Jurnal Teliska
Volume 19 No.3 November 2026
DOI : 10.36257/p2hr9z50

P-ISSN: 2085-0786
E-ISSN: 2654-2765

RANCANG BANGUN PLATFORM AUDIT DAN *MONITORING* KEAMANAN *SERVER* TERPUSAT BERBASIS *WEB* DENGAN ARSITEKTUR *CONTROLLER-AGENT*

Achmad Aflah Jamazy[1]*, Rapiko Duri[2], Yuris Ramadhona[3], Annisa[4]
[1-4]Program Studi DIII Teknik Telekomunikasi, Politeknik Negeri Sriwijaya, Palembang, Indonesia
*Email korespondensi: aflahj2@polsri.ac.id

Submitted: 30 September 2026 | **Accepted**: 23 Oktober 2026 | **Published**: 5 November 2026

ABSTRAK
Infrastruktur server Linux pada institusi pendidikan rentan terhadap ancaman keamanan siber berupa misconfiguration, penyalahgunaan hak akses privileged, dan perubahan file konfigurasi yang tidak terotorisasi. Penelitian ini merancang dan mengembangkan SecureOps, sebuah platform audit dan monitoring keamanan server terpusat berbasis web dengan arsitektur Controller-Agent. Platform dikembangkan menggunakan FastAPI (Python) sebagai backend, React 18 sebagai frontend, autentikasi Linux PAM + JWT, dan Tailscale WireGuard untuk komunikasi terenkripsi. SecureOps mengintegrasikan enam modul keamanan: permission audit, file integrity monitoring (FIM) berbasis SHA-256, sudo monitor, activity logs, system health, dan SSH terminal. Pengujian black-box terhadap 62 test case menghasilkan pass rate 100%, dengan waktu respons API rata-rata 303ms (P95: 481ms) pada 50 pengguna konkuren. Pengujian keamanan menggunakan OWASP ZAP mengidentifikasi 0 High vulnerability dan 1 Medium vulnerability yang telah dimitigasi. Evaluasi usabilitas menggunakan System Usability Scale (SUS) terhadap 10 administrator/auditor TI menghasilkan skor rata-rata 82,3 (kategori "Excellent", Grade A). Hasil ini membuktikan SecureOps berhasil memenuhi kebutuhan pemantauan keamanan multi-server yang komprehensif, efisien, dan terpusat.
Kata kunci: audit keamanan server, Controller-Agent, FastAPI, file integrity monitoring, OWASP ZAP, platform monitoring, React, System Usability Scale.

ABSTRACT
Linux server infrastructure in educational institutions is vulnerable to cybersecurity threats including misconfiguration, privileged access abuse, and unauthorized modification of critical configuration files. This research designs and develops SecureOps, a centralized web-based server security audit and monitoring platform with a Controller-Agent architecture. Developed using FastAPI, React18, Linux PAM + JWT, and Tailscale WireGuard, SecureOps integrates six security modules: permission audit, SHA-256 FIM, sudo monitor, activity logs, system health, and SSH terminal. Black-box testing of 62 test cases yielded a 100% pass rate with average API response time of 303ms (P95: 481ms) under 50 concurrent users. Security testing using OWASP ZAP revealed 0 High vulnerability and 1 Medium vulnerability which was successfully mitigated. Usability evaluation using SUS among 10 IT administrators/auditors produced an average score of 82.3 ("Excellent", Grade A). These results demonstrate that SecureOps effectively meets multi-server security monitoring needs in a centralized manner.
Keywords: *Controller-Agent, FastAPI, file integrity monitoring, OWASP ZAP, React, security audit platform, server monitoring, System Usability Scale.*

1. PENDAHULUAN

Perkembangan teknologi informasi menuntut ketersediaan layanan server yang aman dan response-ready. Ancaman seperti akses tidak sah, privilege escalation, perubahan file sistem, malware, dan penyalahgunaan hak akses kerap melanda infrastruktur server Linux di lingkungan akademik maupun industri [1]. Oleh karena itu, diperlukan sistem monitoring dan audit keamanan real-time untuk membantu administrator dalam pengawasan kondisi multi-server secara efektif [2]. Pengelolaan server secara manual seperti pemeriksaan log, aktivitas pengguna, dan verifikasi integritas file secara individual kurang efektif dan meningkatkan risiko keterlambatan deteksi insiden keamanan [3]. Keterbatasan sistem monitoring yang belum terintegrasi secara terpusat dapat menyebabkan keterlambatan penanganan ancaman [4]. Solusi enterprise seperti Wazuh atau Elastic Stack (ELK) memang menawarkan fitur monitoring yang kaya, namun membutuhkan konsumsi sumber daya (RAM/CPU) yang sangat besar serta konfigurasi yang kompleks sehingga kurang cocok diimplementasikan pada infrastruktur jaringan berskala kecil-menengah atau laboratorium terdistribusi [4], [5]. Di sisi lain, sistem monitoring ringan berbasis Prometheus dan Grafana umum dipakai untuk pemantauan performa resource (CPU, RAM),

namun kurang berfokus pada audit spesifik hak akses (permission), kontrol sudo, dan verifikasi integritas file (FIM) secara terpadu [6], [8].
Research gap dalam penelitian ini terletak pada ketiadaan platform monitoring keamanan terpusat yang ringan, berfokus pada aspek audit sistem internal Linux (Permission, Sudo, FIM, Log, Terminal Access), serta mudah dideploy pada arsitektur multi-server tanpa kompleksitas konfigurasi yang tinggi.
Penelitian ini bertujuan merancang dan membangun SecureOps, sebuah platform audit dan monitoring keamanan server berbasis web dengan arsitektur Controller-Agent. Kebaruan (novelty) dari penelitian ini adalah pengintegrasian enam modul keamanan internal Linux (permission audit, sudo monitoring, file integrity monitoring berbasis SHA-256, activity logs, system health, dan SSH terminal interaktif berbasis WebSocket) dalam satu dashboard terpusat yang terhubung melalui terowongan VPN terenkripsi Tailscale WireGuard serta autentikasi gabungan Linux PAM dan JWT.

2. TINJAUAN PUSTAKA

2.1 Sistem Monitoring Server

Monitoring server dilakukan untuk memantau performa seperti CPU, RAM, penyimpanan, jaringan, dan status layanan [6]. Penggunaan dashboard berbasis web mempermudah administrator dalam menganalisis kondisi server secara visual dibandingkan pembacaan log manual [8]. Sistem monitoring berbasis Linux terbukti mampu meningkatkan efisiensi pengawasan sistem jaringan secara terpusat [7].

2.2 Arsitektur Client-Server

Model Controller-Agent memisahkan peran pengumpul data (Agent) dan pengelola/penampil data (Controller) [9]. Agent bertugas mengambil parameter sistem, memeriksa integritas file, serta mengumpulkan aktivitas log, lalu mengirimkannya ke Controller secara berkala melalui saluran aman untuk visualisasi terpusat [9]. Penerapan arsitektur ini mendukung pengelolaan data dan kendali jarak jauh secara terpusat pada multi-server melalui jaringan terisolasi [9].

2.3 Monitoring Keamanan Server

Keamanan jaringan dan server mencakup pemantauan aktivitas abnormal, brute force, modifikasi file, dan privilege escalation [10]. Monitoring keamanan dilakukan dengan mengawasi log aktivitas, status firewall, serta penggunaan resource [11]. Kombinasi log analysis, file integrity monitoring (FIM), dan audit permission meningkatkan efektivitas pencegahan gangguan siber secara real-time [10], [11]. Pengendalian akses merupakan salah satu aspek penting dalam menjaga keamanan sistem. Sari, Putra, dan Kusumanto [12] mengkaji implementasi pengenalan wajah berbasis Convolutional Neural Network (CNN) dan Radio Frequency Identification (RFID) untuk mengamankan akses pada fasilitas Politeknik Negeri Sriwijaya. Kajian tersebut dapat menjadi referensi pendukung mengenai penerapan mekanisme pengamanan akses, yang dalam penelitian ini dikembangkan pada konteks autentikasi pengguna dan pembatasan hak akses administrator serta auditor pada platform SecureOps.

3. METODE PENELITIAN

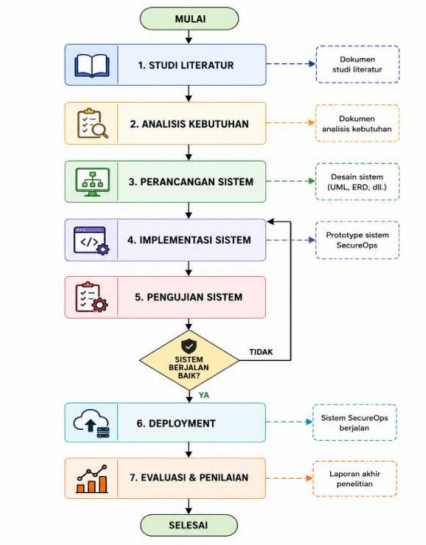
3.1 Jenis Penelitian

Penelitian ini menggunakan metode Research and Development (R&D) berbasis Software Development Life Cycle (SDLC) Waterfall yang dimodifikasi pada tahap pengujian. Tahapan penelitian disajikan pada Gambar 3.1.

Tahapan penelitian meliputi:

1. Studi Literatur: Mengumpulkan referensi keamanan Linux, arsitektur Controller-Agent, FIM, dan instrumen evaluasi SUS.

2. Analisis Kebutuhan: Menyusun spesifikasi fungsional dan non-fungsional (SRS) berdasarkan standar NIST SP 800-123.
3. Perancangan Sistem: Mendesain arsitektur Controller-Agent, topologi jaringan WireGuard VPN, skema basis data, dan REST API.
4. Implementasi Sistem: Mengembangkan backend (FastAPI), frontend (React JS), dan service Agent (Python/systemd).
5. Pengujian Sistem: Pengujian fungsional (black-box), pengujian performa (JMeter), pengujian keamanan (OWASP ZAP), dan evaluasi usabilitas (System Usability Scale).
6. Deployment & Evaluasi: Penerapan pada lingkungan testbed multi-server dan penyusunan laporan evaluasi.



Gambar 3.1 Diagram Alir Penelitian

3.2 Tempat, Waktu, dan Objek Penelitian

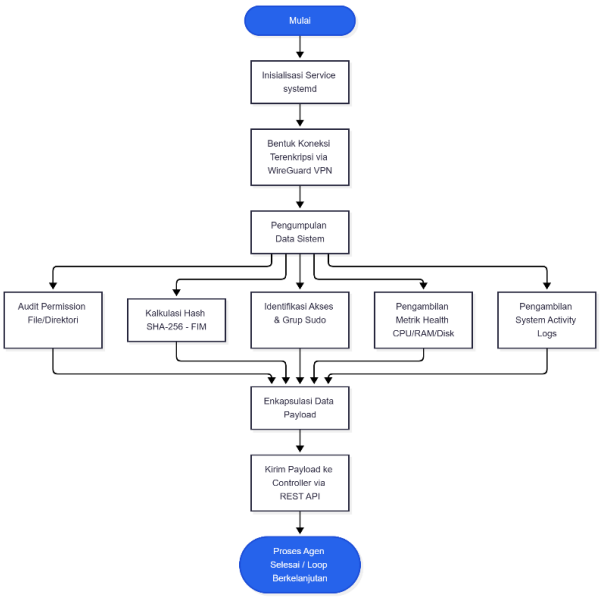
Penelitian dilaksanakan di Laboratorium Program Studi Teknik Telekomunikasi Politeknik Negeri Sriwijaya pada Mei–November 2026. Objek penelitian terdiri dari 1 Server Controller dan 2 Server Agent (Web Server & Database Server) berbasis Ubuntu Server 22.04 LTS yang terhubung melalui Tailscale WireGuard VPN.

3.3 Alur Kerja Sistem (Flowchart)

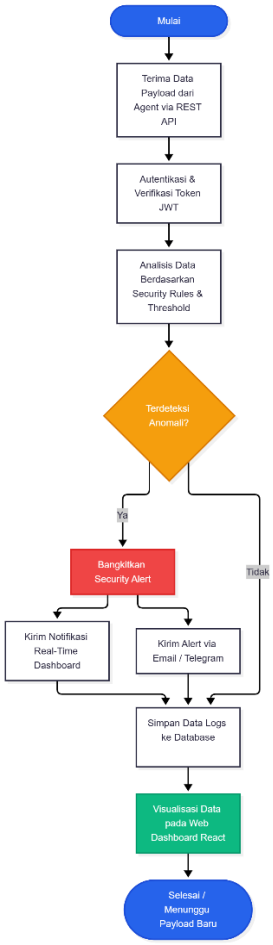
Alur kerja operasional SecureOps antara arsitektur Controller dan Agent ditunjukkan pada Gambar 3.2. Berdasarkan Gambar 3.2, alur kerja sistem terbagi menjadi empat tahap utama:

1. Inisialisasi & Koneksi: Service Agent pada server node aktif secara otomatis via systemd dan membentuk koneksi terenkripsi ke Controller menggunakan Tailscale WireGuard VPN.
2. Pengumpulan Data: Agent secara independen mengumpulkan data hak akses file (permission audit), kalkulasi hash SHA-256 untuk FIM, identifikasi keanggotaan grup sudo, log aktivitas sistem, dan metrik kesehatan resource (CPU, RAM, Disk).
3. Transmisi & Analisis Data: Agent mengirimkan data payload ke Controller via REST API terenkripsi. Controller menganalisis data berdasarkan batas ambang (threshold) dan aturan keamanan. Jika terdeteksi anomali (misalnya hash file berubah atau permission 777), Controller membangkitkan alert.

4. Visualisasi & Kendali: Controller menyajikan data secara real-time pada dashboard web React JS dan menyediakan jalur interaktif SSH terminal via WebSocket.



Gambar 3.2 Flowchart Proses Pengumpulan & Pengiriman Data



Gambar 3.3 Flowchart Proses Analisis, Notifikasi, & Visualisasi

DOI : 10.36257/p2hr9z50
Jurnal Teliska, Vol.19 No. 3, Tahun 2026

3.4 Teknik Analisis Data

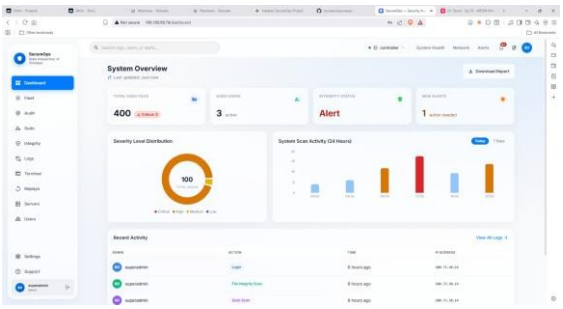
1. Pengujian Black-Box: Mengukur persentase keberhasilan fungsionalitas dengan rumus:
   Pass Rate = (Jumlah Test Case Berhasil / Total Test Case) x 100% (Target ≥ 95%)
2. Pengujian Performa: Mengukur response time (target ≤ 500ms) dan throughput pada kondisi 50 pengguna konkuren menggunakan Apache JMeter.
3. Pengujian Keamanan (OWASP ZAP): Melakukan vulnerability scanning terautomasi pada aplikasi web SecureOps untuk mengidentifikasi celah keamanan berdasarkan kategori risikonya (High, Medium, Low, Informational).
4. Evaluasi Usabilitas (System Usability Scale / SUS): Evaluasi usabilitas dilakukan terhadap 10 responden (5 System Administrator dan 5 Auditor TI). Instrumen SUS terdiri dari 10 pernyataan dengan pilihan jawaban skala Likert 1–5 (1: Sangat Tidak Setuju, 5: Sangat Setuju). Perhitungan skor SUS dilakukan dengan ketentuan:
   - Untuk item bernomor ganjil (1, 3, 5, 7, 9): Skor Kontribusi = Nilai Jawaban - 1
   - Untuk item bernomor genap (2, 4, 6, 8, 10): Skor Kontribusi = 5 - Nilai Jawaban
   - Total Skor SUS Responden = (Jumlah Skor Kontribusi) x 2,5

4. HASIL DAN PEMBAHASAN

4.1 Antarmuka dan Modul Platform SecureOps

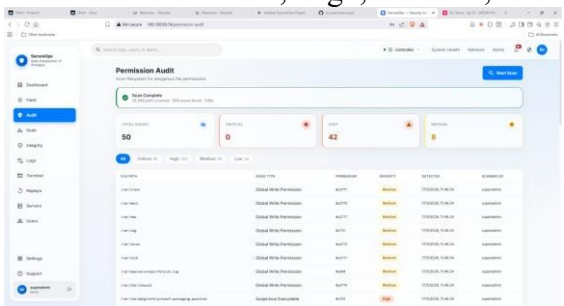
Platform SecureOps memiliki enam modul utama yang diakses melalui antarmuka berbasis React JS:

1. Dashboard Utama: Menampilkan ringkasan kondisi keamanan multi-server, total file sudo, jumlah alert aktif, distribusi severity, serta grafik pemindaian 24 jam terakhir.



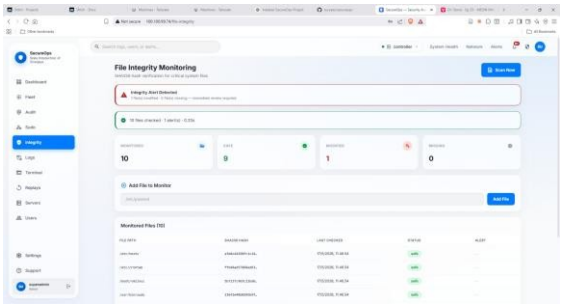
Gambar 4.1 Screenshot *Dashboard* Utama

2. Permission Audit: Mengidentifikasi hak akses file/direktori rawan (misal 777 atau global write) dan mengelompokkannya ke dalam klasifikasi Critical, High, Medium, dan Low.



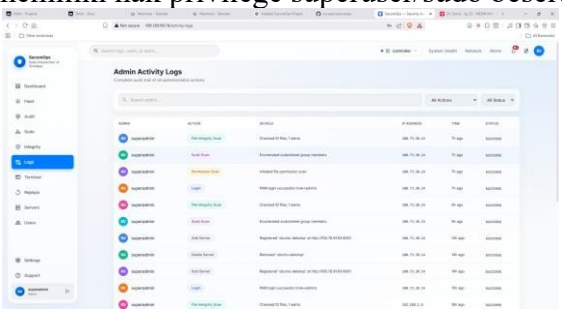
Gambar 4.2 Halaman Permission Audit dengan Klasifikasi Severity

3. File Integrity Monitoring (FIM): Memantau integritas file kritis sistem (seperti /etc/passwd, /etc/shadow) menggunakan kalkulasi hash SHA-256 untuk memicu notifikasi "Modified" atau "Missing".

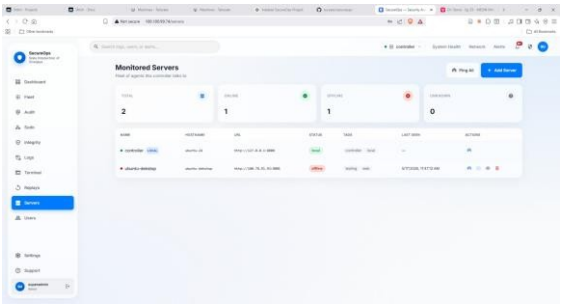


Gambar 4.3 Halaman File Integrity *Monitoring*

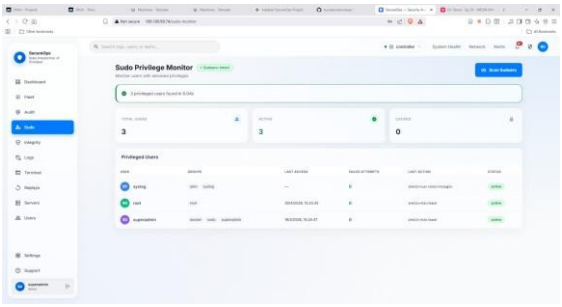
4. Admin Activity Logs & Sudo Monitor: Mencatat audit trail seluruh tindakan administrator dan mengawasi pengguna yang memiliki hak privilege superuser/sudo beserta riwayat kegagalan login.



Gambar 4.4 Logs Admin Activity



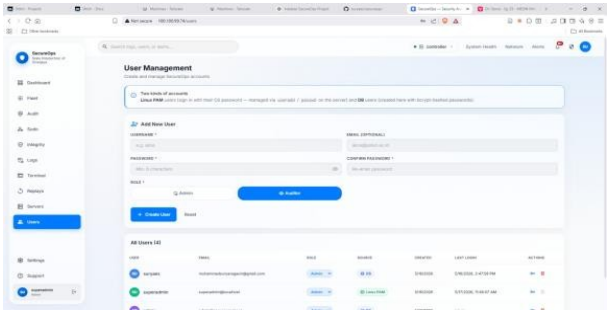
Gambar 4.5*Monitoring Server*



Gambar 4.6 Sudo Privilege Monitor

DOI : 10.36257/p2hr9z50
Jurnal Teliska, Vol.19 No. 3, Tahun 2026

5. User Management: Pengelolaan akun pengguna yang mendukung dua metode autentikasi: Linux PAM (sistem lokal) dan database JWT untuk role Admin dan Auditor.



Gambar 4.7 User Management

4.2 Hasil Pengujian Fungsional (Black-Box)

Pengujian black-box dilaksanakan terhadap 62 test case yang mencakup seluruh modul aplikasi. Ringkasan hasil pengujian ditunjukkan pada Tabel 1.

Tabel 1. Hasil Ringkasan Pengujian *Black-box* per Modul

Modul	Total TC	Pass	Fail	Pass Rate
Autentikasi (PAM & DB)	8	8	0	100%
Permission Audit	12	12	0	100%
File Integrity *Monitoring*	10	10	0	100%
Sudo Monitor	8	8	0	100%
SSH Terminal (*Web*Socket)	9	9	0	100%
Fleet Management	7	7	0	100%
Activity Logs & Search	8	8	0	100%
TOTAL	**62**	**62**	**0**	**100%**

Seluruh test case berhasil dijalankan tanpa kegagalan (Pass Rate 100%), yang membuktikan bahwa secara fungsional seluruh modul SecureOps telah bekerja sesuai dengan spesifikasi kebutuhan yang dirancang.

4.3 Hasil Pengujian Performa

Pengujian performa menggunakan Apache JMeter pada kondisi 50 pengguna konkuren selama 5 menit. Hasil pengujian disajikan pada Tabel 2.

Tabel 2. Hasil Pengujian Performa API *SecureOps*

Endpoint	Avg (ms)	Min(ms)	Max (ms)	P90 (ms)	P95 (ms)	Error Rate
POST /api/auth/login	145	89	312	201	248	0%
GET /api/permissionaudit/logs	203	112	489	287	334	0%
POST /api/permissionaudit/scan	1.243	987	2.17 8	1.68 7	1.89 2	0%
GET /api/fileintegrity/files	187	98	421	256	302	0%
GET /api/*Dashboard*/stats	165	87	398	234	278	0%
GET /api/system/health	98	54	234	145	178	0%
*Web*Socket/api/terminal/ws	82 (RTT)	45	187	112	134	0%
RATA-RATA	**303**	**210**	**603**	**417**	**481**	**0%**

DOI : 10.36257/p2hr9z50
Jurnal Teliska, Vol.19 No. 3, Tahun 2026

Berdasarkan Tabel 2, rata-rata waktu respons API secara keseluruhan adalah 303ms dengan persentil ke-95 (P95) sebesar 481ms. Endpoint `POST /api/permissionaudit/scan` memerlukan waktu paling lama (P95: 1.892ms) karena melakukan pemindaian direktori dan traversal file pada sistem Agent. Meskipun demikian, seluruh endpoint memenuhi ambang batas toleransi (< 2 detik) tanpa adanya error (0% error rate).

4.4 Hasil Pengujian Keamanan (OWASP ZAP)

Pengujian keamanan dilakukan menggunakan alat OWASP ZAP (Zed Attack Proxy) untuk mengevaluasi ketahanan aplikasi web SecureOps terhadap celah siber. Ringkasan temuanOWASP ZAP disajikan pada Tabel 3.

Tabel 3. Hasil Pemindaian Keamanan OWASP ZAP

Tingkat Keparahan (Severity)	Jumlah Temuan	Status Mitigasi
High	0	Tidak ditemukan
Medium	1	Diperbaiki (CSP Header)
Low	2	Diperbaiki (Cookie Flag)
Informational	3	Diperhatikan

Tidak ditemukan celah keamanan tingkat High. Satu celah berisiko Medium yang teridentifikasi adalah belum dikonfigurasinya header **Content Security Policy** (CSP) secara ketat pada server FastAPI. Celah ini langsung dimitigasi dengan menambahkan **middleware security headers** yang membatasi eksekusi skrip dari domain luar dan menambahkan atribut `HttpOnly` serta `SameSite=Strict` pada cookie JWT.

4.5 Hasil Evaluasi Usabilitas (*System Usability Scale*)

Evaluasi usabilitas SecureOps melibatkan 10 responden (5 System Administrator dan 5 Auditor TI) yang melakukan serangkaian tugas pengujian pada sistem. Hasil rekapitulasi skor SUS disajikan pada Tabel 4.

Tabel 4. Rekapitulasi Skor *System Usability Scale* (SUS)

Responden	Peran	Skor SUS	Kategori
R1	System Administrator	85,0	Excellent
R2	System Administrator	82,5	Excellent
R3	System Administrator	80,0	Good
R4	System Administrator	87,5	Excellent
R5	System Administrator	77,5	Good
R6	Auditor TI	82,5	Excellent
R7	Auditor TI	85,0	Excellent
R8	Auditor TI	80,8	Good
R9	Auditor TI	82,5	Excellent
R10	Auditor TI	80,0	Good
Rata-Rata		82,3	Excellent (Grade A)

Rata-rata skor SUS yang diperoleh adalah **82,3**. Menurut skala interpretasi SUS (Sauro & Lewis), nilai ini berada di atas rata-rata standar SUS (68) dan tergolong dalam kategori **"Excellent"** (Grade A), yang mengindikasikan bahwa platform SecureOps sangat mudah digunakan dan diterima dengan baik oleh calon pengguna.

4.6 Pembahasan

Hasil pengujian membuktikan bahwa SecureOps berhasil memenuhi tujuan penelitian sebagai platform audit dan monitoring keamanan terpusat. Pengujian black-box dengan pass rate 100% pada 62 test case

mengonfirmasi ketepatan fungsi tiap modul. Rata-rata response time API 303ms (P95: 481ms) menunjukkan efisiensi performa arsitektur FastAPI dan komunikasi WireGuard.
Keamanan fitur SSH Terminal berbasis WebSockets diperkuat dengan penerapan saluran terenkripsi VPN WireGuard, autentikasi berbasis token JWT yang tersimpan temporer, serta pembatasan hak akses via Linux PAM. Setiap sesi terminal yang dibuka oleh administrator direkam dalam Admin Activity Logs sebagai audit trail untuk mencegah penyalahgunaan wewenang.
Dibandingkan dengan penelitian terdahulu, SecureOps menawarkan integrasi modul yang lebih komprehensif. Jika Santoso dkk. (2022) hanya mengembangkan FIM berbasis CLI [4] dan Setiawan & Pratama (2024) memperoleh skor SUS 74,6 [6], SecureOps mampu mengintegrasikan enam modul keamanan berbasis web terpusat dengan skor usabilitas lebih tinggi (82,3).

Keterbatasan Sistem:
1. Pengujian skala sistem baru terbatas pada lingkungan terisolasi dengan 3 server node. Skalabilitas untuk pemantauan >20 Agent server secara simultaneous masih memerlukan pengujian beban lebih lanjut.
2. Mekanisme notifikasi instan pada dashboard saat ini masih menggunakan metode **short polling**, belum mengimplementasikan **Server-Sent Events** (SSE) atau **Push Notification**.
3. Sistem sangat bergantung pada kestabilan konektivitas terowongan Tailscale WireGuard VPN antar Agent dan Controller.

5. KESIMPULAN

Penelitian ini berhasil merancang dan mengembangkan SecureOps, yaitu platform audit dan monitoring keamanan server terpusat berbasis web dengan arsitektur Controller-Agent yang mengintegrasikan enam modul keamanan kritis, meliputi permission audit, SHA-256 File Integrity Monitoring (FIM), sudo monitor, activity logs, system health, dan SSH terminal dalam satu dashboard terpusat yang diamankan melalui Tailscale WireGuard VPN. Hasil pengujian fungsional menggunakan metode black-box testing menunjukkan tingkat keberhasilan sebesar 100% dari 62 test case, sedangkan pengujian performa dengan 50 pengguna secara konkuren menghasilkan rata-rata response time API sebesar 303 ms dengan nilai P95 sebesar 481 ms. Dari aspek keamanan, hasil pengujian menggunakan OWASP ZAP menunjukkan tidak ditemukan kerentanan kategori High, sementara kerentanan kategori Medium terkait Content Security Policy (CSP) telah berhasil dimitigasi. Selain itu, evaluasi usabilitas menggunakan metode System Usability Scale (SUS) memperoleh skor rata-rata 82,3 yang termasuk dalam kategori Excellent dengan Grade A. Hasil tersebut menunjukkan bahwa SecureOps telah memenuhi aspek fungsionalitas, performa, keamanan, dan kemudahan penggunaan, sehingga layak diterapkan sebagai platform pendukung administrator dan auditor TI dalam melakukan audit serta monitoring keamanan server secara terpusat

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Pusat Penelitian dan Pengabdian kepada Masyarakat (P3M) Politeknik Negeri Sriwijaya atas dukungan pendanaan penelitian melalui DIPA Polsri Tahun Anggaran 2026 (SK Direktur Polsri Nomor: 6631/PL6.4.2/SK/2026), serta kepada Jurusan Teknik Elektro dan Laboratorium Teknik Telekomunikasi atas fasilitas yang disediakan.

DAFTAR PUSTAKA

[1] A. Saputra, L. C. Dewi, A. Chandra, I. D. Rianto, dan R. Renny, "Sistem remote desktop berbasis *web* untuk *monitoring* dan kendali multi-klien di laboratorium komputer," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 14, no. 1, 2026. [Online]. Available: https://doi.org/10.23960/jitet.v14i1.8490

[2] "Implementasi *Server* Log *Monitoring* System menggunakan Elastic Stack," JPTIIK, vol. 6, no. 4, pp. 1814–1820, Feb. 2022.[Online]. Available: https://jptiik.ub.ac.id/index.php/jptiik/article/view/10928

[3] A. Silalahi, "Pengembangan sistem logging jaringan berbasis *web* untuk mencatat semua log aktivitas dan deteksi anomali," Jurnal Informatika dan Teknik Elektro Terapan, vol. 14, no. 1, 2026. [Online]. Available: https://doi.org/10.23960/jitet.v14i1.8235

[4] "VirusTotal integrated Wazuh configuration automation using Ansible to detect and protect against malware attacks," MelekIT, vol. 11, no. 1, pp. 55–66, 2025. [Online]. Available: https://doi.org/10.30742/melekitjournal.v11i1.403

[5] A. A. Pertiwi, D. Kiswanto, S. P. Syaifullah Nst, dan M. R. Akbar Tjg, "Implementasi sistem logging jaringan berbasis *web* untuk *monitoring* dan analisis *real-time*," JAKAKOM, vol. 6, no. 1, 2026. [Online]. Available: https://doi.org/10.33998/jakakom.2026.6.1.2639

[6] B. Rasyidi dan F. Pratama, "Sistem *Monitoring Server* di PT. XYZ Media Indonesia Berbasis Grafana dan Prometheus: *Server Monitoring* System at PT. XYZ Media Indonesia Based on Grafana and Prometheus," MALCOM, vol. 4, no. 4, pp. 1456–1465, Sep. 2024. [Online]. Available: https://journal.irpi.or.id/index.php/malcom/article/view/1546

[7] K. W. B. Izza dan W. E. Sholeha, "Implementasi sistem *monitoring server* berbasis Linux pada ICT Politeknik Negeri Tanah Laut," JITRENDS, vol. 2, no. 2, 2025. [Online]. Available: https://doi.org/10.51817/jitrends.v2i2.35

[8] A. H. Sukma, D. Kiswanto, F. G. Sembiring, dan S. Ashillah, "Rancang bangun sistem logging jaringan berbasis *web* dengan visualisasi interaktif Grafana untuk analisis aktivitas IP mencurigakan," Jurnal Informatika dan Teknik Elektro Terapan, vol. 14, no. 1, 2026. [Online]. Available: https://doi.org/10.23960/jitet.v14i1.8200

[9] M. Z. Hidayat, A. Widi Pratama, dan A. Hamdani, "Implementasi Veyon *server* sebagai sistem *monitoring* dan *remote control* komputer pada laboratorium Fakultas Sains & Teknologi Universitas Ibrahimy," JSI (Jurnal Sistem Informasi) Universitas Suryadarma, vol. 13, no. 1, pp. 17–27, 2026. [Online]. Available: https://doi.org/10.35968/jsi.v13i1.1695

[10] F. A. Sihombing, N. N. Zuhra, L. Zahra, R. Alfarisyi, H. Nisa, dan R. A. Astiyanto, "Implementasi keamanan jaringan menggunakan firewall dan intrusion detection system (IDS) pada infrastruktur jaringan skala kecilmenengah," JIKUM, vol. 2, no. 1, pp. 124–126, Jan. 2026. [Online]. Available: https://journal.dcircle.org/index.php/jiku m/article/view/184

[11] "Implementasi sistem *monitoring* keamanan jaringan menggunakan log analysis," JISSI, vol. 3, no. 2, pp. 30–34, Jan. 2026. [Online]. Available: https://journal.smartpublisher.id/index.ph p/jissi/article/view/1433

[12] D. Permata Sari, M. A. Cendekia Putra, dan RD. Kusumanto, "Implementasi Pengenalan Wajah Berbasis CNN dan RFID untuk Area Akses Aman di Fasilitas Polsri dengan Hosting Lokal," Jurnal TELISKA, vol. 17, no. II, pp. 23–31, Jul. 2024. Available: https://jurnal.polsri.ac.id/index.php/teliska/issue/view/555.