Department of Digital Business
Journal of Artificial Intelligence and Digital Business (RIGGS)
Homepage: https://journal.ilmudata.co.id/index.php/RIGGS
Vol. 4 No. 4 (2026) pp: 13862-13869
P-ISSN: 2963-9298, e-ISSN: 2963-914X

Analisa Deteksi Dan Mitigasi Serangan SQL Injection Menggunakan SIEM Security Onion : Studi Kasus Infrastruktur Teknologi Informasi PT Neotech Cakrawala Indonesia

Farhan Fauzan[1], Muhammad Helmy Fakhruddin[2], Anwar T. Sitoroes[3], Samroh[4]
[1,2,3,4]Sistem Informasi, Sekolah Tinggi Manajemen dan Ilmu Komputer Mercusuar
[1]farhan270402@gmail.com, [2]muhammadhelmyf@gmail.com, [3]anwar@mercusuar.ac.id, [4]samroh74@gmail.com

Abstrak

Serangan SQL Injection merupakan salah satu ancaman keamanan aplikasi web yang masih sering terjadi akibat lemahnya validasi input dan pengelolaan kueri basis data. Serangan ini berpotensi menyebabkan kebocoran data, gangguan layanan, hingga kerugian finansial dan reputasi organisasi. Oleh karena itu, diperlukan mekanisme deteksi dini dan penanganan insiden yang efektif serta terstruktur. Penelitian ini bertujuan untuk menganalisis kemampuan sistem Security Information and Event Management (SIEM) berbasis open source, yaitu Security Onion, dalam mendeteksi dan mendukung mitigasi serangan SQL Injection pada infrastruktur teknologi informasi. Penelitian dilakukan melalui pendekatan kuantitatif-eksperimental dengan simulasi serangan menggunakan Damn Vulnerable Web Application (DVWA) sebagai media uji dan SQLMap sebagai alat eksploitasi. Proses deteksi dan penanganan insiden dianalisis berdasarkan kerangka kerja NIST SP 800-61 Rev. 2 yang meliputi tahap persiapan, deteksi dan analisis, penanggulangan, pemulihan, serta aktivitas pasca-insiden. Hasil penelitian menunjukkan bahwa Security Onion, dengan dukungan komponen Suricata, Zeek, dan Elastic Stack, mampu mendeteksi berbagai skenario SQL Injection secara real-time dengan tingkat konsistensi yang tinggi dan false positive yang relatif rendah. Sistem ini juga mampu menyajikan informasi insiden secara terpusat melalui dashboard visual yang mendukung proses analisis dan pengambilan keputusan. Implementasi SIEM berbasis open source ini terbukti efektif dalam meningkatkan visibilitas keamanan serta memperkuat respons insiden terhadap serangan SQL Injection. Penelitian ini diharapkan dapat menjadi referensi praktis dan akademik dalam penerapan SIEM open source untuk pengamanan aplikasi web di lingkungan organisasi.

***Kata kunci**: Damn Vulnerable Web Application, NIST SP 800-61, Security Onion, SIEM, SQL Injection*

1. Latar Belakang

Perkembangan teknologi informasi yang semakin pesat telah mendorong organisasi untuk mengandalkan aplikasi web sebagai tulang punggung operasional bisnis, layanan publik, dan pengelolaan data. Digitalisasi ini membawa efisiensi dan fleksibilitas yang tinggi, namun sekaligus meningkatkan kompleksitas dan risiko keamanan siber. Aplikasi web yang terhubung langsung dengan basis data menjadi target utama serangan karena menyimpan informasi sensitif dan bersifat kritikal bagi keberlangsungan organisasi (Hughes, 2023).

Salah satu jenis serangan yang hingga saat ini masih mendominasi insiden keamanan aplikasi web adalah SQL Injection. Serangan ini memanfaatkan kelemahan dalam validasi input dan konstruksi kueri SQL, sehingga memungkinkan penyerang memperoleh akses tidak sah, memodifikasi data, bahkan mengambil alih sistem basis data secara keseluruhan (Kristian, 2024). Meskipun berbagai teknik pengamanan telah dikembangkan, SQL Injection tetap relevan karena banyak aplikasi web masih dikembangkan tanpa praktik keamanan yang memadai.

Ancaman SQL Injection tidak hanya berdampak pada aspek teknis, tetapi juga berimplikasi pada kerugian finansial, reputasi organisasi, serta kepercayaan pengguna. Kebocoran data akibat eksploitasi SQL Injection dapat memicu sanksi hukum dan gangguan operasional yang signifikan. Oleh karena itu, organisasi dituntut tidak hanya memiliki mekanisme pencegahan, tetapi juga kemampuan deteksi dini dan respons insiden yang efektif (Whitman & Mattord, 2021).

Dalam konteks keamanan siber modern, pendekatan berbasis *Security Information and Event Management* (SIEM) menjadi semakin penting. SIEM memungkinkan pengumpulan, korelasi, dan analisis log dari berbagai sumber secara terpusat dan real-time, sehingga mendukung proses deteksi ancaman yang lebih komprehensif dibandingkan

metode konvensional yang terpisah-pisah (Hughes, 2023). Implementasi SIEM juga menjadi fondasi utama dalam membangun security operation center (SOC) yang adaptif terhadap ancaman dinamis.

Seiring dengan meningkatnya kecanggihan serangan siber, teknologi SIEM juga mengalami perkembangan signifikan. *Platform* SIEM modern tidak hanya mengandalkan *signature-based detection*, tetapi juga memanfaatkan analisis perilaku, korelasi multi-event, serta integrasi dengan kerangka kerja keamanan global. Kondisi ini menuntut organisasi untuk mampu memanfaatkan SIEM secara optimal agar tidak tertinggal dalam menghadapi pola serangan yang semakin kompleks dan terdistribusi (Bennouk, 2024).

PT Neotech Cakrawala Indonesia adalah perusahaan *System Integrator* di bidang Solusi Keamanan Teknologi Informasi yang berfokus pada penyediaan produk, layanan, dan solusi TI, mencakup perangkat keras, perangkat lunak perlindungan data, pemantauan risiko. Sebagai penyedia solusi SIEM (*Security Information and Event Management*) komersial untuk klien, perusahaan ini juga memanfaatkan *platform* SIEM *Open Source Security Onion* di lingkungan internal. dan digunakan untuk melakukan simulasi serangan keamanan guna menguji efektivitasnya sebagai solusi alternatif yang dapat diterapkan pada perusahaan kecil hingga menengah yang membutuhkan solusi SIEM tanpa lisensi komersial. sekaligus mendukung pelaksanaan penelitian skripsi. Platform ini memungkinkan proses analisis dan deteksi serangan siber, seperti *SQL Injection* secara efisien melalui pemantauan aktivitas jaringan.

Berbagai penelitian terdahulu menunjukkan bahwa penerapan SIEM dapat meningkatkan visibilitas keamanan jaringan dan mempercepat proses identifikasi insiden. Studi yang dilakukan oleh Rahman (2024) menunjukkan bahwa integrasi IDS dengan SIEM mampu meningkatkan akurasi deteksi serangan berbasis jaringan. Penelitian lain menekankan bahwa penggunaan SIEM berbasis *open source* dapat menjadi alternatif efektif bagi organisasi dengan keterbatasan anggaran tanpa mengorbankan fungsi utama deteksi keamanan (Purbo, 2020).

Namun demikian, sebagian besar penelitian sebelumnya lebih berfokus pada aspek deteksi teknis dan belum secara mendalam mengaitkannya dengan proses penanganan insiden yang terstruktur. Banyak implementasi SIEM berhenti pada tahap menghasilkan alert, tanpa panduan yang jelas mengenai bagaimana *alert* tersebut dianalisis, diklasifikasikan, dan ditindaklanjuti secara sistematis. Hal ini berpotensi menimbulkan ketidakefisienan dan inkonsistensi dalam respons keamanan (Whitman & Mattord, 2021).

Selain itu, permasalahan *false positive* masih menjadi tantangan utama dalam penerapan SIEM. Jumlah alert yang tinggi sering kali membebani analis keamanan, sehingga meningkatkan risiko terlewatnya insiden yang sebenarnya kritis. Kondisi ini menunjukkan adanya kesenjangan antara kemampuan teknis SIEM dengan praktik operasional penanganan insiden di lapangan (Hughes, 2023).

Kerangka kerja NIST SP 800-61 Rev. 2 hadir sebagai standar internasional yang memberikan panduan sistematis dalam penanganan insiden keamanan informasi. Kerangka ini menekankan tahapan terstruktur mulai dari persiapan, deteksi dan analisis, penanggulangan, pemulihan, hingga aktivitas pasca-insiden. Integrasi SIEM dengan kerangka kerja NIST diyakini mampu meningkatkan efektivitas respons terhadap serangan siber, termasuk *SQL Injection* (Whitman & Mattord, 2021).

Di sisi lain, penggunaan open source SIEM seperti Security Onion menawarkan peluang besar bagi organisasi kecil dan menengah untuk mengadopsi sistem keamanan tingkat lanjut tanpa biaya lisensi yang tinggi. *Security Onion* mengintegrasikan berbagai komponen penting seperti *Suricata*, *Zeek*, dan *Elastic Stack*, yang memungkinkan analisis lalu lintas jaringan dan log secara mendalam. Namun, kajian empiris yang mengulas efektivitas *Security Onion* secara spesifik dalam mendeteksi dan mendukung mitigasi *SQL Injection* masih relatif terbatas.

Kesenjangan penelitian terlihat pada minimnya studi yang mengombinasikan simulasi serangan *SQL Injection*, pemanfaatan SIEM berbasis *open source*, dan penerapan kerangka kerja penanganan insiden secara terstandar dalam satu pendekatan yang utuh. Sebagian penelitian hanya menyoroti aspek teknis serangan atau konfigurasi alat, tanpa membahas keterkaitannya dengan alur respons insiden yang terintegrasi dan berorientasi praktik.

Berdasarkan kondisi tersebut, penelitian ini dilakukan untuk menjawab kebutuhan akan pendekatan yang lebih komprehensif dalam deteksi dan mitigasi serangan SQL Injection. Penelitian ini tidak hanya menilai kemampuan Security Onion dalam menghasilkan alert, tetapi juga menganalisis bagaimana sistem tersebut mendukung proses penanganan insiden berdasarkan NIST SP 800-61 Rev. 2 dalam lingkungan simulasi yang terkendali.

Dengan demikian, kebaruan penelitian ini terletak pada integrasi antara simulasi serangan *SQL Injection*, pemanfaatan SIEM *open source Security Onion*, serta analisis penanganan insiden berbasis standar NIST secara *end-to-end*. Penelitian ini bertujuan untuk menjawab pertanyaan utama mengenai sejauh mana *Security Onion*

DOI: https://doi.org/10.31004/riggs.v4i4.5773

efektif dalam mendeteksi serangan *SQL Injection* secara real-time dan bagaimana hasil deteksi tersebut dapat dimanfaatkan untuk mendukung proses mitigasi insiden keamanan aplikasi web secara sistematis dan terstandar.

Dari uraian latar belakang diatas, dalam upaya untuk meningkatkan keamanan aplikasi web, simulasi serangan siber menjadi metode penting untuk mengukur efektivitas sistem pertahanan. Salah satu ancaman yang masih sering terjadi adalah *SQL Injection*, yang dapat dimanfaatkan untuk mengakses data tanpa izin. PT. Neotech Cakrawala Indonesia menggunakan *Damn Vulnerable Web Application* (DVWA) sebagai media simulasi untuk menguji deteksi dan mitigasi serangan di jaringan internal.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan kuantitatif-eksperimental dengan metode simulasi serangan keamanan siber pada lingkungan terkontrol. Pendekatan ini dipilih untuk menguji secara langsung kemampuan sistem *Security Information and Event Management* (SIEM) berbasis *open source* dalam mendeteksi serangan SQL Injection serta mendukung proses penanganan insiden. Desain penelitian bersifat eksplanatif, dengan fokus pada hubungan antara aktivitas serangan, keluaran sistem SIEM, dan tahapan respons insiden berdasarkan kerangka kerja NIST SP 800-61 Rev. 2.

Lingkungan penelitian dibangun menggunakan arsitektur virtualisasi dengan tiga komponen utama, yaitu mesin penyerang, server aplikasi web, dan server SIEM. Server aplikasi menggunakan *Damn Vulnerable Web Application* (DVWA) yang dijalankan pada sistem operasi Linux dengan konfigurasi basis data MySQL. Server SIEM menggunakan *Security Onion* versi terbaru yang mengintegrasikan Suricata sebagai *intrusion detection system*, Zeek untuk analisis lalu lintas jaringan, dan *Elastic Stack* untuk visualisasi dan manajemen log. Seluruh sistem dijalankan pada jaringan lokal tertutup untuk menghindari gangguan eksternal.

Teknik pengumpulan data dilakukan melalui pencatatan log dan alert yang dihasilkan oleh *Security Onion* selama simulasi serangan berlangsung. Serangan *SQL Injection* dilakukan menggunakan alat SQLMap dengan skenario serangan dasar hingga tingkat lanjut, termasuk *boolean-based*, *error-based*, dan *time-based injection*. Setiap skenario serangan direplikasi sebanyak tiga kali untuk memastikan konsistensi hasil, dengan durasi pengujian rata-rata 15–20 menit per sesi serangan.

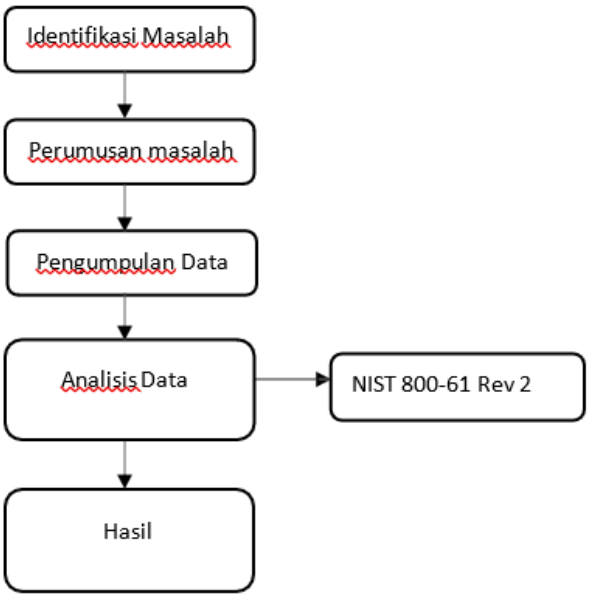
Variabel utama yang diamati meliputi waktu deteksi (*detection time*), jenis alert yang dihasilkan, tingkat keparahan insiden, serta kesesuaian alert dengan aktivitas serangan yang dilakukan. Data yang terkumpul dianalisis secara deskriptif dan komparatif untuk menilai kemampuan SIEM dalam mendeteksi pola *SQL Injection* secara real-time. Selain itu, dilakukan analisis terhadap potensi *false positive* dan *false negative* berdasarkan perbandingan antara aktivitas serangan aktual dan log sistem.

Proses penanganan insiden dianalisis dengan memetakan keluaran SIEM ke dalam tahapan NIST SP 800-61 Rev. 2, yaitu persiapan, deteksi dan analisis, penanggulangan, pemulihan, serta aktivitas pasca-insiden. Setiap *alert* yang relevan dievaluasi untuk menentukan apakah informasi yang disajikan oleh *Security Onion* cukup untuk mendukung pengambilan keputusan pada setiap tahap penanganan insiden. Pendekatan ini memungkinkan evaluasi tidak hanya pada aspek teknis deteksi, tetapi juga pada aspek operasional respons keamanan.

Untuk menjamin reprodusibilitas penelitian, seluruh konfigurasi sistem, aturan deteksi (*ruleset*), serta parameter serangan didokumentasikan secara rinci. Pengujian dilakukan dalam kondisi jaringan yang sama dengan beban lalu lintas normal yang dikendalikan, sehingga variasi hasil dapat diminimalkan. Metode ini diharapkan memungkinkan peneliti lain mereplikasi eksperimen dengan lingkungan serupa dan membandingkan hasilnya secara objektif.

Dalam konteks penanganan insiden keamanan informasi, perancangan sistem mengacu pada pedoman NIST SP 800-61 Rev. 2 yang terdiri dari tahapan preparation, detection and *analysis*, *containment*, *eradication and recovery*, serta *post-incident activity*. Masing-masing tahapan dirancang secara menyeluruh melalui penyusunan alur kerja (*workflow*), struktur organisasi tim respons insiden (CSIRT), serta perancangan antarmuka yang mendukung proses identifikasi, pelaporan, penanganan, hingga dokumentasi insiden. Dengan pendekatan ini, sistem diharapkan mampu merespons setiap insiden secara cepat, tepat, dan terdokumentasi dengan baik sesuai kebutuhan operasional pengguna. Tahapan penelitian mencangkup langkah – langkah pelaksanaan dari awal sampai akhir, adapun langkahnya sebagai berikut:

DOI: https://doi.org/10.31004/riggs.v4i4.5773



Gambar 1.
Tahapan Penelitian

3. Hasil dan Diskusi

Hasil penelitian menunjukkan bahwa sistem SIEM *Security Onion* mampu merekam seluruh aktivitas lalu lintas jaringan dan aplikasi selama simulasi serangan *SQL Injection* berlangsung. Seluruh paket data yang melewati jaringan tercatat secara konsisten oleh komponen Suricata dan Zeek, kemudian disimpan dan divisualisasikan melalui Elastic Stack. Kondisi ini menunjukkan bahwa arsitektur *Security Onion* telah terkonfigurasi dengan baik untuk melakukan pemantauan menyeluruh terhadap aktivitas jaringan dan aplikasi web secara *real-time*.

Pada kondisi awal tanpa serangan, lalu lintas jaringan yang tercatat relatif stabil dan didominasi oleh permintaan HTTP normal. Namun, ketika simulasi serangan *SQL Injection* mulai dijalankan, terjadi peningkatan signifikan pada volume trafik mencurigakan, khususnya pada parameter HTTP *request* yang mengarah langsung ke kueri basis data. Perbedaan pola lalu lintas ini menjadi indikator awal yang penting dalam proses deteksi insiden keamanan aplikasi web.

Dalam skenario serangan *boolean-based SQL Injection*, *Security Onion* mampu menghasilkan alert dalam rentang waktu 2–4 detik setelah payload dikirimkan ke server aplikasi. Alert yang dihasilkan diklasifikasikan sebagai aktivitas anomali dengan tingkat keparahan menengah hingga tinggi. Hal ini menunjukkan bahwa aturan deteksi berbasis pola yang digunakan oleh Suricata efektif dalam mengenali manipulasi logika SQL yang umum digunakan pada jenis serangan ini.

Selain itu, Zeek berperan penting dalam mendukung analisis serangan *boolean-based* dengan mencatat anomali pada struktur parameter HTTP request. Informasi yang dihasilkan oleh Zeek memungkinkan analis keamanan untuk mengidentifikasi pola komunikasi yang tidak wajar antara klien dan server, sehingga memperkuat proses korelasi log dan meningkatkan keakuratan deteksi insiden.

Hasil pengujian pada skenario *error-based SQL Injection* menunjukkan performa deteksi yang lebih cepat dibandingkan skenario lainnya. Rata-rata waktu deteksi berada pada rentang 1–3 detik, dengan alert yang dikategorikan memiliki tingkat keparahan tinggi. Hal ini disebabkan oleh munculnya pesan kesalahan basis data yang terekspos, sehingga memudahkan sistem SIEM dalam mengidentifikasi adanya upaya eksploitasi terhadap struktur kueri SQL.

Log yang dihasilkan pada skenario *error-based injection* memperlihatkan bahwa Suricata secara konsisten mendeteksi pola kueri berbahaya, sementara Elastic Stack menyajikan visualisasi insiden secara terpusat dan mudah dipahami. Kondisi ini memperlihatkan bahwa eksposur error basis data tidak hanya meningkatkan risiko keamanan, tetapi juga memperbesar peluang deteksi serangan oleh sistem SIEM.

DOI: https://doi.org/10.31004/riggs.v4i4.5773

Pada skenario *time-based SQL Injection*, sistem tetap mampu mendeteksi aktivitas serangan meskipun dengan waktu respons yang relatif lebih lambat. Rata-rata waktu deteksi tercatat antara 5–8 detik setelah payload dikirimkan. Keterlambatan ini disebabkan oleh karakteristik serangan yang memanfaatkan jeda waktu eksekusi kueri, sehingga tidak langsung memunculkan pola eksploitasi yang eksplisit pada paket data awal.

Dalam skenario ini, Zeek memainkan peran dominan dengan mengidentifikasi anomali durasi respons server yang signifikan dibandingkan kondisi normal. Analisis berbasis perilaku waktu ini melengkapi kemampuan Suricata yang lebih berfokus pada deteksi berbasis signature, sehingga memperlihatkan pentingnya kombinasi berbagai teknik analisis dalam sistem SIEM terintegrasi.

Replikasi pengujian yang dilakukan sebanyak tiga kali untuk setiap skenario serangan menunjukkan tingkat konsistensi hasil deteksi yang tinggi. Tidak ditemukan perbedaan signifikan pada jenis alert, waktu deteksi, maupun tingkat keparahan insiden antar replikasi. Konsistensi ini mengindikasikan stabilitas konfigurasi Security Onion serta reliabilitas lingkungan pengujian yang digunakan dalam penelitian.

Analisis terhadap *false positive* menunjukkan bahwa sebagian kecil alert muncul akibat lalu lintas HTTP normal, khususnya pada permintaan dengan parameter dinamis. Namun, jumlah *false positive* tersebut relatif rendah dan tidak mengganggu proses identifikasi insiden utama. Hal ini menunjukkan bahwa korelasi log dari berbagai sumber mampu meningkatkan akurasi deteksi dibandingkan penggunaan alat deteksi secara terpisah.

Secara keseluruhan, hasil sebelum penyajian Tabel 1 menunjukkan bahwa Security Onion mampu mendeteksi berbagai bentuk serangan SQL Injection secara real-time dengan tingkat konsistensi dan akurasi yang baik. Temuan ini menegaskan bahwa SIEM berbasis *open source* dapat diandalkan sebagai sistem pendukung deteksi dan respons awal terhadap serangan keamanan aplikasi web, khususnya ketika dikombinasikan dengan analisis lalu lintas jaringan dan aplikasi secara terintegrasi.

Tabel 1. Ringkasan Hasil Deteksi Serangan *SQL Injection* Menggunakan *Security Onion*

No	Jenis Serangan SQL Injection	Rata-rata Waktu Deteksi (detik)	Komponen Deteksi Utama	Tingkat Keparahan Alert	Konsistensi Replikasi	Keterangan
1	Boolean-based Injection	2–4	Suricata, Zeek	Menengah–Tinggi	Konsisten (3/3)	Pola manipulasi logika SQL terdeteksi pada parameter HTTP
2	Error-based Injection	1–3	Suricata	Tinggi	Konsisten (3/3)	Respons error basis data mempermudah identifikasi serangan
3	Time-based Injection	5–8	Zeek, Suricata	Menengah	Konsisten (3/3)	Anomali waktu respons server terdeteksi
4	Lalu lintas normal (kontrol)	–	Suricata, Zeek	Rendah	Konsisten	Beberapa *false positive* pada parameter dinamis
5	Keseluruhan skenario	–	SIEM Terintegrasi	–	Stabil	Tidak ditemukan *false negative*

Hasil penelitian ini menunjukkan bahwa *Security Onion* sebagai SIEM berbasis *open source* mampu mendeteksi berbagai bentuk serangan SQL Injection secara real-time dengan tingkat konsistensi yang tinggi. Kemampuan ini menegaskan bahwa integrasi beberapa komponen keamanan, seperti Suricata, Zeek, dan Elastic Stack, memberikan visibilitas yang lebih komprehensif terhadap aktivitas jaringan dan aplikasi. Pendekatan terintegrasi ini memungkinkan identifikasi pola serangan secara lebih akurat dibandingkan penggunaan sistem deteksi tunggal yang berdiri sendiri.

Deteksi yang cepat pada skenario *error-based SQL Injection* memperlihatkan bahwa eksposur pesan kesalahan basis data masih menjadi indikator utama dalam proses identifikasi serangan. Kondisi ini menunjukkan bahwa praktik pengelolaan error yang tidak aman tetap menjadi kelemahan serius pada banyak aplikasi web. Dari perspektif keamanan, temuan ini menegaskan pentingnya penerapan mekanisme *error handling* yang tepat untuk meminimalkan risiko eksploitasi sekaligus mengurangi peluang kebocoran informasi sensitif.

Pada skenario *boolean-based SQL Injection*, hasil deteksi yang stabil menunjukkan efektivitas aturan deteksi berbasis *signature* yang digunakan oleh Suricata. Namun, temuan ini juga mengindikasikan bahwa keberhasilan

DOI: https://doi.org/10.31004/riggs.v4i4.5773

deteksi sangat bergantung pada kualitas dan pembaruan aturan yang digunakan. Tanpa pemeliharaan dan pembaruan yang berkelanjutan, sistem berpotensi gagal mengenali variasi serangan baru yang terus berkembang seiring meningkatnya kompleksitas teknik eksploitasi SQL Injection.

Sementara itu, waktu deteksi yang relatif lebih lambat pada *time-based SQL Injection* mencerminkan tantangan dalam mengidentifikasi serangan berbasis anomali waktu. Serangan jenis ini tidak selalu menampilkan pola eksploitasi yang eksplisit pada paket data awal, sehingga membutuhkan analisis perilaku lalu lintas yang lebih mendalam. Peran Zeek dalam menganalisis durasi respons server menjadi faktor penting dalam melengkapi keterbatasan deteksi berbasis pola yang dimiliki Suricata.

Konsistensi hasil deteksi pada seluruh replikasi pengujian menunjukkan bahwa konfigurasi sistem Security Onion cukup stabil dan andal dalam lingkungan pengujian terkontrol. Stabilitas ini penting karena sistem deteksi yang tidak konsisten dapat menurunkan kepercayaan analis keamanan terhadap alert yang dihasilkan. Dengan tingkat konsistensi yang tinggi, sistem SIEM dapat menjadi fondasi yang kuat dalam mendukung operasional keamanan siber secara berkelanjutan.

Analisis terhadap *false positive* menunjukkan bahwa sebagian kecil alert dipicu oleh lalu lintas HTTP normal yang memiliki parameter dinamis dan pola permintaan yang menyerupai aktivitas anomali. Meskipun demikian, proporsi *false positive* yang relatif rendah mengindikasikan bahwa korelasi log dari berbagai sumber, seperti Suricata dan Zeek, mampu meningkatkan akurasi deteksi. Temuan ini menegaskan bahwa implementasi SIEM memerlukan proses *tuning* dan evaluasi berkelanjutan agar aturan deteksi dapat disesuaikan dengan karakteristik aplikasi, pola penggunaan, serta kondisi operasional organisasi yang dinamis.

Integrasi hasil deteksi dengan kerangka kerja NIST SP 800-61 Rev. 2 memperlihatkan bahwa *Security Onion* tidak hanya berfungsi sebagai alat pemantauan pasif, tetapi juga sebagai pendukung utama dalam proses penanganan insiden yang terstruktur. Informasi yang dihasilkan pada tahap deteksi dan analisis, seperti jenis serangan, waktu kejadian, dan sumber ancaman, terbukti cukup untuk mengarahkan pengambilan keputusan pada tahap penanggulangan dan pemulihan. Hal ini menunjukkan adanya keselarasan yang kuat antara kemampuan teknis SIEM dan praktik respons insiden berbasis standar internasional.

Dari sudut pandang praktis, temuan penelitian ini menunjukkan bahwa organisasi dengan keterbatasan sumber daya dapat memanfaatkan SIEM *open source* sebagai alternatif solusi keamanan yang efektif. Security Onion mampu menyediakan fungsi deteksi dan analisis insiden yang sebanding dengan solusi komersial tertentu, tanpa biaya lisensi yang tinggi. Hal ini menjadikan SIEM *open source* sebagai opsi strategis bagi organisasi kecil dan menengah dalam meningkatkan ketahanan keamanan siber.

Secara keseluruhan, hasil dan diskusi setelah Tabel 1 menegaskan bahwa penerapan Security Onion memberikan kontribusi signifikan dalam meningkatkan kesiapan deteksi dan respons terhadap serangan SQL Injection. Integrasi antara pengujian teknis, analisis SIEM, dan kerangka kerja penanganan insiden menghasilkan pendekatan yang lebih komprehensif dibandingkan penelitian sebelumnya. Temuan ini memperkuat posisi penelitian sebagai rujukan dalam pengembangan sistem keamanan aplikasi web berbasis SIEM *open source*.

Temuan penelitian ini memperlihatkan bahwa deteksi serangan SQL Injection tidak hanya bergantung pada kecepatan sistem SIEM dalam menghasilkan alert, tetapi juga pada kualitas informasi yang disajikan untuk dianalisis lebih lanjut. Security Onion mampu menyediakan konteks insiden yang cukup lengkap, termasuk sumber serangan, waktu kejadian, serta jenis aktivitas mencurigakan. Informasi kontekstual ini sangat penting untuk membantu analis keamanan memahami pola serangan dan menentukan prioritas penanganan secara tepat dan terukur.

Kemampuan *Security Onion* dalam menyajikan data insiden secara terpusat melalui *dashboard* visual *Elastic Stack* memberikan nilai tambah dalam proses analisis keamanan. Visualisasi *log* dan *alert* memudahkan analis dalam mengidentifikasi tren serangan, korelasi antar kejadian, serta eskalasi ancaman. Dengan pendekatan ini, proses analisis tidak hanya bersifat reaktif terhadap alert individual, tetapi juga proaktif dalam mengenali pola serangan yang berulang atau terstruktur.

Integrasi antara Suricata dan Zeek dalam lingkungan Security Onion menunjukkan sinergi yang efektif antara deteksi berbasis *signature* dan analisis perilaku jaringan. Suricata unggul dalam mengenali pola serangan yang telah diketahui, sementara Zeek mampu memberikan wawasan mendalam terhadap perilaku lalu lintas yang tidak wajar. Kombinasi ini memperkuat kemampuan SIEM dalam menghadapi variasi teknik *SQL Injection* yang tidak selalu memiliki ciri eksploitasi yang eksplisit.

Dari perspektif penanganan insiden, pemetaan hasil deteksi ke dalam tahapan NIST SP 800-61 Rev. 2 menunjukkan bahwa Security Onion mendukung proses deteksi dan analisis secara optimal. Informasi yang

DOI: https://doi.org/10.31004/riggs.v4i4.5773

dihasilkan memungkinkan identifikasi cepat terhadap insiden, klasifikasi tingkat keparahan, serta penentuan tindakan awal. Hal ini menegaskan bahwa SIEM berperan sebagai fondasi penting dalam membangun respons insiden yang terstruktur dan berbasis standar.

Pada tahap penanggulangan, data yang disediakan oleh *Security Onion* memungkinkan organisasi untuk melakukan isolasi sumber serangan secara lebih terarah. Informasi mengenai alamat IP penyerang, jenis *payload*, dan waktu kejadian dapat digunakan sebagai dasar dalam penerapan tindakan mitigasi awal, seperti pemblokiran akses atau penyesuaian aturan *firewall*. Dengan demikian, SIEM tidak hanya berfungsi sebagai alat deteksi, tetapi juga sebagai pendukung keputusan operasional keamanan.

Tahap pemulihan dalam kerangka NIST juga mendapat dukungan dari ketersediaan log dan rekaman insiden yang lengkap. Data historis yang tersimpan memungkinkan organisasi melakukan evaluasi dampak serangan terhadap sistem dan memastikan bahwa layanan kembali beroperasi secara normal. Selain itu, informasi ini dapat digunakan untuk memverifikasi efektivitas tindakan pemulihan yang telah dilakukan serta mencegah terulangnya insiden serupa di masa mendatang.

Aktivitas pasca-insiden merupakan aspek krusial yang sering kurang mendapatkan perhatian dalam praktik keamanan informasi. Hasil penelitian menunjukkan bahwa log dan laporan insiden yang dihasilkan oleh *Security Onion* dapat dimanfaatkan secara optimal untuk proses evaluasi menyeluruh dan pembelajaran organisasi. Informasi tersebut mendukung penyusunan rekomendasi perbaikan, pembaruan kebijakan keamanan, serta penyesuaian konfigurasi dan aturan deteksi. Dengan demikian, sistem keamanan dapat menjadi lebih adaptif, berkelanjutan, dan responsif terhadap perkembangan pola serangan siber yang semakin kompleks.

Meskipun hasil penelitian menunjukkan performa *Security Onion* yang baik dalam mendeteksi serangan *SQL Injection*, temuan ini juga mengindikasikan pentingnya pemeliharaan dan tuning sistem secara berkelanjutan. Lingkungan aplikasi web yang dinamis, ditandai oleh perubahan fitur, pola akses pengguna, serta konfigurasi infrastruktur, menuntut penyesuaian aturan deteksi agar tetap relevan dan tidak menghasilkan false positive berlebihan. Oleh karena itu, keberhasilan implementasi SIEM tidak hanya bergantung pada teknologi, tetapi sangat dipengaruhi oleh kompetensi sumber daya manusia dalam mengelola, menafsirkan, dan menganalisis data keamanan secara konsisten serta berkelanjutan dalam konteks operasional keamanan organisasi modern saat ini.

Dari sisi implikasi akademik, penelitian ini memperluas kajian mengenai penerapan SIEM *open source* dengan mengaitkannya secara langsung pada kerangka kerja penanganan insiden. Pendekatan ini memberikan sudut pandang yang lebih komprehensif dibandingkan penelitian sebelumnya yang cenderung berfokus pada aspek teknis deteksi semata. Dengan demikian, penelitian ini berkontribusi dalam menjembatani kesenjangan antara teori keamanan dan praktik operasional di lapangan.

Secara keseluruhan, pengembangan diskusi ini menegaskan bahwa *Security Onion* dapat berperan sebagai komponen strategis dalam sistem keamanan aplikasi web. Integrasi antara deteksi teknis, analisis perilaku, dan kerangka kerja NIST menghasilkan pendekatan *end-to-end* yang lebih matang dalam menghadapi serangan *SQL Injection*. Temuan ini memperkuat argumen bahwa SIEM *open source* dapat menjadi solusi yang efektif, terjangkau, dan berkelanjutan bagi organisasi dalam meningkatkan ketahanan keamanan siber.

4. Kesimpulan

Penelitian ini menyimpulkan bahwa implementasi *Security Information and Event Management* (SIEM) berbasis *open source* menggunakan *Security Onion* mampu mendeteksi serangan *SQL Injection* secara efektif dan konsisten dalam lingkungan pengujian terkontrol. Sistem ini berhasil mengidentifikasi berbagai skenario serangan, termasuk *boolean-based*, *error-based*, dan *time-based SQL Injection*, dengan waktu deteksi yang relatif cepat dan tingkat false positive yang rendah. Integrasi komponen *Suricata*, *Zeek*, dan *Elastic Stack* terbukti memberikan visibilitas keamanan yang memadai serta mendukung proses analisis insiden secara real-time. Selain itu, penelitian ini menegaskan bahwa pemanfaatan *Security Onion* yang dikaitkan dengan kerangka kerja NIST SP 800-61 Rev. 2 mampu memperkuat proses penanganan insiden keamanan aplikasi web secara sistematis. Kebaruan penelitian terletak pada pendekatan end-to-end yang mengombinasikan simulasi serangan, deteksi SIEM open source, dan analisis respons insiden berbasis standar internasional. Temuan ini memberikan implikasi praktis bahwa organisasi, khususnya dengan keterbatasan sumber daya, dapat mengadopsi solusi SIEM open source sebagai alternatif yang layak untuk meningkatkan kesiapan dan ketahanan keamanan siber.

DOI: https://doi.org/10.31004/riggs.v4i4.5773

Referensi

1. Adhiatma, N., 2020. *Master CCNA Belajar Network itu Mudah*. Jawa Barat: Nirifa Publisher.
2. Bennouk, A. (2024). Advanced cyber threat detection using behavioral analytics and anomaly-based methods. *Journal of Information Security and Applications*, 78, 1–12. https://doi.org/10.1016/j.jisa.2024.103682
3. Chrisantus Tristianto, S. M., 2022. *Mengenal Lebih Dekat Dengan VMWARE Workstation*. s.l.:Penerbit Adab.
4. Hughes, J. (2023*). SIEM and SOC modernization: Strategies for real-time threat detection*. Cybersecurity Press.
5. Husen, Z., 2020. *Membangun Server dan Jaringan Komputer Dengan Linux UBuntu*. Banda Aceh: Syiah Kuala University Press.
6. Kareem, F. Q., 2021. SQL Injection Attacks Prevention System. *Asian Journal of Research in Computer Science* .
7. Kristian, A. (2024). Analysis of SQL injection vulnerabilities in modern web applications. *International Journal of Computer Network and Information Security*, 16(2), 1–10. https://doi.org/10.5815/ijcnis.2024.02.01
8. Purbo, O. W. (2020). *Keamanan jaringan komputer dan internet*. Elex Media Komputindo.
9. Pratama, A. M., 2024. *Keamanan Data dan Informasi*. s.l.:Kaizen Media Publishing.
10. Rahman, A. (2024). Integrasi intrusion detection system dan SIEM dalam meningkatkan keamanan jaringan. *Jurnal Teknologi Informasi dan Keamanan Siber*, 5(1), 60–72.
11. Sastya Hendri Wibowo, S. M., 2023. *Jaringan Komputer & Komunikasi Data*. s.l.:Deepublish.
12. Uky Yudatama, 2024. *Memahami Teknologi Informasi*. Bandung: Kaizen Media Publishing.
13. Whitman, M. E., & Mattord, H. J. (2021). *Principles of information security (7th ed.)*. Cengage Learning.
14. National Institute of Standards and Technology. (2012). *Computer security incident handling guide (SP 800-61 Rev. 2)*. NIST. https://doi.org/10.6028/NIST.SP.800-61r2
15. OWASP Foundation. (2023). *OWASP top 10: The ten most critical web application security risks*. https://owasp.org/www-project-top-ten/

DOI: https://doi.org/10.31004/riggs.v4i4.5773