JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 14 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v14i1.8201

SISTEM LOGGING JARINGAN BERBASIS WEBSITE DENGAN IMPLEMENTASI NOTIFIKASI REAL-TIME UNTUK PERINGATAN AKSES MENCURIGAKAN

Raihan Insan Pratama[1*], Dedy Kiswanto[2],Lastri Elisabet Butarbutar[3], Feby Juliana Silalahi[4],

[1,2,3]Program Studi Ilmu Komputer, Universitas Negeri Medan, Medan, Indonesia

Keywords:
Sistem Logging, Keamanan Jaringan, Notifikasi Real-Time, Flask, PostgreSQL

Corespondent Email:
raihansiagian218@mhs.unimed.ac.id

Abstrak. Perkembangan teknologi jaringan komputer yang pesat menimbulkan tantangan baru dalam menjaga keamanan sistem dari ancaman seperti akses ilegal dan serangan *brute-force*. Untuk mengatasi hal tersebut, penelitian ini mengembangkan Sistem Logging Jaringan Berbasis Website dengan Implementasi Notifikasi Real-Time untuk Peringatan Akses Mencurigakan. Tujuan penelitian ini adalah membangun sistem yang mampu mencatat aktivitas jaringan dan memberikan peringatan otomatis terhadap akses mencurigakan. Metode yang digunakan adalah Research and Development (R&D) dengan tahapan analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi. Sistem dikembangkan menggunakan Flask sebagai *backend*, PostgreSQL sebagai basis data, serta Socket.IO untuk komunikasi real-time. Pengujian menggunakan metode Black Box Testing menunjukkan bahwa sistem mampu mendeteksi aktivitas login mencurigakan secara real-time, mengirimkan notifikasi otomatis ke dashboard admin dan pengguna, serta menyimpan data log secara terstruktur. Kesimpulannya, sistem ini efektif dalam meningkatkan keamanan jaringan dan dapat dikembangkan lebih lanjut dengan integrasi *machine learning* untuk mendeteksi pola serangan yang lebih kompleks.

Abstract. *The rapid development of computer network technology poses new challenges in maintaining system security from threats such as illegal access and brute-force attacks. To address this issue, this study developed a Website-Based Network Logging System with Real-Time Notification Implementation for Suspicious Access Alerts. The objective of this study was to build a system capable of recording network activity and providing automatic alerts for suspicious access. The method used is Research and Development (R&D) with stages of needs analysis, design, implementation, testing, and evaluation. The system was developed using Flask as the backend, PostgreSQL as the database, and Socket.IO for real-time communication. Testing using the Black Box Testing method showed that the system was able to detect suspicious login activity in real-time, send automatic notifications to the admin and user dashboards, and store log data in a structured manner. In conclusion, this system is effective in improving network security and can be further developed with the integration of machine learning to detect more complex attack patterns.*

1. PENDAHULUAN

Keamanan jaringan telah menjadi salah satu aspek krusial di era digital yang semakin terhubung. Perkembangan teknologi informasi mendorong pertumbuhan layanan berbasis internet secara masif, mulai dari transaksi

perbankan, komunikasi daring, hingga pengelolaan infrastruktur kritis. Namun, kemajuan ini juga diiringi dengan meningkatnya ancaman siber yang dapat mengganggu ketersediaan, kerahasiaan, dan integritas data. Peningkatan akses internet dan konektivitas antarperangkat berdampak pada tingginya risiko serangan siber salah satu ancaman yang paling menonjol adalah serangan Distributed Denial of Service (DDoS), yang dapat melumpuhkan layanan dalam hitungan detik dengan membanjiri server atau jaringan target menggunakan lalu lintas berlebihan. Oleh karena itu, dibutuhkan sistem yang mamppu melakukan pencatatan (*logging)* aktivitas jaringan dan memberikan peringatan secara real-time ketika terjaidi aktivitas mencurigakan. Sistem logging jaringan berbasis website menjadi salah satu solusi efektif karena dapat menampilkan informasi aktivitas jaringan melalui antarmuka web yang mudah diakses dan dikelola oleh administrator [1].

Penelitian terdahulu telah banyak mengkaji sistem monitoring dan deteksi intrusi berbasis IDS dan media notifikasi. mengembangkan sistem *Intrusion Detection System (IDS)* menggunakan Snort yang terintregasi dengan Telegram, dan berhasil memberikan peringatan otomatis terhadap aktivitas berbahaya seperti *port scanning* dan *buffer overflow.* Qomarudin [2] merancang sistem monitoring jaringan real-time berbasis *Internet Control Massage Protocol (ICMP)* untuk memantau konektivitas antarperangkat melalui web dashboard. Penelitian pada Prosiding SNIV [3] menerapkan Zeek IDS yang menghasilkan log aktivitas jaringan secara detail dan mengirimkan notifikasi langsung melalui Telegram.

Implementasi lain menggunakan Zabbix dan Grafana [4], yang menunjukkan efektivitas visualisasi data log jaringan serta pengiriman notifikasi otomatis terhadap gangguan server. Nisa [5] meneliti analisis log server untuk mendeteksi serangan DDoS, yang menjadi dasar dalam pengembangan sistem notifikasi berbasis pola serangan. Sobah dan Amrulloh [6] mengembangkan sistem montitoring jaringan berbasis web dengan intregasi API Mikrotik untuk mempemudah deteksi trafik abmodrmal secara real-time.

Parenreng [7] membangun sistem logging berbasis PostgreSQL untuk mencatat aktivitas *Brute force* pada router Mikrotik, memperlihatkan efisiensi basis data dalam mengelola log besar. Ardiyansyah [8] menerapkan IDS berbasis Snort dengan integrasi *chatbot Telegram* yang dapat memberikan notifikasi secara otomatis ketika terjadi aktivitas mencurigakan pada jaringan. Rusli [9] mengembangkan aplikasi web untuk monitoring server dan pencatatan log yang terintregasi agar administrator dapat dengan mudah menganalisis performa server. Maududy [10] memperluas konsep logging berbasis web dengan penerapan *data logging real-time* yang diadaptasi dari sistem industri otomatisasi, menunjukkan kesamaan kebutuhan antara pemantauan produksi dan pemantauan jaringan.

2. TINJAUAN PUSTAKA

2.1. Konsep Sistem Logging Jaringan

Sistem logging jaringan merupakann mekanisme pencatatan aktivitas jaringan yang digunakan untuk memantau dan menganalisis setiap peristiwa yang terjadi pada infrastruktur jaringan. Menurut penelitian Nur'Aini, penggunaan teknologi *Internet of Things (IoT)* berbasis cloud seperti AWS Cloud dapat meningkatkan efisiensi penyimpanan dan transmisi data log jaringan. Sistem ini tidak hanya mencatat seluruh aktivitas komunikasu antarperangkat, tetapi juga mampu memberikan peringatan secara otomatis melalui media notifikasi seperti Telegram. Penerapan konsep ini menjadi dasar penting dalam pengembangan sistem logging jaringan berbasis website karena memungkinkan administrator melakukan pemantauan jaringan secara terpusat dan real-time [11].

2.2. Implementasi Monitoring Jaringan Berbasis Web

Monitoring jaringan berbasis web berperan penting dalam meningkatkan efektivitas pengawasan terhadap infrastruktur jaringan. Penelitian yang dilakukan oleh **Ulum dan Badri (2023)** pada *Jurnal Informatika dan Teknik Elektro Terapan (JITET)* menunjukkan bahwa penerapan sistem monitoring berbasis IoT yang terintegrasi dengan

antarmuka web mampu menampilkan data sensor secara real-time melalui jaringan internet. Sistem tersebut dirancang untuk memberikan informasi kondisi lingkungan secara langsung sekaligus mengirimkan peringatan otomatis apabila terdeteksi nilai ambang batas tertentu. Konsep ini dapat diadaptasi pada implementasi monitoring jaringan berbasis web, di mana data aktivitas jaringan dikirimkan ke server untuk divisualisasikan melalui dashboard web interaktif. Dengan dukungan teknologi ini, administrator jaringan dapat memantau status perangkat dan trafik secara efisien, responsif, serta dapat diakses dari berbagai lokasi tanpa batasan waktu [12].

2.3. Penerapan Notifikasi Real-Time pada Sistem Monitoring

Notifikasi real-time merupakan komponen pentng yang memastikan administrator segera mengetahui adanya perubzhan status pada perangkat jaringan. Fitriana mengembangkan sistem monitoring status perangkat berbasis web dengan integrasi *Laravel Scheduler* dan API WhatsApp. Sistem ini mengirimkan notifikasi otomatis ketika perangkat jaringan mengalami gangguan, sehingga mempercepat proses respon dan perbaikan. Penerapan sistem notifikasi real-time terbukti efektif dalam meminimalkan waktu tanggap (*response time*) terhadap permasalahan jaringan dan menjadi landasan bagi pengembang sistem akses mencurigakan [13].

2.4. Efisiensi Monitoring Jaringan di Lingkungan Perusahaan

Efisiensi monitoring jaringan menjadi kebutuhan utama bagi perusahaan yang memiliki banyak perangkat dan koneksi data. YUazijah melakukan penelitian mengenai penerapan sistem monitoring jaringan berbasis web di PT Atlas Lintas Indonesia. Hasil penelitian menunjukkan bahwa sistem ini mampu menampilkan kondisi perangkat, aktivitas pengguna, serta lalu lintas data jaringan melalui dashboard terpusat. Dengan memanfaatkan antarmuka web, administrator dapat dengan cepat mengidentifikasi masalah dan mengoptimalkan performa jaringan tanpa harus melakukan pengecekan manual pada setiap perangkat [14].

2.5. Aplikasi Notifikasi Troubleshooting Jaringan

Salah satu inovasi dalam sistem pengawasan jaringan adalah pengembangan aplikasi notifikasi otomatis untuk *troubleshooting.* Sintar merancang aplikasi *SINTAR* yang mampu mengirimkan peringatan secara langsung kepada teknisi jaringan ketika terjadi gangguan pada perangkat atau koneksi. Sistem ini memanfaatkan teknologi notifikasi real-time untuk mempercepat penanganan permasalahan dan mengurangi waktu pemulhan layanan (*downtime*). Konsp aplikasi ini menunjukkan bahwa integrasi antara logging, monitoring, dan sistem peringatan otomatis dapat meningkatkan keandalan operasional jaringan [15].

3. METODE PENELITIAN

3. 1 Jenis Penelitian

Jenis Penelitian yang digunakan dalam penelitian ini adalah Research and Development (R&D) yang berfokus pada perancangan dan pembuatan sistem berbasis website untuk mendukung keamanan jaringan. Tujuan utama penelitian ini adalah mengambangkan sebuah sistem yang mampu mencatat aktivitas jaringan (logging) serta memberikan notifikasi secara real-time ketika terdeteksi adanya akses yang mencurigakan.

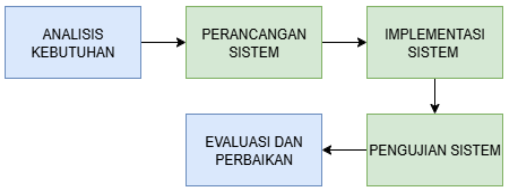
3. 2 Tahapan Penelitian

1. Analisis Kebutuhan

   Pada tahap ini dilkukan identifikasi terdahap kebutuhan pengguna dan sistem. Analisis meliputi data apa saja yang perlu dicatat dalam log jaringan, kondisi yang dianggap mencurigakan, serta mekanisme notifikasi real-time yang dibutuhkan untuk memberikan peringatan dini terhadap akses yang tidak normal.
2. Perancangan Sistem

Setelah kebutuhan dianalisis, tahap selanjutnya adalah merancang arsitektur sistem, basis data, serta antarmuka pengguna (user Interface). Sistem dirancang menggunakan konsep client-server agar data log dapat dikelola oleh server dan ditampilkan melalui antarmuka web yang responsif.

3. Implementasi Sistem
   Tahap ini mencakup proses pembuatan aplikasi sesuai rancangan yang telah dibuat. Bahasa pemrograman dan teknologi web digunakan untuk membangun sistem, seperti HTML, CSS, dan JavaScript untuk tampilan antaramuka, serta PHP atau Python pada sisi server. Fitur notifikasi real-time diimplementasikan dengan teknologi WebSocket.
4. Pengujian Sistem
   Sistem yang telah dikembangkan diuji menggunakan metode black box testing untuk memastikan seluruh fungsi berjalan sesuai kebutuhan. Selain itu, dilakukan simulasi aktivitas jaringan untuk menguji kemampuan sistem dalam mendeteksi akses mencurigakan dan mengirimkan notifikasi secara real-time.
5. Evaluasi dan Perbaikan
   Hasil pengujian dievaluasi untuk menilai kendala dan efektivitas sistem. Jika ditemukan kekurangan, maka dilakukan perbaikan agar sistem dapat berjalan optimal dan memberikan hasil yang akurat dalam memantau aktivitas jaringan.



Gambar 1. Tahapan Penelitian

3. 3 Alat dan Bahan

Dalam pengembangan sistem ini digunakan beberapa alat dan bahan sebagai berikut:

1. Perangkat keras (hardware)
   Perangkat keras yang digunakan dalam pengembangan sistem meliputi laptop dengan spesifikasi minimal prosesor Intel Core i3 generasi ke-12 atau setara, RAM sebesar 8 GB, penyimpanan SSD 256 GB, serta sistem operasi Windows 10 atau 11. Perangkat ini berfungsi untuk menjalankan server lokal, menguji apliasi web, dengan mengelola basis data. Selain itu, digunakan pula router atau jaringan lokal (LAN/Wifi) yang berperan dalam menghubungkan beberapa perangkat selama proses pengujian sistwm. Melalui jaringan ini, sistem dapat diakses secara lokal menggunakan alamat IP, sehingga memungkinkan pengujian multi-user dan deteksi aktivitas jaringan secara langsung.
2. Perangkat lunak (software)
   Perangkat lunak yang digunakan terdiri atas berbagai aplikasi dan framework pendukung pengembangan sistem. Bahasa pemrograman utama yang digunakan adalah Python versi 3.10 atau lebih tinggi, dengan Flash Framework sebagai basis pengembangan aplikasi web, manajemen session, serta pengaturan routing. Basis data sistem menggunakan PostgreSQL, yang berfungsi menyimpan data pengguna, log aktivitas, token reset password, serta data alert keamanan. Pengelolaan database dilakukan melalui antarmuka pgAdmin 4, sementara proses penulisan kode, debugging, dan integrasi fitur dilakukan di Visual Studi Code sebagai editor utama. Pengujian tampilan antarmuka dilakukan menggunakan browser Google Chrome atau Microsoft Edge, sedangkan Virtual Environment (venv) digunakan utnuk emngisolasi dependensi Python agar tidak menganggu sistem global.
3. Bahan
   Bahan yang digunakan dalam pengambangan sistem meliputi data dan pustaka pendukung. Salah satunya adalah dataset log aktivitas sistem (simulasi) yang berisi data

percobaan login sukses dan gagal dari beberapa alamat IP. Dataset ini digunakan untuk menguji fitur deteksi brute-force dan notifikasi real-time. Selain itu, digunakan berbagai modul dan library Python seperti Flask-Mail, Flask-SocketIO, dan Flask-SQLAlchemy untuk menangani routing, notifikasi, serta komunikasi real-time. Modul itsdangerous digunakan untuk pembuatan dan validasi token verifikasi akun, sedangkan werkzeug.security berfungsi untuk proses hashing dan validasi kata sandi. Modul datetime, os, random, dan sqlalchemy mendukung pengelolaan waktu, file, serta logika sistem. Untuk antarmuka pengguna, digunakan template HTML, CSS, dan JavaScript sebagai tampilan halaman login, dashboard admin, dan dashboard user. Selain itu, email dummy berbasis SMTP Gmail digunakan untuk menguji pengiriman email verifikasi akun dan pemulihan kata sandi.

4. HASIL DAN PEMBAHASAN

4.1 Hasil Implementasi Sistem

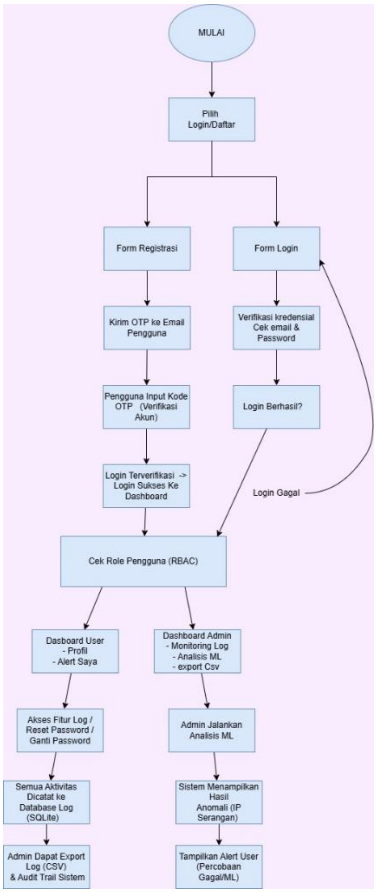
Hasil dari penelitian ini adalah sebuah sistem logging jaringan berbasis website dengan implementasi notifikasi real-time untuk peringatan akses mencurigakan, yang dinamakan SKD&J (Sistem Keamanan Data & Jaringan. Sistem ini dibangun menggunakan framework Flask (python) sebagai backend, PostgreSQL sebagai basis data, serta TailwindCSS dan Chart.js untuk tampilan dan visualisasi data.

Sistem ini memiliki dua peran utama pengguna, yaitu user biasa dan administrator (admin). User berfungsi untuk memantau keamanan akunnya secara personal, sedangkan admin memiliki hak untuk mengelola seluruh log dan memantau aktivitas jaringan secara menyeluruh.

4.2 Arsitektur dan Alur Sistem

Secara umum, sistem terdiri dari tiga komponen utama. Backend Flask Server, yang berfungsi menerima event log melalui API endpoint dan menjalankan logika pendeteksian brute-force secara real-time. Database PostgreSQL, yang menyimpan data log ke tabel *log_entries* menggunakan SQLAlchemy ARM. Fronted Web, yang menampilkan hasil logging dan peringatan secara visual kepada user dan admin.

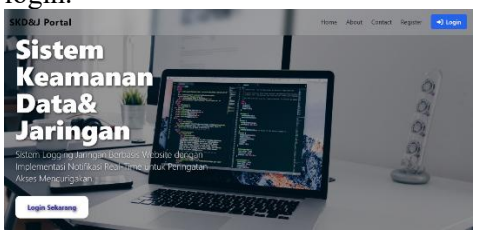
Ketika terjadi percobaan login ke sistem, setiap kejadian baik berhasil maupun gagal akan dikirim ke server dan disimpan dalam database. Sistem kemudian memeriksa pola failed login berulang dari alamat IP yang sama dalam waktu tertentu. Jika melebihi ambang batas, maka akan dipicu log bertipe ALERT yang dikirim ke dashboard secara real-time menggunakan Socket.IO.



Gambar 2. Alur Sistem

4.3 Tampilan dan Fungsionalitas Sistem

1. Halaman Utama (Home Page)
   Halaman utama menampilkan identitas portal "Sistem Keamanan Data & Jaringan (SKD&J Portal)" dengan desain hero section berlatar gambar penuh. Tampilan ini menggunakan TailwindCSS untuk menjaga tampilan tetap modern, responsif, dan elegan. Tombol "Login Sekarang" berfungsi sebagai call-to-action yang mengarahkan pengguna ke halaman login.

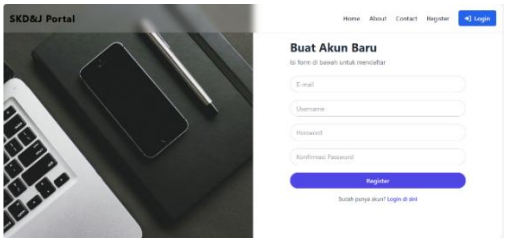


Gambar 3. Tampilan halaman utama SKD&J Portal

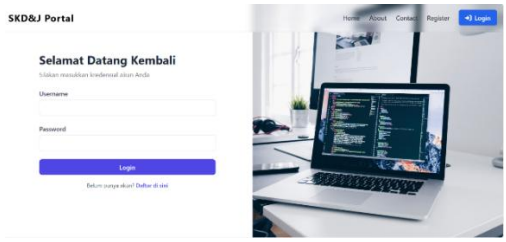
2. Halaman About
   Halaman ini menjelaskan tujuan pengembangan sistem, yaitu mendeteksi dan mencegah aktivitas login mencurigakan atau serangan *brute-force* secara real-time. Selain itu, ditampilkan juga informasi anggota tim pengembang dan pembagian perannya.

Gambar 4. Tampilan halaman About

3. Halaman Login dan Registrasi
   Halaman Login menjadi pintu masuk ke sistem, di mana pengguna memasukkan username dan password. Proses autentikasi dilakukan dengan hashing password untuk mencegah penyimpanan plaintext. Setiap percobaan login gagal otomatis dicatat dalam database, dan apabila jumlah kegagalan dari satu IP melebihi ambang batas, sistem memicu notifikasi real-time ke dashboard.
   Halaman Registrasi memungkinkan pengguna membuat akun baru agar setiap login bisa terindentifikasi dengan aman.



Gambar 5. Tampilan halaman register



Gambar 6. Tampilan halaman Login

4. Dashboard User
   Dashboard user menyediakan beberapa halaman utama yang membantu pengguna memantau keamanan aunnya secara real-time, yaitu halaman Beranda yang menampilkan ringkasan aktivitas login pengguna dalam bentuk grafik doughnut yang memperbandingkan jumlah login suskses dan gagal. Tersedia juga informasi singkat seperti total akses dan jumlah alert aktif.
   Kemudian ada halaman Aktiitas yang berisi daftar riwayat login akun pengguna, termasuk waktu, IP sumber, dan status autentikasi. Halaman ini membantu pengguna memeriksa adanya aktivitas login yang mencurigakan.
   Selanjutnya ada halaman Alert Saya yang menampilkan daftar alert yang dihasilkan sistem akibat percobaan login berulang atau aktivitas mencurigakan. Setiap alert ditampilkan dengan detail waktu, IP sumber, dan pesan peringatan.
   Terakhir halaman Profil yang berfungsi untuk mengubah kata sandi secara aman. Sistem memverifikasi kata sandi lama dan mengenkripsi kata sandi baru sebelum disimpan ke database.

Gambar 6. Dashboard User

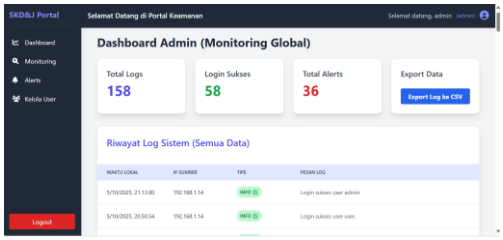
5. Dashboard Admin

Dashboard Admin digunakan untuk memantau seluruh aktivitas jaringan dan pengguna dalam sistem. haalman utamanya meliputi Halama Dashboard Utama yang menampilkan indikator keamanan utama seperti total log, login sukses, dan jumlah alert. Data ditampilkan secara dinamis untuk membantu admin menilai kondisi sistem.

Kemudian ada halaman Monitoring & Analisis Jaringan yang memvisualisasikan tren aktivitas login dan distribusi tipe log menggunakan grafik. Hal ini membantu admin mendeteksi lonjakan aktivitas yang tidak normal.

Selanjutnya ada halaman Alert Sistem Global yang menampilkan seluruh alert dari seluruh pengguna secara real-time, lengkap dengan waktu dan IP sumber. Halaman ini menjadi pusat pemantauan ancaman sistem.

Terakhir Halaman Kelola User yang menampilkan daftar pengguna dna perannya dalam sistem. Hanya admin yang memiliki akses ke halaman ini sebagai bagian dari penerapan Role-Based Access Control.



Gambar 7. Dashboard Admin

4.4 Pengujian Sistem (Blackbox testing)

Untuk memastikan sistem yang dikembangkan berfungsi sesuai kebutuhan, dilakukan pengujian menggunakan metode Black Box Testing. Metode ini berfokus pada pengujian fungsi sistem dari sisi pengguna tanpa memperhatikan struktur internal kdoe program. Setiap komponen diuji berdasarkan input dan output yang dihaislkan, sehingga dapat diketahui apakah fitur berjalan sesuai spesifikasi.

No.	Fungsi Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Aktual	Kesimpulan
1.	Akses Publik & Navigasi	Pengguna mengakses alamat IP aplikasi pertama kali (/home).	Halaman Home (Landing Page) dimuat dengan header transparan dan link Login serta Register.	Tampilan Home muncul, header transparan berfungsi, link Login dan Register dapat diakses.	Berhasil
2.	Registrasi User Baru	Pengguna mengisi form Register (Username, Email, Password) dan menekan Register.	Pesan sukses ditampilkan. Link konfirmasi akun dikirimkan ke email terdaftar.	Pesan sukses ditampilkan. Link verifikasi email tercetak di konsol Flask (simulasi pengiriman email).	Berhasil
3.	Login dan RBAC	Pengguna login menggunakan akun Admin yang terverifikasi (Username: admin).	Sistem mengarahkan ke halaman Dashboard Admin (/dashboard/admin) yang menampilkan Sidebar lengkap.	Redirect berhasil. Sidebar dimuat dengan menu Kelola User dan Monitoring.	Berhasil
4.	Otorisasi (Access Control)	User biasa login (role: user) dan mencoba mengakses endpoint Admin (/admin/users).	Sistem menolak akses dan mengembalikan error 403 Forbidden atau me-redirect kembali ke Dashboard User.	Akses ditolak. Pesan Access Denied / redirect ke halaman User terjadi.	Berhasil
5.	Fitur Lupa Password	Pengguna mengklik Lupa Password? dan memasukkan Username terdaftar.	Sistem menampilkan pesan sukses (message "Link pemulihan telah dikirimkan...") dan link unik tercetak di konsol.	Link pemulihan yang valid (tidak kedaluwarsa) tercetak. Pesan sukses muncul di frontend.	Berhasil

6.	Deteksi Brute-Force	Admin menjalankan simulasi 3 failed logins dari Source IP yang sama dalam 60 detik.	Backend mencatat log ALERT (Sesi ditutup) di database.	Log ALERT berhasil dicatat. Admin dapat melihat Alert tersebut di menu Alerts Global.	Berhasil
7.	Integrasi Alert User	User login dan membuka menu Alert Saya (/user/alerts).	Tabel hanya menampilkan Alerts spesifik yang dihasilkan oleh sistem (e.g., Sesi ditutup, percobaan ke-3). membuktikan filtering ketat berfungsi.	Tabel menampilkan alerts yang relevan (bukan log INFO gagal login biasa).	Berhasil
8.	Ubah Password	Pengguna di halaman Profil mencoba mengubah password dengan password lama yang salah.	Sistem menampilkan pesan error "Password lama salah" tanpa mengizinkan perubahan.	Pesan error muncul. Database tidak diperbarui (diverifikasi melalui login ulang).	Berhasil
9.	Export Data	Admin menekan tombol Export Log ke CSV di Dashboard Admin.	File .csv berisi semua log sistem berhasil diunduh ke perangkat.	File CSV (security_logs_[tanggal].csv) berhasil di-generate dan diunduh.	Berhasil
10	Verifikasi Email	Pengguna register dengan email baru, lalu mencoba login sebelum mengklik link verifikasi.	Sistem menolak login dan menampilkan pesan: "Akun Anda belum diverifikasi!"	Pesan penolakan muncul. Login tidak diizinkan sampai verifikasi email diklik.	Berhasil

12	Ubah Password Aman	User login dan di halaman Profil mengubah password dengan memasukkan password lama yang BENAR.	Sistem meng-hash password baru ke DB dan menampilkan pesan sukses "Password Anda berhasil diperbarui."	Password baru berfungsi untuk login selanjutnya. Log sukses tercatat di DB.	Berhasil
13	Lupa Password & Reset	Pengguna meminta link pemulihan (Forgot Password) lalu mengklik link token di console (simulasi email).	Halaman Reset Password dimuat. Token diterima dan divalidasi (tidak kedaluwarsa).	Link pemulihan valid. Form Reset Password muncul (bukan error kedaluwarsa).	Berhasil
14	Deteksi Brute-Force (Rule-Based)	Attacker (Kali Linux) mengirim 4 failed logins dalam 5 detik ke route /login.	Detection Engine membuat Log ALERT (percobaan ke-3 dan Sesi ditutup).	Log ALERT berhasil dibuat di DB. Alert muncul di Dashboard Admin.	Berhasil
15	Filtering Alert User	User login dan membuka menu Alert Saya. (Setelah skenario No. 5).	Tabel hanya menampilkan Alerts yang dihasilkan sistem (e.g., Sesi ditutup), mengabaikan log INFO gagal login biasa.	Filtering berhasil. Tabel menyajikan Alerts yang ringkas dan relevan.	Berhasil
16	Deteksi Anomali (ML)	Admin memicu Jalankan Analisis ML setelah traffic dicampur (5 IP normal + 1 IP brute-force intensif).	Model Isolation Forest mengklasifikasikan IP attacker sebagai Anomaly (Score negatif) dan mencatatnya.	Analisis menunjukkan ANOMALI KRITIS: 1 IP terdeteksi sebagai serangan. Anomaly Score ditampilkan sebagai bukti.	Berhasil

Gambar 8. Blackbox Testing

Pengujian dilakukan terhadap berbagai fungsi utama sistem, seperti login, registrasi, kontrol akses, deteksi brute-force, pengiriman alert real-time, manajemen profil, hingga ekspor data log. Setiap skenario pengujian dirancang untuk memverifikasi bahwa sistem merespons dengan benar terhadap kondisi normal maupun kondisi akses ilegal.

4.5 Pembahasan

Hasil prngujain menunjukkan bahwa sistem mampu mendeteksi serangan brute-force secara real-time dan mengirimkan peringatan otomatis kepada pengguna melalui dashboard admin maupun user. Seluruh aktivitas jaringan berhasil direkam dan disimpan alam baris data PostgreSQL dengan struktur tabel yang terorganisir. Data log yang tersimpan mencakup kolom timestamp, sourc_ip, log_type, dan massage, sehingga setiap percobaan login baik yang berhasil maupun gagal dapat dilacak dengan jelas. Selain itu, fitur notifikasi real-time berhasil mengirimkan alert secara langsung ketika abang batas percobaan login tercapai, membuktikan bahwa sistem berfungsi sesuai rancangan dan efektif dalam mendeteksi antivitas mencurigakan sejaka dini.

Sistem juga mampu menampilkan data log dan alert secara interaktif serta responsif melalui antarmuka web, sehingga pengguna dan admin dapat memantau aktivitas jaringan dengan mudah. Mekanisme Role-Based Access Control (RBAC) berjalan dengan baik, memastikan setiap pengguna hanya dapat mengakses fitur sesuai perannya. Secara keseluruhan, penerapan sistem ini meningkatkan kesadaran dan respons pengguna terhadap pontensi ancaman keamanan jaringan. Integrasi antara visualisasi data dan notifikasi real-time menjadikan sistem ini tidak hanya sebagai alat pencatat log, tetapi juga sebagai sarana pemantauan aktif yang mendukung keamanan jaringan.

5. KESIMPULAN

a. Penelitian ini berhasil mengembangkan sistem logging jaringan berbasis website yang mampu mencatat aktivitas login dan mendeteksi akses mencurigakan secara real-time.
b. Implementasi teknologi Flask, PostgreSQL, dan Socket.IO memungkinkan sistem melakukan pencatatan log secara terstruktur sekaligus memberikan notifikasi otomatis ketika terdeteksi serangan brute-force
c. Hasil pengujian menunjukkan seluruh fitur utama berjalan sesuai spesifikasi, termasuk autentikasi pengguna, kontrol akses berbasis peran, serta pengiriman notifikasi perigatan secara langsung ke dashboard admin dan user
d. Sistem ini efektif dalam meningkatkan kesadaran dan respons pengguna terhadap ancaman keamanan jaringan, sekaligus berpotensi dikembangkan lebih lanjut menambahkan integrasi machine learning untuk deteksi serangan yang lebih kompleks.

DAFTAR PUSTAKA

[1] D. Kiswanto, F. Ramadhani, N. M. Surbakti, and N. A. Nasution, "Pengembangan dan Implementasi Sistem Deteksi Serangan DDoS Berbasis Algoritma Random Forest," vol. 6, no. 3, 2025, doi: 10.47065/bit.v5i2.2203.

[2] Qomarudin, M. F. (2022). *Sistem Monitoring Jaringan Real-Time Berbasis Internet Control Message Protocol (ICMP)*. JINTECH: Journal of Information Technology, 3(2), 55–62.https://journal.ar-raniry.ac.id/index.php/jintech/article/view/1935

[3] Prosiding SNIV. (2025). *Rancang Bangun Intrusion Detection System Berbasis Zeek untuk Deteksi Serangan dan Notifikasi Telegram*. Seminar Nasional Inovasi Vokasi (SNIV), 2025, 233–240. https://prosiding.poliupg.ac.id/index.php/sniv/article/view/404

[4] Zabbix–Grafana Integration Team. (2025). *Implementasi Sistem Monitoring Jaringan dan Server Menggunakan Zabbix Terintegrasi dengan Grafana dan Telegram*. Jurnal Teknologi dan Sistem Informasi, 4(2), 88–96. https://jurnal.bsi.ac.id/index.php/jinsan/article/view/2432

[5] Nisa, A. R. (2024). *Analisis Log Server untuk Mendeteksi Serangan DDoS pada Keamanan Jaringan Website*. PJISE: Jurnal Pengembangan Ilmu dan Sistem Elektronika, 3(1), 17–26. https://journal.pubmedia.id/index.php/pjise/article/view/2612

[6] Sobah, N., & Amrulloh, M. F. (2023). *Perancangan dan Implementasi Sistem Monitoring Jaringan di MA Darut Taqwa Berbasis Web Mengintegrasikan API Mikrotik*. BIOS: Jurnal Teknologi Informasi dan Rekayasa Komputer, 6(2), 64–72. https://bios.upnjatim.ac.id/index.php/bios/article/view/232

[7] Parenreng, J. M. (2025). *Implementasi Log Mikrotik Berbasis Database PostgreSQL untuk Pencatatan Serangan Brute Force*. Jurnal Informatika Terapan, 9(1), 33–41. https://jurnal.itats.ac.id/index.php/jitter/article/view/1978

[8] Ardiyansyah, F. (2024). *Implementasi IDS pada Jaringan Komputer Menggunakan Snort yang Terintegrasi dengan Chatbot Telegram untuk Notifikasi Real-Time*. MalCom: Journal of Computer Science and Communication, 8(2), 27–35. https://journal.malcom.or.id/index.php/malcom/article/view/874

[9] Rusli, M. (2022). *Aplikasi Sistem Monitoring Server Menggunakan Device Web Logging*. Jurnal Sistem Komputer dan Informatika, 10(1), 12–20. https://jurnal.unmuhjember.ac.id/index.php/jsik/article/view/1572

[10] Maududy, R. (2023). *Pengembangan Real-Time Monitoring dan Data Logging Berbasis Web pada Proses Robot Painting untuk Meningkatkan Efisiensi Produksi*. INDEX: Informatics and Digital Expert Journal, 4(3), 109–118. https://journal.literasisains.id/index.php/index/article/view/1523

[11] Nur'Aini, D. P. (2024). *Analisis Performa Transmisi Data Log Berbasis IoT Cloud (AWS) dan Notifikasi Telegram*. IJAI: Indonesian Journal of Applied Informatics, 5(1), 23–30. https://journal.uns.ac.id/ijai/article/view/3741

[12] Ulum, M. B., & Badri, F. (2023). *Sistem Monitoring Cuaca dan Peringatan Banjir Berbasis IoT Menggunakan Aplikasi MIT App Inventor*. Jurnal Informatika dan Teknik Elektro Terapan (JITET), 11(3), 178–186. https://doi.org/10.23960/jitet.v11i3.3088

[13] Fitriana, D. (2025). *Rancang Bangun Website Monitoring Status Perangkat dengan*

Notifikasi WhatsApp (Laravel Scheduler). Prosiding SNIV: Seminar Nasional Inovasi Vokasi, 2025, 181–188. https://prosiding.poliupg.ac.id/index.php/sniv/article/view/403

[14]Yuazijah, A. (2024). *Sistem Monitoring Jaringan Berbasis Web pada PT Atlas Lintas Indonesia*. Jurnal Teknologi Informasi dan Komunikasi, 6(2), 45–53. https://ejurnal.itn.ac.id/index.php/jtik/article/view/2539

[15] Sintar, R. (2023). *Aplikasi Sistem Notifikasi Troubleshooting Jaringan (SINTAR)*. Jurnal Rekayasa Sistem Komputer dan Jaringan, 5(3), 91–99. https://ejurnal.resolusi.ac.id/index.php/jrskj/article/view/914