

Integrasi File Log dan TCPDump untuk Analisis Forensik Serangan DDoS pada Cloud

Hero Wintolo^{1*}, Muh Ikhvan Ade Rosadin², Sudaryanto³, Haruno Sajati⁴, Imam Riadi⁵,
Anton Yudhana⁶

^{1,2,3,4}Program Studi Informatika, Institut Teknologi Dirgantara Adisujtpto, Yogyakarta

⁵Program Studi Sistem Infromasi, Universitas Ahmad Dahlan, Yogyakarta

⁶Program Studi Teknik Elektro, Universitas Ahmad Dahlan, Yogyakarta

Email: herowintolo@itda.ac.id

Abstrak

Penelitian ini bertujuan mengembangkan analisis forensik jaringan berbasis integrasi *file log* server dan TCPDump untuk meningkatkan validitas serta keterlacakan bukti serangan *Distributed Denial of Service* (DDoS) pada lingkungan *private cloud*. Metode yang diterapkan adalah *Knowledge Understanding Assessment Defense* (KUAD) melalui tahapan *Initiation, Acquisition, Execution, Mitigation, dan Disposition*, dengan validasi lintas sumber berdasarkan alamat IP, waktu kejadian, tujuan akses, dan pola aktivitas. Lingkungan eksperimen menggunakan server Intel Core i5-4590 3,30 GHz dengan RAM 8 GB, Ubuntu Server 22.04, OwnCloud 10.15.2, MikroTik CCR1016-12G, Apache2, TCPDump 4.99.1, Python, MySQL, serta dua perangkat berbasis Kali Linux yang menjalankan GoldenEye untuk menghasilkan serangan DDoS berbasis HTTP. Hasil penelitian menunjukkan bahwa *file log* merekam 8.864 aktivitas, sedangkan TCPDump merekam 8.086 paket dari tiga alamat IP sumber. Validasi terhadap dataset menghasilkan 91% *agreement* berdasarkan kesesuaian atribut utama antar-sumber. Dua alamat IP menunjukkan pola distribusi yang relatif konsisten, sedangkan satu alamat IP memperlihatkan selisih pencatatan yang lebih besar akibat perbedaan karakteristik pencatatan pada lapisan aplikasi dan jaringan. Pemantauan sistem juga menunjukkan peningkatan beban CPU dan memori selama serangan. Secara umum, integrasi *file log* dan TCPDump dalam kerangka KUAD mampu memberikan bukti forensik yang saling menguatkan dan mendukung rekonstruksi serangan DDoS secara lebih komprehensif dibandingkan penggunaan satu sumber data.

Kata kunci— Analisa forensik; file log; TCP Dump; DDoS; Cloud Computing

1. PENDAHULUAN

Perkembangan *cloud computing* telah meningkatkan fleksibilitas dan skalabilitas layanan digital, tetapi pada saat yang sama memperluas permukaan serangan terhadap ketersediaan layanan. Salah satu ancaman yang paling signifikan adalah *Distributed Denial of Service* (DDoS), yang membanjiri sumber daya jaringan atau aplikasi dengan trafik dalam jumlah besar sehingga layanan tidak dapat diakses oleh pengguna yang sah. Skala ancaman tersebut terus meningkat. Laporan DDoS tahun 2025 mencatat sebanyak 47,1 juta serangan DDoS, meningkat sekitar 121% dibandingkan tahun sebelumnya, dengan rata-rata 5.376 serangan yang dimitigasi setiap jam. Dari jumlah tersebut, sebanyak 34,4 juta merupakan serangan pada lapisan jaringan, meningkat lebih dari tiga kali lipat dibandingkan 11,4 juta serangan pada tahun 2024. Selain peningkatan frekuensi, intensitas serangan juga mencapai skala hiper-volumetrik, dengan serangan terbesar

tercatat mencapai 31,4 Tbps, sedangkan kampanye DDoS berbasis HTTP mencapai lebih dari 200 juta request per second[1]. Kondisi tersebut menunjukkan bahwa serangan DDoS bukan hanya berpotensi memenuhi kapasitas jaringan, tetapi juga dapat mengganggu stabilitas infrastruktur dan ketersediaan layanan berbasis cloud.

Besarnya volume dan sifat terdistribusi dari serangan DDoS menimbulkan tantangan dalam proses investigasi karena bukti aktivitas serangan dapat tersebar pada beberapa lapisan sistem. Analisis yang hanya mengandalkan satu sumber data berpotensi menghasilkan informasi yang tidak lengkap. *File log* server, misalnya, merekam aktivitas pada lapisan aplikasi berupa alamat IP, waktu akses, metode HTTP, URL, dan status respons, sedangkan *packet capture* menyediakan informasi mengenai komunikasi yang benar-benar melintasi jaringan. Oleh karena itu, korelasi bukti dari beberapa sumber diperlukan untuk meningkatkan keterlacakan kejadian, memvalidasi konsistensi bukti, dan mengurangi kemungkinan kesalahan interpretasi dalam rekonstruksi serangan DDoS pada lingkungan cloud.

Secara konteks metodologis, beberapa kerangka telah digunakan dalam investigasi forensik digital, termasuk pendekatan National Institute of Standards and Technology (NIST) dan *Network Forensics Development Life Cycle* (NFDLC). NIST SP 800-86 menyediakan panduan umum untuk pengumpulan, pemeriksaan, analisis, dan pemanfaatan berbagai sumber data forensik, termasuk *files*, sistem operasi, trafik jaringan, dan aplikasi. Namun, NIST dirancang sebagai panduan praktik forensik yang fleksibel dalam konteks *incident response* dan bukan sebagai prosedur investigasi langkah demi langkah yang secara khusus mengintegrasikan tindakan pemulihan layanan[2]. Sementara itu, NFDLC menggunakan tahapan *Initiation*, *Acquisition*, *Implementation*, *Operation*, dan *Disposition* sehingga memberikan alur investigasi jaringan yang lebih terstruktur dan telah diterapkan pada berbagai kasus forensik jaringan[3]. Meskipun demikian, NFDLC terutama berorientasi pada identifikasi, akuisisi, analisis, dan pengamanan bukti setelah insiden, sedangkan mekanisme mitigasi dan pemulihan layanan belum ditempatkan sebagai bagian eksplisit dari siklus investigasinya.

Berdasarkan karakteristik tersebut, penelitian ini memilih metode *Knowledge Understanding Assessment Defense* (KUAD)[3]. KUAD dikembangkan dengan mempertahankan prinsip dasar investigasi NFDLC, tetapi memperluas siklus forensik melalui integrasi tahap *Initiation*, *Acquisition*, *Execution*, *Mitigation*, dan *Disposition*. Tahap *Execution* mengintegrasikan aktivitas implementasi dan operasi, sedangkan tahap *Mitigation* menghubungkan hasil identifikasi dan validasi bukti dengan tindakan pertahanan serta pemulihan kondisi layanan. Dengan demikian, keunggulan KUAD dalam konteks penelitian ini terletak pada kemampuannya membentuk closed-loop forensic-defense process, yaitu menghubungkan akuisisi bukti, validasi serangan, analisis, tindakan pertahanan, dan dokumentasi dalam satu siklus yang berkesinambungan[4]. Pemilihan KUAD tidak dimaksudkan untuk menyatakan bahwa metode ini secara universal lebih unggul daripada NIST atau NFDLC, karena perbandingan kinerja eksperimental langsung antar-framework masih diperlukan, tetapi KUAD lebih sesuai dengan tujuan penelitian ini, yang tidak hanya merekonstruksi aktivitas DDoS tetapi juga menghubungkan bukti forensik dengan evaluasi dampak dan proses pertahanan pada lingkungan cloud.

Selain aspek teknis, validitas dan integritas bukti merupakan komponen penting dalam forensik digital. Dalam konteks hukum Indonesia, Pasal 5 Undang-Undang Republik Indonesia Nomor 1 Tahun 2024 tentang Perubahan Kedua atas Undang-Undang Informasi dan Transaksi Elektronik menegaskan bahwa Informasi Elektronik, Dokumen Elektronik, dan/atau hasil cetaknya merupakan alat bukti hukum yang sah dan merupakan perluasan dari alat bukti yang diakui dalam hukum acara Indonesia[5]. Konsekuensinya, *file log* dan hasil *packet capture* yang digunakan dalam investigasi tidak hanya harus relevan terhadap kejadian serangan, tetapi juga harus dipertahankan integritas, autentisitas, dan keterlacakannya. Prinsip tersebut berkaitan dengan *chain of custody*, yaitu dokumentasi pergerakan bukti sejak proses pengumpulan, penyimpanan, analisis, hingga pemindahan bukti dengan mencatat pihak yang menangani bukti, waktu pengumpulan atau pemindahan, dan tujuan setiap proses tersebut[6]. Dalam penelitian forensik jaringan, penerapan prinsip ini diperlukan agar bukti dari *file log* dan TCPDump dapat

ditelusuri asal-usulnya serta dapat diverifikasi bahwa data yang dianalisis tetap konsisten dengan data yang diperoleh pada saat akuisisi.

Permasalahan utama dalam penelitian ini adalah sulitnya mengidentifikasi serangan *Distributed Denial of Service* (DDoS) secara akurat pada lingkungan *cloud computing*[7] akibat keterbatasan analisis yang hanya menggunakan satu sumber data. Selain itu, tingginya intensitas trafik serangan menyebabkan gangguan pada performa sistem, seperti peningkatan penggunaan CPU dan memori yang berdampak pada kualitas layanan. Sebagian besar penelitian analisa forensik masih berfokus pada tahap akuisisi dan pengumpulan bukti tanpa mengoptimalkan proses analisis secara menyeluruh. Pendekatan seperti *Network Forensic Development Life Cycle* (NFDLC)[8] dan National Institute of Standards and Technology (NIST)[9] telah digunakan dalam analisa forensik, namun implementasinya cenderung terbatas pada simulasi sederhana seperti email spamming. Selain itu, analisa forensik pada sistem berbasis blockchain masih berorientasi pada pengujian serangan tanpa mengintegrasikan mekanisme investigasi yang komprehensif[10]. Di sisi lain, tantangan besar dalam analisa forensik muncul dari heterogenitas perangkat dan kurangnya standarisasi log yang menghambat proses investigasi[11]. Analisa forensik juga menghadapi kendala pada aspek integrasi framework yang masih terfragmentasi antar domain[12]. Keterbatasan lainnya adalah minimnya fokus pada validasi bukti dan aspek legalitas data digital. Pengembangan model analisa forensik yang lebih terintegrasi, adaptif, dan mampu menangani kompleksitas lingkungan digital modern[13].

Analisa forensik dengan memanfaatkan penggunaan file log yang telah dilakukan penelitian sebelumnya menunjukkan adanya perbedaan signifikan dalam metode pengolahan dan tujuan analisis data[14]. Beberapa penelitian menekankan deteksi anomali secara real time, sementara yang lain lebih fokus pada rekonstruksi forensik dan validasi kualitas sistem[15]. Selain itu, variasi sumber file log dan frekuensi data turut memengaruhi akurasi hasil analisis yang dilaporkan dalam masing masing studi[16]. Di sisi lain, integrasi metode berbasis machine learning masih belum merata dan seringkali terbatas pada konteks tertentu saja. Hal ini menunjukkan adanya gap dalam standarisasi pendekatan analisis file log lintas domain aplikasi[17]. Selain itu, keterbatasan dalam pemanfaatan data historis dan kurangnya interoperabilitas sistem juga menjadi faktor yang memperlebar kesenjangan penelitian. Sehingga diperlukan pengembangan framework terpadu untuk meningkatkan konsistensi dan efektivitas[18].

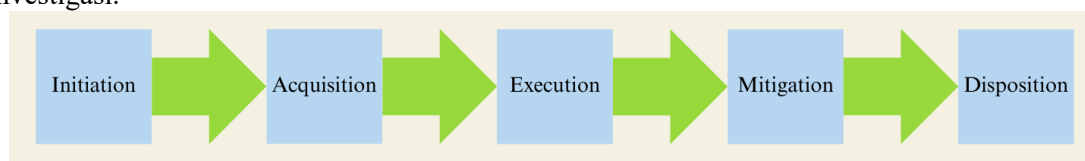
Selain menggunakan file log, penelitian ini juga memanfaatkan TCPDump sebagai tool untuk akuisisi data serangan DDoS. TCPDump secara umum digunakan sebagai alat penangkap paket untuk analisis lalu lintas jaringan dan pembentukan dataset. Namun, terdapat gap utama berupa keterbatasan TCPDump dalam menampilkan seluruh field paket secara lengkap sehingga membutuhkan tools tambahan seperti Wireshark untuk interpretasi lebih lanjut[19]. Selain itu, TCPDump hanya berfungsi sebagai packet capturer tanpa kemampuan klasifikasi atau deteksi otomatis terhadap ancaman jaringan[20]. Penelitian lain juga menunjukkan bahwa TCPDump sering digunakan sebagai sumber data mentah, tetapi tidak menyediakan fitur ekstraksi fitur lanjutan yang dibutuhkan dalam sistem berbasis machine learning[21]. Keterbatasan ini menyebabkan ketergantungan pada tools tambahan dalam proses analisis. Selain itu, penggunaan berbasis command line menjadi hambatan bagi pengguna non-teknis[22]. Dengan demikian, diperlukan integrasi TCPDump dengan sistem analitik yang lebih cerdas[23].

Berdasarkan permasalahan dan analisa gap yang telah disampaikan diatas, penelitian ini memberikan solusi pada penerapan analisis forensik jaringan berbasis integrasi antara *file log server* dan *TCPDump* untuk memperoleh data yang lebih komprehensif dan akurat. Pendekatan ini diperkuat dengan metode KUAD[24] yang mencakup tahapan koleksi, validasi, analisis, dan dokumentasi sehingga proses identifikasi serangan DDoS dapat dilakukan secara sistematis. Penelitian ini juga bertujuan untuk mengintegrasikan data dari file log server dan TCPDump guna meningkatkan validitas serta keandalan hasil analisis. Selain itu, penelitian ini bertujuan

memahami pola serangan dan dampaknya terhadap performa sistem secara komprehensif. Kebaharuan penelitian ini terletak pada integrasi analisis forensik jaringan yang menggabungkan file log server dan TCPDump dalam satu kerangka kerja terpadu. Berbeda dengan penelitian sebelumnya yang umumnya hanya menggunakan satu sumber data, pendekatan ini memanfaatkan multi-layer acquisition untuk meningkatkan akurasi analisis. Penelitian ini juga mengadopsi metode KUAD yang disesuaikan dengan kebutuhan investigasi serangan DDoS pada lingkungan cloud computing. Selain itu, proses validasi data dilakukan secara langsung dengan membandingkan dua sumber data sehingga menghasilkan tingkat keandalan yang tinggi. Kebaharuan lainnya adalah penggunaan visualisasi data untuk mengidentifikasi pola serangan secara lebih jelas dan terukur. Penelitian ini juga mampu menunjukkan hubungan antara pola serangan dan dampaknya terhadap performa sistem secara nyata. Dengan demikian, pendekatan yang diusulkan memberikan kontribusi signifikan dalam meningkatkan efektivitas analisis forensik jaringan dibandingkan metode sebelumnya.

2. METODE PENELITIAN

Metode *Knowledge Understanding Assessment Defense* (KUAD) dalam penelitian ini diadaptasi dan diselaraskan dengan tahapan proses forensik digital yang terdiri dari *Initiation*, *Acquisition*, *Execution*, *Mitigation*, dan *Disposition* yang dapat dilihat pada Gambar 1. Pada tahap *initiation*, dilakukan identifikasi awal terhadap indikasi serangan DDoS serta penentuan ruang lingkup investigasi pada lingkungan *cloud computing*. Selanjutnya, tahap *acquisition* berfokus pada proses pengumpulan data dari berbagai sumber, yaitu *file log server* dan hasil tangkapan paket jaringan menggunakan *TCPDump*, yang masing-masing merepresentasikan lapisan aplikasi dan jaringan. Sebagai upaya untuk menjaga integritas dan keterlacakan bukti digital, proses akuisisi *file log* dan hasil tangkapan *TCPDump* dilengkapi dengan prosedur *chain of custody*. Setiap artefak bukti diberi identitas unik dan dicatat informasi mengenai sumber bukti, waktu akuisisi, perangkat atau *tool* yang digunakan, serta pihak yang melakukan proses akuisisi. Nilai *hash* SHA-256 dihitung terhadap file bukti setelah proses akuisisi untuk menghasilkan nilai referensi integritas. Bukti asli disimpan sebagai *master evidence*, sedangkan proses analisis dilakukan terhadap salinan kerja (*working copy*) untuk meminimalkan perubahan pada bukti asli. Nilai *hash* diverifikasi kembali sebelum dan setelah proses analisis, sedangkan setiap perpindahan atau penggunaan bukti didokumentasikan dalam catatan *chain of custody*. Prosedur tersebut digunakan untuk memastikan bahwa *file log* dan *packet capture* yang dianalisis dapat ditelusuri asal-usulnya serta tidak mengalami perubahan yang tidak terdokumentasi selama proses investigasi.



Gambar 1. Metode Penelitian yang digunakan

Ada beberapa tahapan yang diadopsi dari Metode KUAD yang digunakan dalam penelitian ini. Tahap pertama adalah koleksi data, yang dilakukan dengan mengumpulkan data dari dua sumber utama, yaitu *file log server* dan hasil tangkapan paket jaringan menggunakan *TCPDump*. *File log* menyediakan informasi aktivitas pada lapisan aplikasi seperti alamat IP, waktu akses, metode HTTP, dan URL yang diakses, sedangkan *TCPDump* menangkap paket data pada lapisan jaringan secara *real-time*. Tahap kedua adalah uji (validasi data), di mana data dari kedua sumber dibandingkan untuk memastikan konsistensi dan keakuratan informasi. Proses ini melibatkan pencocokan parameter seperti alamat IP sumber, waktu kejadian, dan tujuan akses. Hasil uji menunjukkan bahwa seluruh data yang diperoleh memiliki tingkat kesesuaian yang tinggi,

sehingga dapat digunakan sebagai dasar analisis yang valid. Tahap ketiga adalah analisis, yang bertujuan untuk mengidentifikasi pola serangan DDoS. Analisis dilakukan dengan melihat frekuensi akses, distribusi alamat IP, serta pola waktu permintaan. Ditemukan adanya pola akses berulang dalam interval waktu yang sangat singkat dari beberapa alamat IP menuju satu target, yang merupakan karakteristik utama serangan DDoS. Selain itu, analisis juga mencakup dampak terhadap performa sistem, seperti peningkatan penggunaan CPU dan memori. Tahap terakhir adalah dokumentasi, yaitu proses penyajian hasil analisis dalam bentuk tabel, grafik, dan visualisasi data untuk mempermudah interpretasi. Dokumentasi ini menjadi bukti forensik yang kuat dalam mengidentifikasi dan memahami karakteristik serangan. Untuk menjamin keterulangan (*reproducibility*) eksperimen, lingkungan pengujian dibangun menggunakan konfigurasi perangkat keras dan perangkat lunak yang ditetapkan secara spesifik. Infrastruktur penelitian terdiri atas satu server sebagai host layanan *private cloud* OwnCloud, router sebagai pengatur lalu lintas jaringan, perangkat penyerang untuk menghasilkan trafik DDoS, serta perangkat monitoring untuk mendukung proses akuisisi dan analisis forensik. Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam eksperimen disajikan pada Tabel 1.

Tabel 1. Spesifikasi Perangkat Keras dan Perangkat Lunak Lingkungan Eksperimen

No.	Kategori	Komponen	Spesifikasi/Versi	Jumlah	Fungsi dalam Penelitian
1	Hardware	Server <i>private cloud</i>	Intel(R) Core(TM) i5-4590 @ 3.30 GHz; RAM VGEN RESCUE DDR3 8 GB; motherboard MB-QWERTY H81 NVMe	1	Menjalankan Ubuntu Server, Apache2, OwnCloud, MySQL, <i>access log</i> , dan TCPDump sebagai objek serangan DDoS
2	Hardware	Media penyimpanan server	Hard disk; kapasitas dan model perlu dilengkapi	1	Menyimpan sistem operasi, OwnCloud, database, <i>access.log</i> , dan data hasil akuisisi
3	Hardware	Router	MikroTik Ethernet Router CCR1016-12G	1	Mengatur komunikasi dan lalu lintas jaringan antara perangkat penyerang dan server
4	Hardware	Laptop penyerang	2 unit laptop; model, CPU, RAM, dan kapasitas penyimpanan perlu dilengkapi	2	Menjalankan Kali Linux dan GoldenEye secara bersamaan untuk menghasilkan trafik DDoS
5	Hardware	Laptop monitoring/remote access	ASUS VivoBook X513UA; CPU dan RAM perlu dilengkapi jika tersedia	1	Mengakses server menggunakan PuTTY dan melakukan pemantauan proses akuisisi
6	Software	Sistem operasi server	Ubuntu Server 22.04	1	Sistem operasi server <i>private cloud</i>
7	Software	Sistem operasi penyerang	Kali Linux; versi perlu dilengkapi	2	Lingkungan eksekusi GoldenEye pada perangkat penyerang
8	Software	OwnCloud	Version 10.15.2	1	Layanan <i>private cloud</i> yang menjadi target serangan DDoS

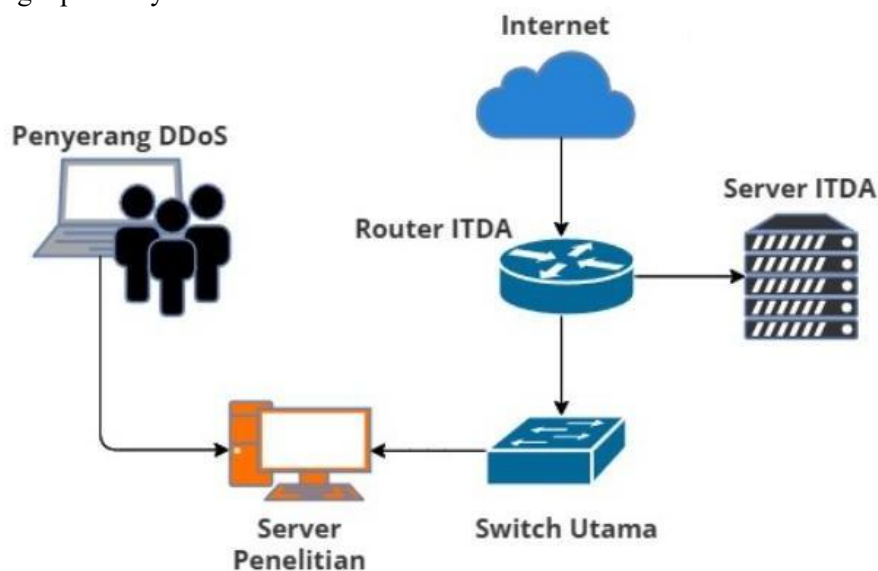
No.	Kategori	Komponen	Spesifikasi/Versi	Jumlah	Fungsi dalam Penelitian
9	Software	Apache HTTP Server	Apache2; versi perlu dilengkapi	1	Menyediakan layanan web OwnCloud dan menghasilkan <i>access log</i>
10	Software/Data Source	Apache Access Log	/var/log/apache2/access.log	1	Sumber bukti pada <i>application layer</i> , berisi IP, waktu, metode HTTP, URL, status respons, ukuran data, <i>referer</i> , dan <i>user-agent</i>
11	Software	TCPDump	Version 4.99.1; libpcap 1.10.1 with TPACKET_V3; OpenSSL 3.0.2	1	Menangkap paket jaringan secara <i>real-time</i> sebagai sumber bukti pada <i>network layer</i>
12	Software	GoldenEye	GoldenEye DDoS tool; versi/commit perlu dilengkapi	2	Menghasilkan serangan HTTP-based DDoS terhadap layanan OwnCloud
13	Software	Python	Python 3; versi minor perlu dilengkapi	1	Menjalankan program integrasi hasil TCPDump agar data tangkapan otomatis dimasukkan ke database
14	Software	MySQL	MySQL Server; versi perlu dilengkapi	1	Menyimpan hasil tangkapan TCPDump dan data pendukung analisis
15	Software	Webmin	Webmin; versi perlu dilengkapi	1	Mengelola server secara <i>remote</i> dan mengunduh file access.log untuk analisis
16	Software	PuTTY	PuTTY; versi perlu dilengkapi	1	Melakukan akses terminal <i>remote</i> ke server
17	Software	htop	htop; versi perlu dilengkapi	1	Memantau penggunaan CPU, memori, dan proses server sebelum dan selama serangan
18	Software	Aplikasi analisis/visualisasi	Aplikasi berbasis web yang mengintegrasikan <i>file log</i> dan TCPDump	1	Melakukan akuisisi, validasi, pencarian, dan visualisasi hasil analisis serangan

Berdasarkan Tabel 1, layanan OwnCloud versi 10.15.2 dijalankan pada Ubuntu Server 22.04 dengan prosesor Intel Core i5-4590 3.30 GHz dan RAM DDR3 8 GB. Server terhubung melalui MikroTik CCR1016-12G, sedangkan simulasi DDoS dilakukan secara simultan menggunakan dua perangkat berbasis Kali Linux yang menjalankan GoldenEye. Aktivitas pada lapisan aplikasi direkam melalui Apache access.log, sedangkan aktivitas pada lapisan jaringan diakuisisi

menggunakan TCPDump 4.99.1. Hasil tangkapan TCPDump selanjutnya diproses menggunakan program Python dan disimpan ke MySQL untuk mendukung proses pencocokan dan validasi dengan data *access log*. Konfigurasi tersebut digunakan secara konsisten pada seluruh tahapan eksperimen agar proses pengujian dapat direplikasi pada lingkungan yang memiliki karakteristik perangkat dan perangkat lunak yang setara.

3. HASIL DAN PEMBAHASAN

Hasil implementasi menunjukkan bahwa sistem yang dibangun mampu melakukan proses akuisisi, validasi, serta visualisasi data serangan secara terstruktur. Data diperoleh dari dua sumber utama, yaitu file log server dan hasil tangkapan jaringan menggunakan TCPDump. Integrasi kedua sumber data ini memungkinkan proses analisis dilakukan secara komprehensif, sehingga aktivitas serangan dapat diidentifikasi dengan tingkat akurasi yang tinggi. Gambar 2 berikut menunjukkan arsitektur jaringan pada lingkungan cloud computing yang digunakan dalam penelitian, di mana alur komunikasi dimulai dari pengguna yang mengakses layanan melalui jaringan, kemudian diteruskan ke router utama (Router ITDA) sebelum mencapai server (Server ITDA) sebagai pusat layanan.

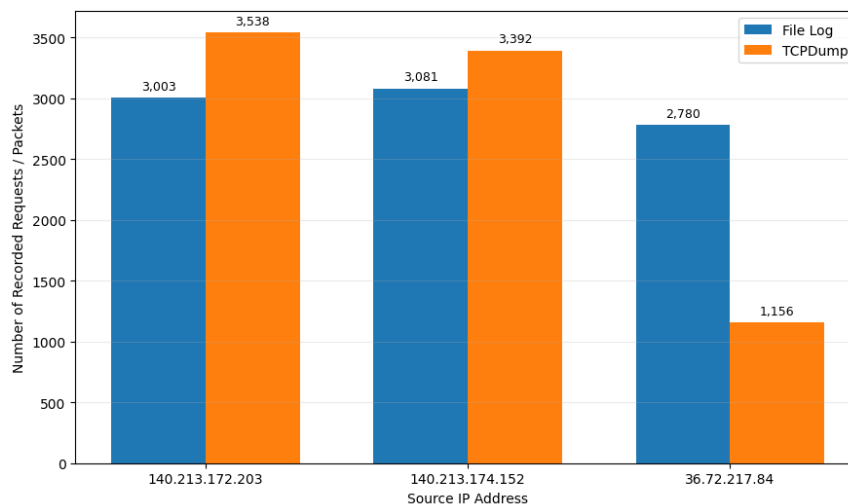


Gambar 2. Susunan peralatan yang digunakan dalam penelitian

Berdasarkan Gambar 2 tersebut, terlihat bahwa router berperan sebagai penghubung utama yang mengatur lalu lintas data antara pengguna dan server, sehingga seluruh aktivitas jaringan, termasuk potensi serangan DDoS, akan melewati titik ini dan dapat dianalisis untuk keperluan forensik jaringan. Pada tahap akuisisi, data serangan berhasil dikumpulkan dari file access log yang mencatat seluruh aktivitas permintaan HTTP ke server. Setiap entri log memuat informasi penting seperti alamat IP, waktu akses, metode HTTP, serta URL yang diakses. Selain itu, TCPDump digunakan untuk menangkap paket data secara langsung pada lapisan jaringan. Hasil penelitian menunjukkan bahwa kedua sumber data tersebut saling melengkapi dalam merekam aktivitas jaringan. Data log memberikan gambaran aktivitas pada level aplikasi, sedangkan TCPDump memberikan bukti pada level jaringan. Hal ini menjadi dasar penting dalam proses validasi serangan. File log merekam aktivitas pada lapisan aplikasi, sedangkan TCPDump menangkap paket pada lapisan jaringan.

Gambaran kuantitatif mengenai konsistensi data hasil akuisisi, dilakukan perbandingan jumlah aktivitas yang tercatat pada *file log* dan TCPDump berdasarkan alamat IP sumber. Perbandingan ini digunakan untuk melihat distribusi trafik yang direkam pada lapisan aplikasi

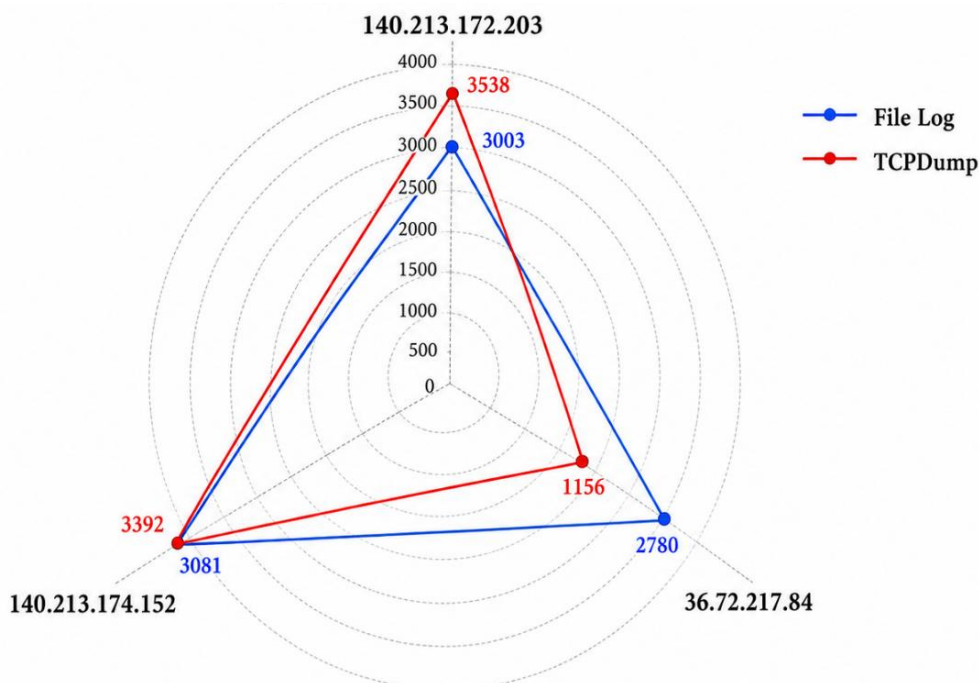
dan lapisan jaringan. Hasil menunjukkan bahwa alamat IP 140.213.172.203 tercatat sebanyak 3.003 aktivitas pada *file log* dan 3.538 paket pada TCPDump, sedangkan alamat IP 140.213.174.152 tercatat sebanyak 3.081 aktivitas pada *file log* dan 3.392 paket pada TCPDump. Sementara itu, alamat IP 36.72.217.84 menunjukkan perbedaan yang lebih besar, yaitu 2.780 aktivitas pada *file log* dan 1.156 paket pada TCPDump. Perbandingan jumlah data dari kedua sumber tersebut disajikan pada Gambar 3.



Gambar 3. Data set hasil penelitian dari TCP dump dan File Log

Berdasarkan Gambar 3, dua alamat IP, yaitu 140.213.172.203 dan 140.213.174.152, menunjukkan pola jumlah data yang relatif sebanding antara *file log* dan TCPDump, meskipun jumlah paket yang direkam TCPDump lebih tinggi. Perbedaan tersebut dapat terjadi karena kedua sumber bekerja pada lapisan yang berbeda, di mana TCPDump merekam paket yang melintasi antarmuka jaringan, sedangkan *file log* hanya mencatat permintaan yang telah diproses oleh layanan web. Perbedaan yang lebih besar terlihat pada alamat IP 36.72.217.84, sehingga data tersebut memerlukan validasi lebih lanjut berdasarkan parameter lain, seperti *timestamp*, alamat tujuan, metode HTTP, dan pola permintaan. Oleh karena itu, kesesuaian bukti forensik tidak hanya ditentukan berdasarkan jumlah data, tetapi melalui korelasi beberapa atribut dari *file log* dan TCPDump agar identifikasi aktivitas serangan DDoS lebih reliabel.

Tahap validasi menunjukkan hasil yang signifikan, di mana seluruh data yang dibandingkan antara file log dan TCPDump dinyatakan valid. Kesesuaian data meliputi alamat IP sumber, waktu kejadian, dan tujuan akses. Hasil ini menunjukkan bahwa aktivitas yang tercatat pada server benar-benar merupakan kejadian nyata di jaringan, bukan kesalahan pencatatan sistem. Selain itu, ditemukan pola akses berulang dari beberapa alamat IP dalam waktu yang berdekatan, yang mengindikasikan adanya serangan DDoS secara terkoordinasi. Pola ini menjadi indikator kuat dalam mengidentifikasi sumber serangan serta intensitas trafik yang dihasilkan. Gambaran visual mengenai proses validasi data hasil tangkapan TCPDump menggunakan *file log*, dilakukan perbandingan jumlah aktivitas yang terekam dari kedua sumber data berdasarkan alamat IP sumber. Visualisasi ini bertujuan untuk menunjukkan tingkat kesesuaian pola data antara *file log* sebagai bukti pada lapisan aplikasi dan TCPDump sebagai bukti pada lapisan jaringan. Dengan menggunakan grafik labu-laba, perbedaan dan kemiripan jumlah aktivitas dari setiap alamat IP dapat diamati secara lebih jelas. Hasil perbandingan tersebut disajikan pada Gambar 4.



Gambar 4. Validasi data TCP dump menggunakan File Log

Berdasarkan Gambar 4, terlihat bahwa alamat IP 140.213.172.203 dan 140.213.174.152 menunjukkan pola jumlah data yang relatif sebanding antara *file log* dan TCPDump, meskipun nilai pada TCPDump cenderung lebih tinggi. Kondisi ini menunjukkan bahwa kedua sumber data memiliki konsistensi yang baik dalam merekam aktivitas jaringan yang terindikasi sebagai serangan. Sementara itu, alamat IP 36.72.217.84 memperlihatkan selisih yang lebih besar antara kedua sumber data, yang mengindikasikan perlunya pemeriksaan lebih lanjut melalui pencocokan parameter lain seperti *timestamp*, tujuan akses, dan pola permintaan. Secara keseluruhan, grafik laba-laba ini memperkuat bahwa validasi menggunakan *file log* dan TCPDump mampu memberikan gambaran yang lebih komprehensif dalam mengidentifikasi dan memverifikasi aktivitas serangan DDoS.

Dari sisi analisis dampak, hasil pemantauan menunjukkan adanya peningkatan signifikan pada penggunaan sumber daya server saat serangan berlangsung. Pengamatan menggunakan tool monitoring memperlihatkan lonjakan penggunaan CPU dan memori yang cukup tinggi dibandingkan kondisi normal. Kondisi ini menunjukkan bahwa serangan DDoS berhasil membebani sistem dengan permintaan dalam jumlah besar secara bersamaan. Akibatnya, performa layanan mengalami penurunan dan berpotensi menyebabkan gangguan akses bagi pengguna yang sah. Hal ini menegaskan bahwa serangan DDoS tidak hanya berdampak pada jaringan, tetapi juga pada stabilitas sistem secara keseluruhan.

Analisis kuantitatif terhadap kedua sumber bukti menunjukkan bahwa *file log* merekam total 8.864 aktivitas dari tiga alamat IP yang dianalisis, sedangkan TCPDump merekam 8.086 paket. Dengan demikian, jumlah data TCPDump setara dengan sekitar 91,22% dari jumlah entri *file log*, dengan selisih keseluruhan sebanyak 778 rekaman atau sekitar 8,78%. Pada tingkat alamat IP, 140.213.172.203 menghasilkan 3.003 entri pada *file log* dan 3.538 paket pada TCPDump, sedangkan 140.213.174.152 menghasilkan 3.081 entri pada *file log* dan 3.392 paket pada TCPDump. Kedua alamat tersebut menunjukkan pola distribusi yang relatif berdekatan meskipun TCPDump mencatat jumlah yang lebih tinggi masing-masing sebesar 17,82% dan 10,09% dibandingkan *file log*. Sebaliknya, alamat IP 36.72.217.84 menunjukkan perbedaan yang lebih besar, yaitu 2.780 entri pada *file log* dibandingkan 1.156 paket pada TCPDump.

Perbedaan jumlah tersebut tidak secara langsung menunjukkan inkonsistensi bukti karena kedua sumber data bekerja pada lapisan dan mekanisme pencatatan yang berbeda. TCPDump merekam paket yang melewati antarmuka jaringan, sedangkan *Apache access log* hanya menghasilkan entri setelah sebuah permintaan diterima dan diproses oleh layanan web. Satu permintaan HTTP dapat melibatkan lebih dari satu paket jaringan, sementara paket tertentu juga dapat tidak menghasilkan entri pada *access log* apabila koneksi dihentikan sebelum permintaan diproses oleh aplikasi. Oleh karena itu, kesamaan jumlah absolut antara TCPDump dan *file log* bukan merupakan prasyarat validitas. Validasi yang lebih tepat dilakukan melalui korelasi atribut, terutama alamat IP sumber, *timestamp*, alamat tujuan, metode atau pola permintaan, serta urutan kejadian.

Atas dasar tersebut, hasil validasi yang memiliki kesesuaian antara *file log* dan TCPDump perlu diinterpretasikan sebagai 91% agreement pada bukti yang diuji, bukan sebagai tingkat akurasi 100% terhadap seluruh dataset. Pada dataset tersebut, kecocokan alamat IP sumber, waktu kejadian, dan alamat tujuan menunjukkan bahwa aktivitas yang tercatat pada lapisan aplikasi mempunyai bukti korespondensi pada lapisan jaringan. Pendekatan ini memperkuat proses rekonstruksi kejadian karena suatu aktivitas tidak hanya disimpulkan berdasarkan satu artefak, melainkan dikonfirmasi melalui dua sumber bukti independen yang merepresentasikan lapisan sistem yang berbeda. Dengan demikian, kontribusi utama proses validasi bukan terletak pada kesamaan jumlah data, tetapi pada kemampuan melakukan *cross-source corroboration* terhadap bukti serangan.

Temuan ini menunjukkan perbedaan penting dibandingkan penelitian forensik DDoS sebelumnya yang menggunakan IDS Snort dan Wireshark. Pada penelitian tersebut, Snort berfungsi mendeteksi DDoS dan menghasilkan log serangan yang kemudian dianalisis menggunakan Wireshark, sehingga bukti forensik sangat bergantung pada hasil deteksi dan aturan IDS. Penelitian tersebut berhasil menemukan tiga alamat IP penyerang dan menunjukkan peningkatan penggunaan CPU dari 22,6% menjadi 67,0% serta memori dari 55,7% menjadi 61,0%. Pada penelitian ini, bukti tidak hanya berasal dari keluaran sistem deteksi, tetapi dikorelasikan langsung antara *access log* layanan OwnCloud dan paket jaringan yang ditangkap TCPDump. Perbedaan tersebut memberikan keuntungan berupa kemampuan untuk memverifikasi apakah suatu aktivitas yang tercatat oleh aplikasi benar-benar memiliki jejak komunikasi pada jaringan.

Pola serupa juga terlihat pada penelitian NFDLC untuk kasus *e-mail spamming*, phishing, fake Wi-Fi, dan botnet. Penelitian *e-mail spamming* memanfaatkan Wireshark untuk memperoleh alamat IP pelaku dan korban serta menerapkan *chain of custody* pada tahap *disposition*. Penelitian phishing menggabungkan analisis *e-mail header*, Wireshark, dan NetworkMiner untuk memperoleh artefak seperti alamat IP, alamat pengirim, dan URL phishing. Pada kasus fake Wi-Fi, Wireshark dan NetworkMiner digunakan untuk menemukan artefak HTTP, ARP, serta file HTML dan CSS yang mencurigakan. Sementara itu, penelitian botnet menggunakan Wazuh untuk melakukan pemantauan serangan secara *real-time* dan mengelompokkan berbagai jenis serta tingkat serangan. Hasil-hasil tersebut menunjukkan bahwa penggunaan beberapa artefak dan alat forensik dapat memperkaya informasi investigasi, tetapi sebagian besar penelitian tersebut berorientasi pada identifikasi bukti pada konteks serangan tertentu.

Penelitian ini memperluas pendekatan tersebut melalui penggunaan dua sumber bukti yang secara langsung merepresentasikan dua lapisan berbeda pada insiden yang sama. *File log* memberikan konteks semantik pada lapisan aplikasi, seperti alamat IP, waktu akses, metode HTTP, URL, dan status respons, sedangkan TCPDump menyediakan bukti bahwa komunikasi benar-benar terjadi pada lapisan jaringan. Ketika kedua sumber tersebut memiliki kesesuaian pada atribut utama, tingkat keyakinan terhadap rekonstruksi kejadian menjadi lebih kuat dibandingkan penggunaan satu sumber bukti saja. Sebaliknya, ketika terdapat selisih seperti pada IP 36.72.217.84, perbedaan tersebut dapat digunakan sebagai indikator untuk melakukan

pemeriksaan lebih mendalam terhadap waktu penangkapan paket, durasi akuisisi, kemungkinan *packet loss*, mekanisme pencatatan aplikasi, maupun perbedaan batas waktu antara proses TCPDump dan *access logging*.

Dari perspektif metode KUAD, hasil tersebut menunjukkan bahwa tahap *Acquisition* tidak berhenti pada proses pengumpulan data, tetapi dilanjutkan dengan pengujian konsistensi bukti sebelum digunakan dalam proses analisis. Korelasi antara *file log* dan TCPDump memberikan dasar yang lebih kuat untuk mengidentifikasi pola akses berulang dari beberapa sumber menuju satu target server. Namun, hasil penelitian saat ini belum dapat digunakan untuk menyatakan bahwa KUAD secara empiris lebih unggul daripada NFDLC atau framework forensik lainnya, karena belum dilakukan pengujian komparatif menggunakan dataset dan skenario serangan yang sama. Kontribusi yang dapat dinyatakan secara lebih tepat adalah bahwa penerapan KUAD pada penelitian ini mendukung integrasi proses akuisisi, validasi lintas sumber, analisis pola serangan, dan dokumentasi bukti secara sistematis.

Secara keseluruhan, analisis *multi-source* menunjukkan bahwa kombinasi *file log* dan TCPDump memberikan perspektif yang lebih lengkap terhadap kejadian DDoS dibandingkan analisis satu sumber. *File log* menjelaskan bagaimana layanan aplikasi merespons permintaan, sedangkan TCPDump mengkonfirmasi eksistensi aktivitas komunikasi pada jaringan. Kesesuaian pola pada dua alamat IP utama serta keberhasilan pencocokan dataset menjadi bukti mendukung penggunaan kedua sumber tersebut secara komplementer. Sementara itu, perbedaan yang relatif besar pada satu alamat IP menegaskan bahwa analisis forensik tidak seharusnya bergantung pada kesamaan jumlah rekaman secara absolut, tetapi harus didasarkan pada korelasi atribut dan konteks kejadian untuk menghasilkan kesimpulan yang lebih reliabel.

4. KESIMPULAN

Penelitian ini menunjukkan bahwa integrasi *file log* server dan TCPDump dalam kerangka *Knowledge Understanding Assessment Defense* (KUAD) mampu mendukung analisis forensik serangan DDoS pada *private cloud* OwnCloud secara lebih komprehensif melalui korelasi bukti pada *application layer* dan *network layer*. *File log* merekam 8.864 aktivitas, sedangkan TCPDump merekam 8.086 paket dari tiga alamat IP yang dianalisis. Validasi yang dilakukan menunjukkan 91% *agreement* berdasarkan kesesuaian alamat IP, waktu kejadian, dan tujuan akses. Hasil ini menunjukkan bahwa validitas bukti tidak bergantung pada kesamaan jumlah rekaman, tetapi pada korelasi atribut antar-sumber. Pola akses berulang dalam interval waktu singkat serta peningkatan beban CPU dan memori memperkuat indikasi terjadinya DDoS berbasis HTTP.

Efektivitas KUAD pada penelitian ini terletak pada integrasi proses akuisisi, validasi lintas sumber, analisis, dokumentasi, dan orientasi mitigasi dalam satu alur investigasi. Dibandingkan NFDLC, KUAD memberikan perluasan melalui tahap *mitigation*, sedangkan dibandingkan NIST, KUAD memberikan alur yang lebih spesifik untuk menghubungkan bukti forensik dengan tindakan pertahanan. Namun, penelitian ini belum melakukan perbandingan eksperimental langsung antara KUAD, NFDLC, dan NIST sehingga belum dapat menyatakan KUAD secara universal lebih unggul.

Keterbatasan penelitian meliputi penggunaan satu platform OwnCloud, satu jenis serangan berbasis GoldenEye, belum dilakukannya pengukuran *latency*, *throughput*, *packet loss*, *precision*, *recall*, dan efektivitas mitigasi *real-time*. Penerapan pada *public cloud*, container, Kubernetes, CDN, *load balancer*, atau *serverless computing* memerlukan penyesuaian sumber log, akses *packet capture*, sinkronisasi waktu, dan mekanisme korelasi. Penelitian selanjutnya perlu memperluas platform cloud, jenis serangan, dataset, indikator kuantitatif, serta melakukan perbandingan langsung antar-framework.

 DAFTAR PUSTAKA

- [1] O. Yoachimik, J. Pacheco, and C. One, “2025 Q4 DDoS threat report: A record-setting 31.4 Tbps attack caps a year of massive DDoS assaults,” Cloud Flare .
 - [2] K. Kent, S. Chevalier, T. Grance, and H. Dang, “Special Publication 800-86 Guide to Integrating Forensic Techniques into Incident Response Recommendations of the National Institute of Standards and Technology,” 2006. Accessed: Aug. 18, 2026. [Online]. Available: <https://doi.org/10.6028/NIST.SP.800-86>
 - [3] I. Riadi, S. Sunardi, and F. Tella, “Spamming Forensic Analysis Using Network Forensics Development Life Cycle Method,” *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 6, no. 1, pp. 108–117, Feb. 2022, doi: 10.29407/intensif.v6i1.16830.
 - [4] H. Wintolo, I. Riadi, and A. Yudhana, “Enhancing Private Cloud Security Using Knowledge Understanding Assessment Defense Method for Distributed Denial of Service Attack Mitigation,” *International Journal of Safety and Security Engineering*, vol. 15, no. 11, pp. 2323–2331, Dec. 2025, doi: 10.18280/ijssse.151112.
 - [5] Presiden Republik Indonesia, *Undang-undang Republik Indonesia Nomor 1 Tahun 2024*. 22024.
 - [6] B. Guttman, D. R. White, and T. Walraven, “Digital Evidence Preservation Considerations for Evidence Handlers,” Sep. 2022. doi: 10.6028/NIST.IR.8387.
 - [7] A. R. Kunduru, “Security Concerns and Solutions for Enterprise Cloud Computing Applications,” *Asian Journal of Research in Computer Science*, vol. 15, no. 4, pp. 24–33, May 2023, doi: 10.9734/ajrcos/2023/v15i4327.
 - [8] H. Wintolo, I. Riadi, and A. Yudhana, “Intrusion Detection Analysis on Open Journal System Services Using Network Forensic Development Life Cycle Method,” *SKANIKA: Sistem Komputer dan Teknik Informatika*, vol. 8, no. 1, pp. 133–144, 2025, Accessed: Apr. 15, 2025. [Online]. Available: <https://doi.org/10.36080/skanika.v8i1.3284>
 - [9] A. Yudhana, I. Riadi, and B. Putra, “Analisis Kinerja Perangkat Lunak Forensic Imaging Pada Sistem Operasi Linux Menggunakan Metode Static Forensic,” *INSECT*, vol. 8, no. 1, 2022.
 - [10] I. Riadi, A. Z. Ifani, and R. S. Kusuma, “Optimization and Evaluation of Authentication System using Blockchain Technology,” *Emerging Science Journal*, vol. 4, no. Special issue, pp. 225–240, 2020, doi: 10.28991/esj-2021-SP1-015.
 - [11] A. Al-Dhaqm *et al.*, “Digital Forensics Subdomains: The State of the Art and Future Directions,” *IEEE Access*, vol. 9, pp. 152476–152502, 2021, doi: 10.1109/ACCESS.2021.3124262.
 - [12] A. M. Alashjaee and F. Alqahtani, “Improving Digital Forensic Security: A Secure Storage Model With Authentication and Optimal Key Generation Based Encryption,” *IEEE Access*, vol. 12, pp. 29738–29747, 2024, doi: 10.1109/ACCESS.2024.3369604.
 - [13] J. Oh, S. Lee, and H. Hwang, “Forensic Recovery of File System Metadata for Digital Forensic Investigation,” *IEEE Access*, vol. 10, pp. 111591–111606, 2022, doi: 10.1109/ACCESS.2022.3213030.
 - [14] A. Azzi, G. Heilemann, D. Georg, S. A. Pawiro, T. Mart, and W. Lechner, “Impact of log file source and data frequency on accuracy of log file-based patient specific quality assurance,” *Z. Med. Phys.*, vol. 35, no. 2, pp. 152–160, May 2025, doi: 10.1016/j.zemedi.2023.05.006.
 - [15] D. Wilkinson *et al.*, “Kilovoltage triggered imaging for moderately hypofractionated prostate radiotherapy: Insights into intrafraction motion and intervention through log file analysis,” *Physica Medica*, vol. 134, Jun. 2025, doi: 10.1016/j.ejmp.2025.104998.
 - [16] H. Choi and S. Lee, “Forensic analysis of SQL server transaction log in unallocated area of file system,” *Forensic Science International: Digital Investigation*, vol. 46, Oct. 2023, doi: 10.1016/j.fsidi.2023.301605.
-

- [17] V. Zeufack, D. Kim, D. Seo, and A. Lee, "An unsupervised anomaly detection framework for detecting anomalies in real time through network system's log files analysis," *High-Confidence Computing*, vol. 1, no. 2, Dec. 2021, doi: 10.1016/j.hcc.2021.100030.
- [18] Y. Azizi, M. Azizi, and M. Elboukhari, "Log files Analysis Using MapReduce to Improve Security," in *Procedia Computer Science*, Elsevier B.V., 2019, pp. 37–44. doi: 10.1016/j.procs.2019.01.006.
- [19] S. Shitharth and D. P. Winston, "A Comparative Analysis between Two Countermeasure Techniques to Detect DDoS with Sniffers in a SCADA Network," *Procedia Technology*, vol. 21, pp. 179–186, 2015, doi: 10.1016/j.protcy.2015.10.086.
- [20] D. Pinto, I. Amorim, E. Maia, and I. Praça, "A review on intrusion detection datasets: tools, processes, and features," *Computer Networks*, vol. 262, May 2025, doi: 10.1016/j.comnet.2025.111177.
- [21] E. Degirmenci, Y. S. Kurca, I. Ozcelik, and A. Yazici, "ROSIDS23: Network intrusion detection dataset for robot operating system," *Data Brief*, no. 51, 2023, doi: 10.5281/zenodo.10014434.
- [22] S. A. Muntaka *et al.*, "Mapping the invisible internet: Framework and dataset," *Data Brief*, no. 63, 2025, doi: 10.5281/zenodo.15278912.
- [23] H. Zhang, "Simulation of network forensics model based on wireless sensor networks and inference technology," *Measurement: Sensors*, vol. 34, p. 101261, Aug. 2024, doi: 10.1016/j.measen.2024.101261.
- [24] H. Wintolo, I. Riadi, and A. Yudhana, "Post Attack Mitigation on Open Journal System Services using Knowledge Understanding Assessment Defense (KUAD) Method," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, Oct. 2025, doi: 10.22219/kinetik.v10i4.2279.