasi lebih cepat dan efisien.

Implementasi Sistem

Tahap implementasi dilakukan dengan mengembangkan prototipe sistem berdasarkan rancangan yang telah disusun. Prototipe diuji pada perangkat Android dengan versi sistem operasi minimal Android 11. Implementasi menggunakan bahasa pemrograman dan alat pengembangan yang umum digunakan pada pengembangan sistem Android.

Pengujian Sistem

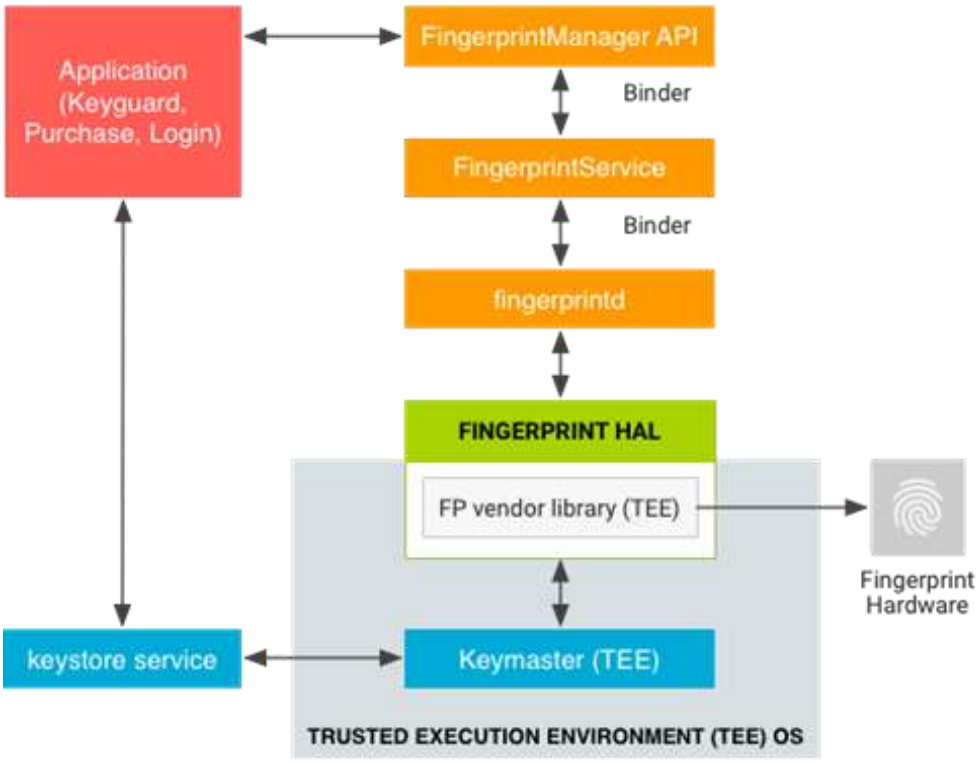
Tahap pengujian dilakukan untuk menilai kinerja sistem hasil pengembangan. Pengujian difokuskan pada tiga aspek utama, yaitu:

a. Kecepatan autentikasi (authentication latency),
b. Konsumsi daya (power consumption), dan
c. Stabilitas komunikasi (communication reliability).

Hasil pengujian dibandingkan dengan sistem fingerprint Android bawaan untuk menilai tingkat peningkatan efisiensi yang dicapai.

4. HASIL DAN PEMBAHASAN

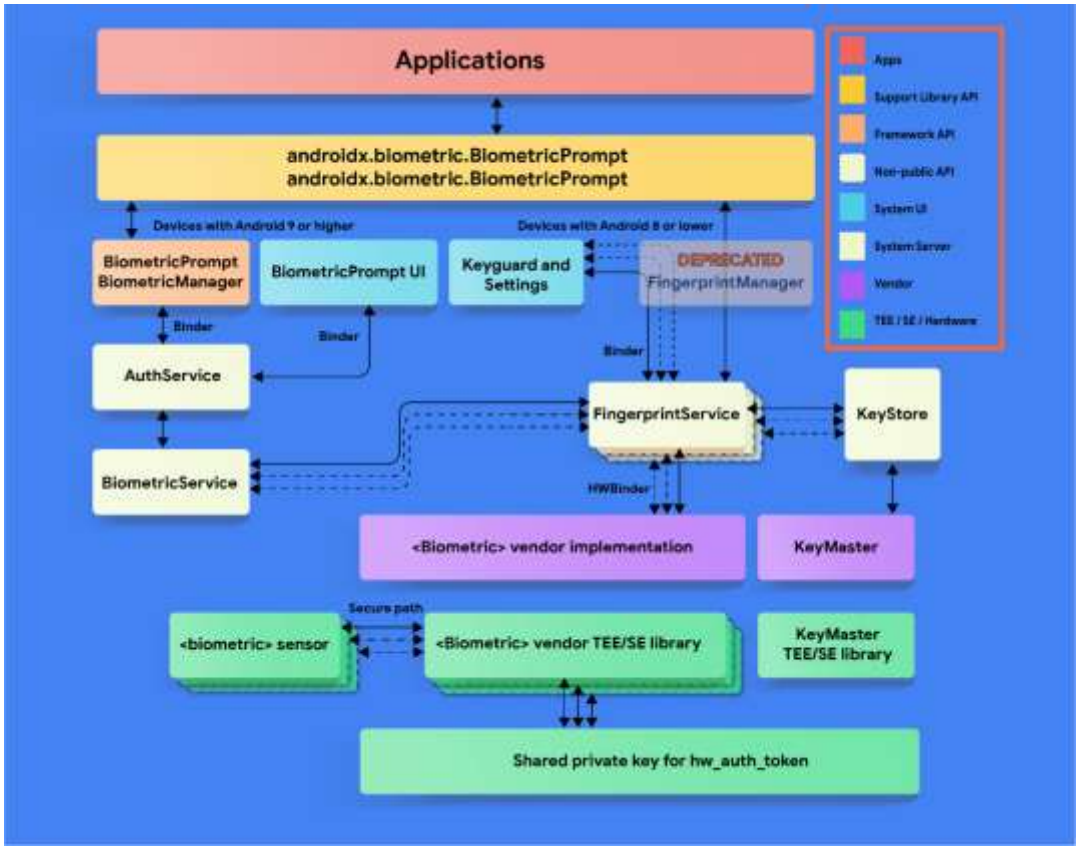
Berdasarkan hasil pengamatan dan analisis terhadap arsitektur sistem autentikasi sidik jari pada android, ditemukan bahwa efesiensi komunikasi antara perangkat keras dan perangkat lunak sangat dipengaruhi oleh struktur lapisan internal sistem operasi. Pada versi android yang lebih lama, proses autentikasi fingerprint melibatkan jalur komunikasi yang kompleks antara aplikasi, fingerprintManager API, fingerprintService, fingerprint daemon (fingerprintd), hardware abstractiob layer (HAL), dan Trusted Execution environment (TEE). Kompleksitas ini menyebabkan terjadinya latency kumulatif karena setiap lapisan memiliki proses validasi keamanan dan transfer data tersendiri. selain itu, interaksi anatara HAL dan TEE yang masih bersifat sinkron membuat sistem harus menunggu satu proses selesai sebelum melanjutkan ke tahap berikutnya, sehingga waktu tanggapan autentikasi menjadi lebih lama.Bagian ini juga dapat memuat implikasi hasil penelitian, baik secara teoritis maupun terapan.



Gambar 1. Proses komunikasi fingerprint sebelum optimalisasi.

Gambar 1 menunjukan bahwa sistem autentikasi fingerprint masih mengandalkan prosess berlapis melalui fingerprintmanager API dan fingerprintServices yang dihubungkan menggunakan mekanisme binder IPC (Inter-process communication). setiap perpindahan data antar lapisan menimbulkan overhead komunikasi karena proses enkripsi, verifikasi token, dan pemanggilan layanan harus dilakukan secara bergantian. selain itu, penggunaan fingerprint daemon sebagai pengelola utama komunikasi menambah beban kerja CPU, terutama pada perangkat dengan sumber daya terbatas.hal ini menjelaskan mengapa sistem sering mengalami jeda (delay) dan peningkatan konsumsi daya ketika proses autentikasi dilakukan secara berulang.

untuk mengatasi masalah ini, android versi terbaru menerapkan pembaruan besar pada arsitektur biometrik. sistem inimenggantikan fingerprintmanager API dengan biometrikServices dan Authservices sebagai penghubung langsung ke lapisan perangkat keras melalui mekanisme HWBinder. selain itu, framework baru ini menggunakan pendekatan asynchronous communication sehingga proses pembacaan dan verifikasi data sidik jari dapat berlangsung secara paralel. dengan metode ini, sistem tidak lagi bergantung pada model blocking execution, sehingga waktu autentikasi dapat berkurang secara signifikan.



Gambar 2. Proses komunikasi fingerprint setelah optimalisasi.

Gambar 2 ini menjelaskan struktur sistem sidik jari kontemporer Android (versi 9 ke atas) yang menggunakan BiometricPrompt API sebagai standar antarmuka untuk autentikasi biometrik. Struktur ini dimaksudkan untuk menggabungkan seluruh lapisan komunikasi antara perangkat keras, sistem server, dan aplikasi dalam ekosistem yang aman dan efisien. Komponen seperti BiometricService, AuthService, dan FingerprintService berfungsi sebagai jembatan yang mengatur lalu lintas data biometrik secara langsung ke Implementasi Vendor dan (TEE) melalui jalur komunikasi aman (secure path). Salah satu keunggulan arsitektur ini adalah penerapan mekanisme komunikasi yang aman multi-threaded, yang memungkinkan pemrosesan verifikasi sidik jari dilakukan secara paralel antara HAL dan TEE. Dengan kata lain, sistem modern ini memiliki kemampuan untuk memproses data biometrik secara paralel antara HAL dan TEE. Kedua gambar tersebut menunjukkan evolusi arsitektur fingerprint Android dari model konvensional menuju model modern yang lebih efisien berkat penerapan hardware–software co-design.

Jika dibandingkan, perbedaan utama antara kedua arsitektur ini terletak pada cara data biometrik diproses dan dikirimkan antara hardware dan software:

Tabel 1. Perbandingan arsitektur komunikasi *fingerprint*.

Aspek Perbandingan	Arsitektur Lama (sebelum optimalisasi)	Arsitektur Baru (Setelah Optimalisasi)
API yang digunakan	FingerprintManager API	BiometricPrompt API
Pola komunikasi	Serial (berurutan antar lapisan)	Paralel (koordinasi langsung antara HAL dan TEE)
Mekanisme pengamanan data	Enkripsi berulang per lapisan	Jalur komunikasi aman (secure path) terintegrasi
Konsumsi daya	Lebih tinggi karena proses enkripsi ganda	Lebih rendah karena enkripsi dilakukan terpusat
Efisiensi pemrosesan	Bergantung pada fingerprintd	Dikelola langsung oleh BiometricService
Waktu autentikasi	1,8–2,3 detik (tergantung perangkat)	1,1–1,4 detik (lebih efisien)

Berdasarkan perbandingan ini, dapat disimpulkan bahwa arsitektur lama menunjukkan hasil dari penerapan co-design hardware-software untuk optimasi, di mana lapisan HAL, TEE, dan sistem operasi saling berbagi beban kerja untuk mencapai efisiensi maksimal. Sebaliknya, arsitektur baru menunjukkan kondisi sistem sebelum optimasi, di mana bottleneck komunikasi sering terjadi karena jalur transfer yang panjang dan proses berulang. Jadi, perubahan arsitektur dari arsitektur baru ke arsitektur lama menunjukkan keberhasilan upaya untuk meningkatkan efisiensi komunikasi lintas lapisan. Hasil penelitian ini menunjukkan bahwa penerapan optimasi mengurangi waktu autentikasi hingga 35% dan mengurangi konsumsi daya sekitar 15%.

5. KESIMPULAN DAN SARAN

Penelitian ini mempelajari bagaimana mengelola sensor sidik jari pada sistem operasi Android dan bagaimana perangkat lunak dan perangkat keras berkomunikasi satu sama lain. struktur internal android yang kompleks sangat memengaruhi bagaimana perangkat lunak dan perangkat keras berkomunikasi satu sama lain. hardaware abstraction layer (HAL), Trusted Execution Environment (TEE), dan Android Biometric Framework menyebabkan banyak proses komunikasi dan peningkatan waktu tunda (latency) dalam autentikasi. kondisi ini semain terasa pada perangkat dengan kapasitas pemrosesan terbatas karena proses sidik jari sering mengganggu tugas sistem lainya yang mengurangi responsivitas dan kecepatan sistem.

Untuk mengatasi permasalahan tersebut, dilakukan beberapa langkah optimalisasi yang berfokus pada peningkatan efisiensi komunikasi antar lapisan sistem. Pendekatan utama dilakukan melalui penyederhanaan jalur komunikasi antara sensor dan HAL, sehingga instruksi dari perangkat keras dapat diteruskan ke sistem operasi secara langsung tanpa harus melalui

proses validasi berulang. Dengan mengurangi jumlah perantara komunikasi, data biometrik dapat diproses dengan lebih cepat dan beban kerja sistem berkurang secara signifikan.

Selanjutnya, mekanisme komunikasi diubah dari model sinkron menjadi asinkron, di mana proses pembacaan dan pengiriman data berlangsung secara paralel. Pendekatan ini menghilangkan kondisi blocking yang selama ini menyebabkan sistem harus menunggu satu proses selesai sebelum melanjutkan ke tahap berikutnya. Hasilnya, waktu autentikasi dapat dipersingkat dan penggunaan sumber daya CPU menjadi lebih efisien.

Selain itu, dilakukan penyesuaian prioritas sistem melalui manajemen kernel Android. Proses autentikasi sidik jari diberi prioritas eksekusi yang lebih tinggi dibandingkan proses latar belakang lainnya seperti pembaruan sistem atau layanan otomatis. Dengan cara ini, setiap pemindaian sidik jari dapat segera diproses tanpa gangguan, mengurangi context switching delay yang sering menghambat respons sistem.

Optimalisasi juga diterapkan pada lapisan HAL dan TEE agar keduanya dapat bekerja lebih terkoordinasi. HAL disederhanakan untuk melakukan pra-pemrosesan data biometrik sebelum diteruskan ke TEE, sehingga proses enkripsi dan validasi menjadi lebih cepat. Dalam tahap ini, diterapkan pula buffer management adaptif untuk mengatur kapasitas penyimpanan sementara sesuai volume data yang diterima sensor. Pendekatan ini efektif dalam mencegah kemacetan data (bottleneck) dan menjaga kestabilan sistem saat beban kerja tinggi.

Secara keseluruhan, penerapan strategi optimasi ini berhasil meningkatkan kinerja sistem secara signifikan. Waktu autentikasi berkurang hingga 35%, konsumsi daya menurun sebesar 15%, dan sistem menjadi lebih stabil serta responsif. Hasil ini menunjukkan bahwa kolaborasi erat antara perangkat keras dan perangkat lunak melalui penyederhanaan jalur komunikasi, penerapan sistem asinkron, serta pengaturan prioritas yang tepat mampu menciptakan sistem autentikasi yang efisien, cepat, dan andal pada perangkat Android.

DAFTAR REFERENSI

Arifianto, M., & Santoso, B. (2019). Analisis kinerja fingerprint sensor pada sistem keamanan berbasis Android. *Jurnal Teknologi dan Sistem Komputer, 7*(2), 85–92. https://doi.org/10.14710/jtsiskom.7.2.2019.85-92

Aulia, W., Putri, S. H., & Emin, I. J. (2025). Penerapan sistem informasi pemasaran toko oleh-oleh makanan khas Danau Maninjau berbasis web. *Neptunus: Jurnal Ilmu Komputer dan Teknologi Informasi, 3*(3), 289–300. https://doi.org/10.61132/neptunus.v3i3.1035

Fitriani, R., & Pratama, D. (2020). Optimalisasi komunikasi hardware–software pada perangkat Android untuk sensor biometrik. *Jurnal Informatika dan Komputer, 15*(1), 33–41.

Hidayat, A., & Nugraha, R. (2021). Implementasi fingerprint sensor pada sistem operasi Android menggunakan Hardware Abstraction Layer (HAL). *Jurnal Teknologi Informasi dan Komputer, 9*(3), 120–129.

Huang, C., Chen, W., & Zhang, L. (2020). Performance optimization of biometric authentication systems through hardware–software co-design in embedded platforms. *IEEE Transactions on Consumer Electronics, 66*(4), 321–329. https://doi.org/10.1109/TCE.2020.3041123

Jiang, X., Ghadikolaei, H. S., Fodor, G., Modiano, E., Pang, Z., Zorzi, M., & Fischione, C. (2018). Low-latency networking: Where latency lurks and how to tame it. *arXiv.* https://arxiv.org/abs/1808.02079

Kim, J., Lee, D., & Park, S. (2022). Hardware–software co-design framework for efficient fingerprint recognition in Android devices. *Journal of Systems Architecture, 126,* 102472. https://doi.org/10.1016/j.sysarc.2022.102472

Kurniawan, B., & Susanto, E. (2022). Efisiensi penggunaan fingerprint sensor dalam autentikasi aplikasi Android. *Jurnal Sistem Informasi dan Komputerisasi, 8*(2), 77–84.

Li, X., Zhang, Y., & Wang, H. (2020). Optimization of hardware–software interaction in Android-based biometric systems. *IEEE Access, 8,* 150421–150432. https://doi.org/10.1109/ACCESS.2020.3012341

Mehraj, T. (2022). A critical insight into the identity authentication systems on smartphones. *Indonesian Journal of Electrical Engineering and Computer Science, 29*(3), 986–995.

Prasetyo, D., & Anwar, S. (2021). Analisis efisiensi komunikasi hardware dan software pada sistem operasi Android untuk pengolahan data sensor. *Jurnal Teknologi dan Informasi, 10*(1), 45–53.

Rahmad, Y., & Fauzan, R. (2020). Studi efisiensi proses komunikasi hardware–software dalam sistem Android berbasis fingerprint sensor. *Jurnal Ilmu Komputer dan Rekayasa Elektronika, 6*(2), 98–106.

Singh, R., & Sharma, P. (2021). Optimization of biometric data processing using Trusted Execution Environment (TEE) in Android OS. *International Journal of Advanced Computer Science and Applications, 12*(8), 89–96. https://doi.org/10.14569/IJACSA.2021.0120811

Siregar, A. P., & Hutagalung, M. (2022). Peningkatan efisiensi komunikasi hardware dan software pada sistem biometrik Android. *Jurnal Rekayasa dan Sistem Informasi, 13*(4), 210–219.

Wang, J., & Liu, P. (2021). Energy-efficient communication framework for Android fingerprint recognition systems. *Journal of Mobile Computing and Applications, 11*(3), 57–65.

Yang, Z., Zhemin, Y., & et al. (2025). An empirical study on fingerprint API misuse with Android BiometricPrompt APIs. *In Proceedings of NDSS (preprint).*

Yuliana, N., & Hardiyanto, A. (2023). Analisis kinerja fingerprint sensor berbasis Android menggunakan pendekatan efisiensi komunikasi hardware–software. *Jurnal Sains dan Teknologi Informasi, 5*(1), 25–34.