KRESNA: Jurnal Riset dan Pengabdian Masyarakat
Volume 4, Nomor 2, November, 2024, Hal: 208-216 E-ISSN: 2809-6509

Deteksi Anomali Jaringan Menggunakan *Isolation Forest* pada *Log* Wazuh dengan Pemberitahuan WhatsApp di PT XYZ

Network Anomaly Detection Using Isolation Forest on Wazuh Logs with WhatsApp Notifications at PT XYZ

Richie Punta Dewa[1], Windarto[2*]

[12] Fakultas Teknologi Informasi
Universitas Budi Luhur
E-mail: [1]richiepuntadewa48@gmail.com, [2*]windarto@budiluhur.ac.id
(* corresponding author)

Abstract

PT XYZ, a financial technology service provider, faces significant challenges in maintaining cybersecurity, particularly in manually monitoring and analyzing log alerts. This manual process is inefficient and may overlook critical threats, making it difficult for the Security Operations Center (SOC) team to process large volumes of data quickly and accurately. This study aims to enhance anomaly detection performance on PT XYZ's computer network by implementing the Isolation Forest algorithm. Three main programs are used in the sistem: `detect_anomalies.py` to detect anomalies in log alert data from the Wazuh platform, `watchdog_script.py` to monitor changes in the `alerts.json` file and automatically trigger detection, and `send_notification.py` to send anomaly notifications via WhatsApp. Testing results show that the Isolation Forest algorithm successfully detected 17 anomalies out of 2,000 alert data with 100% accuracy and an average processing time of 10.9 seconds. Real-time notifications sent within 5 seconds enable the SOC team to respond quickly to cybersecurity threats. The implementation of this sistem reduces reliance on time-consuming manual processes, increasing efficiency and effectiveness in threat detection. The findings of this study make a tangible contribution by providing a model that can be adopted by other fintech companies to strengthen their network and information sistem security.

Keywords**:** *cybersecurity, anomaly detection, Isolation Forest, Wazuh, real-time notification)*

Abstrak

PT XYZ sebagai penyedia layanan teknologi keuangan menghadapi tantangan signifikan dalam menjaga keamanan siber, terutama dalam memantau dan menganalisis *log* peringatan (log alert) secara manual. Proses manual ini tidak efisien dan berpotensi melewatkan ancaman penting, mengakibatkan tim Security Operations Center (SOC) kesulitan memproses data dalam jumlah besar dengan cepat dan akurat. Penelitian ini bertujuan untuk meningkatkan efektivitas kinerja dalam mendeteksi anomali pada jaringan komputer PT XYZ dengan menerapkan algoritma Isolation Forest. Tiga program utama digunakan dalam sistem yaitu detect_anomalies.py untuk mendeteksi anomali dalam data *log alert* dari platform Wazuh, watchdog_script.py untuk memantau perubahan pada file alerts.json dan menjalankan deteksi secara otomatis, serta send_notification.py untuk mengirimkan notifikasi anomali melalui WhatsApp. Hasil pengujian menunjukkan bahwa algoritma *Isolation Forest* berhasil mendeteksi 17 anomali dari 2000 data alert dengan akurasi 100% dan rata-rata waktu pemrosesan 10,9 detik. Notifikasi real-time yang dikirimkan dalam waktu 5 detik memungkinkan respon cepat dari tim SOC untuk menangani sebuah ancaman siber. Implementasi sistem ini mengurangi ketergantungan pada proses manual yang membutuhkan waktu lebih lama, meningkatkan efisiensi, serta efektivitas deteksi terhadap ancaman siber. Hasil dari penelitian ini memberikan kontribusi yang nyata dengan menyediakan sebuah model yang dapat diadopsi oleh perusahaan *fintech* lain untuk memperkuat keamanan jaringan dan sistem informasi mereka.

Kata kunci: keamanan siber, deteksi anomali, *Isolation Forest*, Wazuh, notifikasi real-time

1. PENDAHULUAN

Di era digital sekarang, keamanan jaringan menjadi hal yang sangat penting bagi industri dan perusahaan untuk melindungi data serta informasi sensitif yang dimilikinya [1]. PT XYZ, sebagai penyedia layanan *fintech* yang mengutamakan solusi pembayaran digital, memiliki tanggung jawab besar dalam memastikan kelancaran dan keamanan transaksi bagi para pengguna layanannya. Ancaman siber yang semakin kompleks dan bervariasi tidak hanya mengancam kelangsungan bisnis tetapi juga dapat merusak reputasi perusahaan serta menurunkan tingkat kepercayaan pelanggan. Dalam menghadapi tantangan ini, PT XYZ telah membentuk *Security Operation Center (SOC)* di divisi *IT Security*, yang menggunakan *platform* Wazuh untuk pemantauan dan deteksi terhadap ancaman keamanan jaringan. Namun, kendala masih ditemui karena proses pengecekan *log alert* yang masih dilakukan secara manual. Pendekatan tradisional dalam pemfilteran *log* seringkali hanya terbatas pada aturan statis, yang menyebabkan banyak aktivitas mencurigakan yang tidak terdeteksi atau bahkan aktivitas normal yang dianggap sebagai ancaman [2].

Penelitian sebelumnya menemukan bahwa pengetahuan dan pemahaman pengguna mengenai kerusakan jaringan Wi-Fi dapat ditingkatkan dengan bantuan sistem pakar yang merupakan bagian dari kecerdasan buatan (artificial intelligence). Sistem pakar ini dapat membantu admin IT atau teknisi jaringan dalam mengidentifikasi serta memberikan solusi awal untuk kerusakan, sehingga meminimalisir masalah yang terjadi. Selain itu, aplikasi ini mendukung admin jaringan dan IT Support dalam mendiagnosis kerusakan jaringan Wi-Fi [3]. Penelitian lain juga menunjukkan bahwa penerapan kecerdasan buatan menggunakan algoritma machine learning efektif dalam mendeteksi anomali pada jaringan komputer, sehingga meningkatkan efektivitas keamanan siber [4]. Studi relevan menyoroti pentingnya menjaga keseimbangan antara akurasi dan efisiensi algoritma untuk menghadapi serangan yang semakin canggih dan bervariasi. Fokus utama penelitian ini adalah mengembangkan algoritma yang mampu mendeteksi anomali secara cepat dan akurat pada lalu lintas jaringan yang dinamis, serta dapat beradaptasi secara real-time terhadap ancaman yang muncul [4].

Dalam penelitian ini, peneliti berupaya mengatasi tantangan keamanan yang dihadapi PT XYZ dengan mengimplementasikan sistem deteksi anomali menggunakan algoritma *Isolation Forest* pada log *alert* yang dihasilkan oleh Wazuh. Pendekatan ini termasuk dalam bagian dari *artificial intelligence* (AI), di mana algoritma *machine learning* seperti *Isolation Forest* memungkinkan sistem secara otomatis mengenali pola dan mendeteksi aktivitas mencurigakan atau anomali dalam data. Algoritma ini dipilih karena kemampuannya yang efektif dalam mendeteksi data anomali. Selain itu, sistem ini akan diintegrasikan dengan notifikasi *real-time* melalui aplikasi pesan WhatsApp, memungkinkan tim *SOC* PT XYZ menerima peringatan segera dimanapun mereka berada, yang pada gilirannya akan mempercepat respon terhadap potensi ancaman. Deteksi anomali ini bekerja dengan cara mengidentifikasi penyimpangan dari pola normal, yang kemudian ditandai sebagai kemungkinan serangan [5]. Integrasi AI dalam proses ini meningkatkan efektivitas dan efisiensi dalam mengidentifikasi ancaman keamanan secara *real-time*, membantu melindungi jaringan komputer PT XYZ dari potensi serangan dan kerentanan. Oleh karena itu, deteksi anomali trafik jaringan menjadi kebutuhan mendesak untuk menjaga keamanan dan kestabilan jaringan [6].

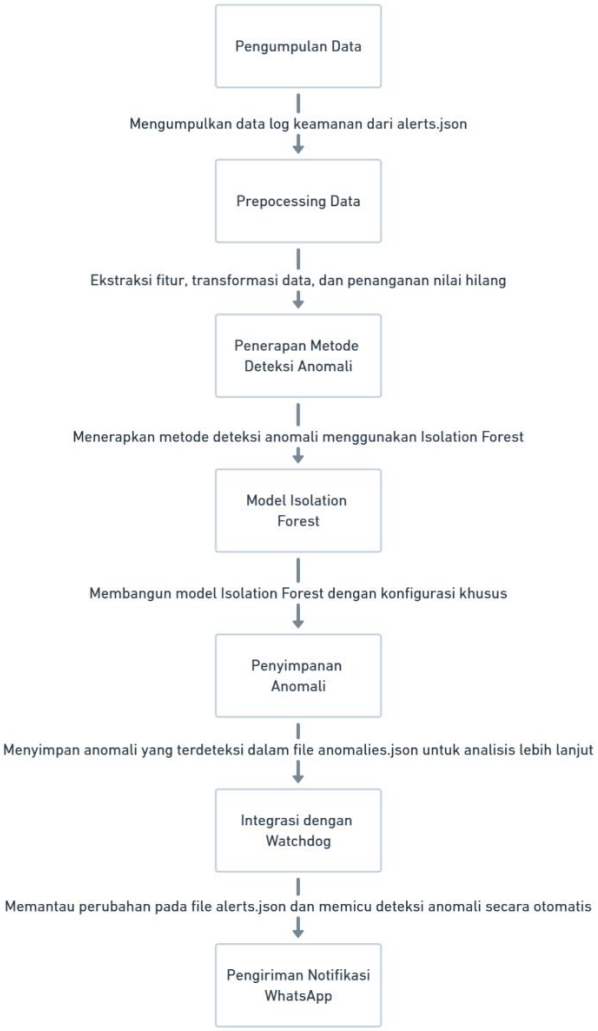
Penelitian ini memiliki tujuan untuk mengembangkan suatu sistem deteksi anomali berbasis *machine learning* yang lebih efisien dan responsif guna meningkatkan keamanan siber di PT XYZ. Melalui penerapan teknologi ini, PT XYZ dapat mengurangi ketergantungan pada proses manual yang memakan waktu lebih lama serta mempercepat tindakan pencegahan terhadap ancaman siber, sehingga memperkuat posisi perusahaan dalam menjaga kepercayaan pelanggan di era digital yang semakin rentan terhadap ancaman siber.

2. METODE PENELITIAN

Penelitian ini menerapkan metode eksperimental dengan fokus pada pengiriman notifikasi deteksi anomali melalui pesan WhatsApp. Tujuan dari penelitian ini adalah mendeteksi anomali pada jaringan komputer yang akan dikirimkan via Whatsapp guna membantu tim SOC dalam mendeteksi adanya ancaman keamanan jaringan menggunakan algoritma *Isolation Forest*.

2.1. Tahapan Penelitian

Penelitian ini dilaksanakan melalui beberapa tahap untuk mendeteksi anomali pada data *log* yang dihasilkan oleh sistem Wazuh. Langkah pertama yaitu data dikumpulkan dari file *alerts.json*, yang berisi informasi penting seperti waktu kejadian, level ancaman, nama agen, dan lokasi, yang semuanya digunakan untuk mendeteksi potensi ancaman dalam sistem. Setelah data terkumpul, lalu proses selanjutnya yaitu pengolahan data (*preprocessing*), di mana parameter-parameter penting diekstrak dan data yang bersifat kategorikal diubah menjadi bentuk numerik. Lalu nilai yang hilang akan diganti dengan nilai *default* agar data dapat digunakan untuk analisis lebih lanjut. Setelah data melalui proses *preprocessing*, algoritma *Isolation Forest* diterapkan untuk mendeteksi anomali. Pemilihan algoritma ini didasarkan pada kemampuannya dalam menangani data yang tidak terstruktur dan efisiensinya dalam mendeteksi anomali dengan akurasi tinggi. *Isolation Forest* bekerja dengan membangun sejumlah pohon keputusan yang secara efektif memisahkan data normal dari anomali, sehingga memungkinkan identifikasi pola yang tidak biasa dalam *dataset*. Lalu langkah berikutnya hasil dari deteksi disimpan dalam file *anomalies.json*. Untuk memastikan sistem dapat merespon ancaman secara cepat. Program *Watchdog* digunakan untuk memantau perubahan pada file *alerts.json* dimana setiap kali ada perubahan, deteksi anomali akan dilakukan secara otomatis. Jika ditemukan adanya anomali, sistem akan mengirimkan notifikasi ke tim SOC melalui pesan WhatsApp yang berisi informasi tentang ancaman yang terdeteksi, seperti IP agen, nama agen, dan waktu kejadian, untuk memudahkan tim keamanan dalam merespon ancaman tersebut dengan cepat. pada gambar 1 merupakan tahapan-tahapan pada program dalam mendeteksi anomali lalu program akan mengirimkan notifikasi pesan whatsapp.



Gambar 1. *Flowchart* Proses Deteksi Anomali dan Pengiriman Notifikasi

2.2. Keamanan Jaringan

Keamanan jaringan merujuk pada perlindungan sumber daya dari upaya pengungkapan, modifikasi, penggunaan, pembatasan, dan perusakan oleh pihak yang tidak berwenang. Sistem ini berfungsi untuk mencegah aktivitas yang tidak diinginkan dengan cara mengidentifikasi pengguna yang tidak memiliki hak akses dalam jaringan. Koneksi antar komputer, baik melalui jaringan kabel maupun nirkabel, memungkinkan pihak lain mengakses, mengubah, atau menghapus data dalam jaringan tersebut. Keamanan jaringan melibatkan perlindungan baik di tepi maupun di dalam jaringan, menggunakan pendekatan berlapis. Kerentanannya bisa ditemukan pada perangkat, jalur data, aplikasi, hingga pengguna [7]. Fatur menambahkan bahwa keamanan sistem jaringan komputer bertujuan untuk melindungi data dan sumber daya dari akses yang tidak sah, kerusakan, dan kegagalan penggunaan. Jaringan komputer adalah aset berharga yang perlu dijaga keamanannya, baik dari sisi fisik maupun non-fisik. Salah satu tantangan utama dalam pengelolaan keamanan jaringan adalah rentannya terhadap serangan atau perusakan sistem, dengan meningkatnya jumlah serangan yang terjadi akibat kerentanannya dalam sistem jaringan [6].

2.3. Deteksi Anomali

Menurut Anwar deteksi anomali adalah salah satu metode dalam *Intrusion Detection Sistem* (IDS) yang bertujuan untuk mengidentifikasi serangan yang menyimpang dari pola normal berdasarkan probabilitas statistik. Dengan semakin tingginya penggunaan internet, risiko serangan dari *intruder* atau *cracker* yang mengeksploitasi kelemahan protokol internet dan perangkat lunak juga meningkat. Deteksi anomali menjadi sangat penting untuk mengamankan sistem dari ancaman tersebut [7].

Deteksi anomali dalam analisis data adalah proses penting untuk mengidentifikasi pengamatan atau pola yang berbeda dari mayoritas data. Prediksi berbasis model, seperti *Isolation Forest* atau *Gaussian Mixture Models*, membangun model dengan menggunakan data normal dan mengidentifikasi data yang menyimpang dari model sebagai anomali [8].

2.4. Wazuh

Wazuh adalah perangkat lunak sumber terbuka yang berperan sebagai sistem deteksi keamanan berbasis *host* (endpoint). Wazuh mengintegrasikan fitur *Extended Detection and Response* (XDR) serta *Security Information and Event Management* (SIEM), yang mencakup modul-modul untuk analisis *log*, deteksi intrusi dan *malware*, pemantauan integritas *file*, penilaian konfigurasi berdasarkan standar industri, deteksi kerentanan, dan kepatuhan terhadap regulasi keamanan. Perangkat lunak ini meningkatkan visibilitas keamanan dengan memonitor aktivitas pada *host* di tingkat sistem operasi dan aplikasi. Arsitektur Wazuh terdiri dari tiga komponen utama (Wazuh Indexer, Wazuh Server, dan Wazuh Dashboard) serta komponen *endpoint* (Wazuh Agent) [9].

a. *Wazuh Indexer*

Wazuh Indexer adalah mesin pencari yang digunakan untuk mengindeks dan menyimpan peringatan yang dihasilkan oleh Wazuh Server, memudahkan proses pencarian dan analisis data. Data disimpan dalam format dokumen *JSON*, di mana kumpulan dokumen yang memiliki kaitan disebut sebagai indeks [10].

b. *Wazuh Server*

Wazuh Server bertugas menganalisis data yang diterima dari *Wazuh Agent*. Data tersebut diproses menggunakan *decoder* dan aturan tertentu dengan bantuan *threat intelligence* untuk mendeteksi potensi ancaman. Selain fungsi analisis, *Wazuh Server* juga digunakan untuk mengelola *Wazuh Agent*, termasuk pengaturan konfigurasi dan pembaruan perangkat lunak [10].

c. *Wazuh Dashboard*

Wazuh Dashboard adalah antarmuka *web* yang digunakan untuk visualisasi dan analisis data keamanan. *Dashboard* ini menyajikan informasi mengenai kejadian keamanan, kepatuhan terhadap peraturan, deteksi kerentanan aplikasi, pemantauan integritas *file*, penilaian konfigurasi, dan pemantauan aktivitas di infrastruktur *cloud*. Selain itu, *Wazuh Dashboard* juga menyediakan fasilitas untuk mengelola konfigurasi Wazuh dan memantau status operasionalnya [10].

d. *Wazuh Agent*

Wazuh Agent adalah komponen yang diinstal pada perangkat *endpoint* seperti *Linux, Windows, macOS, Solaris, AIX*, dan sistem operasi lainnya, untuk mendeteksi, mencegah, dan merespon ancaman

keamanan. Agen ini dapat digunakan pada berbagai *platform* dan beroperasi di *endpoint* yang ingin dipantau. *Wazuh Agent* mengirimkan data secara *real-time* ke *Wazuh Server* melalui saluran komunikasi yang terenkripsi dan terautentikasi, memastikan keamanan data yang dikirimkan [10].

2.5. Algoritma *Isolation Forest*

Isolation Forest merupakan metode deteksi anomali dalam *machine learning* yang tidak diawasi (unsupervised) yang pertama kali diperkenalkan oleh [11]. Algoritma ini termasuk dalam kategori *unsupervised learning* dan *nonparametric*, yang berbasis pada algoritma pohon keputusan (decision trees). Proses deteksi anomali dengan menggunakan *Isolation Forest* terdiri dari dua tahap, yaitu tahap pelatihan dan tahap evaluasi. Pada tahap pelatihan, mesin membangun pohon isolasi berdasarkan sampel dari seluruh *dataset*. Sedangkan pada tahap evaluasi, setiap individu atau *instance* dalam *dataset* akan melalui pohon isolasi untuk menghitung skor anomali masing-masing *instance* [12].

Menurut Chaidir, *Isolation Forest* adalah metode *unsupervised machine learning* yang digunakan untuk mendeteksi anomali. Metode ini membangun sebuah *ensemble* pohon keputusan yang strukturnya mirip dengan pohon pencarian biner. Prinsip dasar dalam pendeteksian anomali menggunakan *Isolation Forest* adalah dengan memisahkan ruang data menjadi beberapa subruang secara acak, karena data anomali cenderung terletak pada titik yang jarang muncul dalam *dataset*. Hal ini memungkinkan algoritma untuk mengidentifikasi titik anomali dengan cepat. *Isolation Forest* dikenal karena kemampuannya mendeteksi anomali dengan efisiensi tinggi, dimana algoritma ini dapat mengisolasi *outliers* lebih cepat dibandingkan data normal [12][13]. Skor anomali (anomaly score) adalah nilai yang menunjukkan kriteria anomali pada data tertentu. Secara matematis, skor anomali dihitung menggunakan rumus (1):

$$s(x+n) = 2^{-\frac{E(h(x))}{c(n)}} \quad (1)$$

Path length adalah ukuran jarak yang menunjukkan titik di mana data tidak dapat dibagi lebih lanjut dengan *root node* dalam struktur algoritma pohon keputusan. Skor anomali (anomaly score) memiliki rentang nilai antara -1 hingga 1. Semakin mendekati -1, nilai tersebut dianggap sebagai anomali, sedangkan semakin mendekati 1, nilai tersebut dianggap normal [12].

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan pengujian dari metode *Isolation Forest* dalam mendeteksi anomali. Pengujian yang dilakukan ialah mulai dari pengujian akurasi, pengujian kecepatan, pengujian kirim notifikasi, dan pengujian integrasi. pengujian tersebut dapat dijelaskan berikut ini.

3.1. Sumber Data

Data yang digunakan dalam penelitian ini berupa *log* yang dihasilkan dari notifikasi *alert Wazuh*. Data penelitian ini diperoleh dari 2000 sampel *log* yang diterima oleh sistem keamanan Wazuh milik PT XYZ. *Log* ini direkam dalam format *JSON* dan mencakup berbagai aktivitas jaringan serta potensi ancaman siber. File ini berisi informasi tentang peringatan keamanan yang dihasilkan oleh sistem, termasuk rincian seperti *timestamp*, level aturan, nama agen, IP agen, nama manajer, *log* lengkap, *file* data, dan lokasi. Data ini penting untuk mendeteksi pola-pola yang tidak biasa yang mungkin menunjukkan adanya aktivitas mencurigakan atau anomali dalam sistem keamanan. Parameter yang akan digunakan dalam penelitian ini meliputi *timestamp, rule.level, agent.name, agent.ip, manage_name, Full_log, data_file dan location.*

3.2. Pengujian Akurasi

Dalam pengujian akurasi, pengujian bertujuan untuk mengevaluasi seberapa baik model *Isolation Forest* dalam mendeteksi anomali dengan benar yaitu dengan mengenali data yang benar-benar merupakan anomali *(true positives)* dan menghindari kesalahan dalam mengklasifikasikan data yang bukan anomali *(false positives)*. Data yang digunakan dalam pengujian diambil dari file *alerts.json*, yang mencakup parameter-parameter penting seperti waktu kejadian (*timestamp*), level aturan (*rule_level*), nama dan alamat IP agen, serta informasi terkait lainnya.

Setelah data diambil, proses selanjutnya adalah transformasi data untuk analisis, termasuk mengkonversi atribut kategorikal menjadi format numerik menggunakan *LabelEncoder* dan mengisi nilai-nilai kosong dengan -1. Pada penelitian ini Model *Isolation Forest* dikonfigurasikan dengan tingkat kontaminasi sebesar 0,05, yang menunjukkan sekitar 5% data mungkin mengandung anomali. Lalu *Isolation Forest* digunakan untuk mendeteksi anomali dalam *dataset*, selanjutnya hasil deteksi dibandingkan dengan label data yang sudah diverifikasi guna mengukur tingkat keakuratannya. Dalam pengujian akurasi ini, jumlah data yang digunakan yaitu sebanyak 2000 data.

Tabel 1. Pengujian Anomali

Deskripsi Anomali	Total Anomali	True Anomali	False Anomali	Total Anomali
Listened ports status (netstat) changed (new port opened or closed)	10	17	0	17
Integrity checksum changed.	2			
SCA summary: CIS Benchmark for CentOS 7: Score less than 50% (38)	1			
Auditd: Daemon End	0			
vsftpd: Multiple FTP connection attempts from same source IP	4			

Berdasarkan Tabel 1 setelah melalui proses deteksi anomali menggunakan model *Isolation Forest*, Dari 2000 data yang diuji, model berhasil mendeteksi 17 anomali dengan total *alert* yang di eliminasi (rule level 1-5) sebanyak 336 *alert* dan dengan waktu proses kurang dari 5 detik. Jika dilihat pada tabel 1, peneliti mendapatkan waktu 1 detik, dan tidak ada salah deteksi. Adapun deskripsi *alert* yang didapatkan adalah *alert* dengan nilai *true positive* sehingga bisa dimasukan kedalam kategori anomali. Untuk *alert* yang didapatkan yaitu *Listened ports status (netstat) changed (new port opened or closed), Integrity checksum changed, SCA summary: CIS Benchmark for CentOS 7: Score less than 50% (38), vsftpd: Multiple FTP connection attempts from same source IP*.

3.3. Pengujian Kecepatan

Pengujian kecepatan bertujuan untuk memastikan sistem dapat memproses data dengan cepat dan efisien, terutama dalam kondisi *real-time*. Selama pengujian, waktu pemrosesan diukur dari saat data dimuat hingga hasil deteksi anomali disimpan. Pengukuran ini mencakup analisis durasi yang dibutuhkan untuk berbagai tahapan, seperti proses deteksi anomali dan penyimpanan hasilnya. Evaluasi ini juga menilai seberapa baik sistem dapat menangani volume data yang meningkat dan mempertahankan akurasi dalam waktu yang singkat. Dalam pengujian kecepatan ini, variasi jumlah data *log* digunakan untuk menguji bagaimana kinerja model dibawah kondisi yang berbeda.

Tabel 2. Pengujian Kecepatan

Banyak *Log*	Waktu Proses (Detik)	Keterangan (Jumlah Anomali)	Total *Alert* Yang Di Eliminasi (1-5)	Rata-Rata (Detik)
200	1	0	0	10.9
300	1	3	44	
500	2	7	139	
800	2	9	175	
1000	2	13	255	
1100	2	14	266	
1300	2	15	284	
1500	1	15	290	
1800	1	16	319	
2000	1	17	336	

Berdasarkan pada tabel setelah melalui proses pengujian kecepatan pemrosesan data dari 2000 data yang diuji dengan membuat variasi pada jumlah *log* yang masuk, model berhasil mendeteksi 17 dengan waktu proses kurang dari 30 detik jika dilihat dalam tabel, rata-rata waktu yang didapatkan adalah 10.9 detik.

Tabel 3. Analisis Kecepatan Akurasi

Jumlah Data	Hasil yang Diharapkan	Pengamatan	Hasil Pengujian
2000	Kurang dari 30 detik	Pada tahap pengamatan ini, kecepatan pemrosesan data diuji dengan menggunakan 2000 *log* yang ditambahkan dengan cara membedakan jumlah *log* yang masuk dan dikirimkan satu persatu. Hasil pengamatan menunjukkan bahwa sistem mampu memproses data dalam waktu kurang dari 30 detik. Pengamatan ini menunjukkan bahwa sistem dapat menangani data dengan cepat dan efisien, bahkan ketika data ditambahkan dengan kecepatan yang cukup tinggi.	kurang dari 30 detik

3.4. Pengujian Fungsi *Send Notifications*

Pengujian fungsi *send notifications* bertujuan untuk menilai efektivitas sistem dalam mengirimkan notifikasi via pesan WhatsApp untuk setiap anomali yang terdeteksi dengan cepat dan efisien. Dalam pengujian ini, waktu mulai dan selesai pengiriman notifikasi dicatat secara rinci untuk mengukur durasi keseluruhan dari deteksi anomali hingga pengiriman notifikasi. Evaluasi ini bertujuan untuk memastikan bahwa sistem tidak hanya mampu mendeteksi anomali secara akurat tetapi juga responsif dalam pengiriman notifikasi secara tepat waktu.

Tabel 4. Pengujian *Send_notification*

Jumlah anomali	Hasil yang Diharapkan	waktu proses (detik)	waktu mulai	waktu berakhir	Durasi
17	Kurang dari 30 detik	5	3:56:19	3:56:24	0:00:05

Berdasarkan tabel 4 setelah melalui proses pengujian kecepatan pengiriman notifikasi dari 2000 data yang diuji, program berhasil mendeteksi 17 anomali dengan total *alert* yang di eliminasi (rule level 1-5) sebanyak 336 *alert* dan dengan waktu proses kurang dari 30 detik jika dilihat dalam tabel waktu yang didapatkan program dalam mengirim notifikasi ke pesan WhatsApp adalah 5 detik.

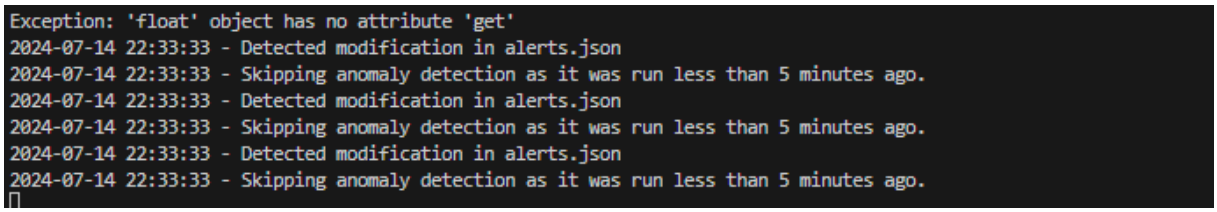
Tabel 5. Analisis Pengujian Pengiriman Notifikasi

Jumlah anomali	Hasil yang Diharapkan	Pengamatan	Hasil Pengujian
17	Kurang dari 30 detik	Pada tahap pengamatan ini, kecepatan dalam mengirimkan notifikasi diuji dengan menggunakan 17 anomali yang terdeteksi. Lalu Hasil pengamatan menunjukkan bahwa sistem mampu mengirimkan notifikasi dalam waktu kurang dari 30 detik. Pengamatan ini menunjukkan bahwa sistem dapat menangani data dengan cepat dan efisien, bahkan ketika data ditambahkan dengan kecepatan yang cukup tinggi.	kurang dari 30 detik

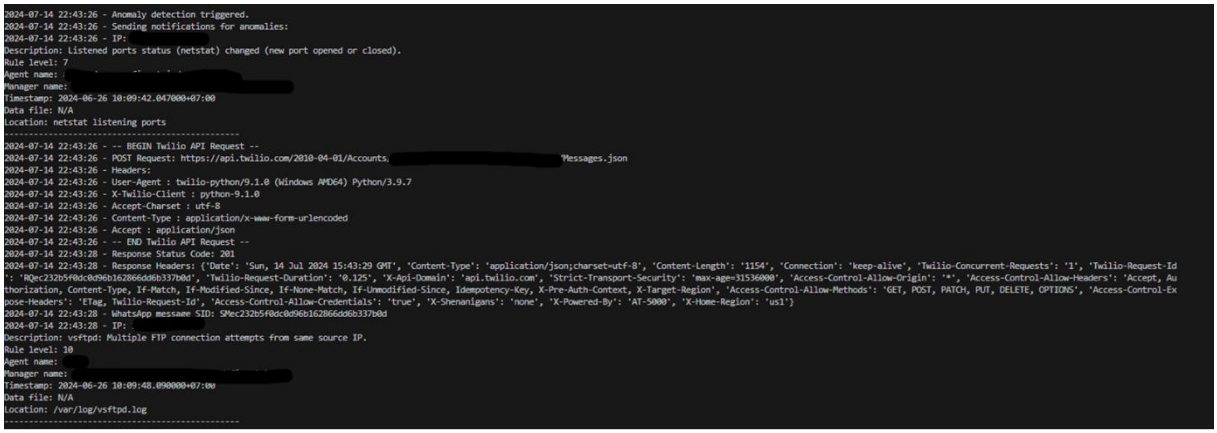
3.5. Pengujian Integrasi

Pengujian integrasi sistem dilakukan untuk mengevaluasi bagaimana ketiga komponen utama proses deteksi anomali, pemantauan perubahan, dan pengiriman notifikasi bekerja secara bersama-sama. Pengujian dimulai dengan mensimulasikan perubahan pada file *alerts.json* guna menguji apakah *program watchdog_script.py* dapat secara otomatis memicu deteksi anomali menggunakan algoritma *Isolation Forest*. Pada gambar 2 menunjukkan bahwa program tersebut bekerja efektif, mendeteksi perubahan pada file, dan memicu analisis anomali setiap lima menit sesuai dengan perubahan yang terdeteksi. Lalu tahap selanjutnya pengujian dilakukan pada fungsi pengiriman notifikasi untuk memastikan bahwa program *send_notification.py* dapat mengirimkan pesan WhatsApp yang berisi informasi akurat tentang anomali yang ditemukan. Pada gambar 3 menunjukkan bahwa program berhasil mengirimkan notifikasi dengan informasi yang tepat, tanpa terjadi duplikasi pesan yang tidak diinginkan. Secara keseluruhan, pada gambar 3 mengonfirmasi bahwa sistem deteksi anomali dan pengiriman notifikasi ini berjalan dengan efisien dan dapat memberikan informasi yang relevan serta tepat waktu kepada Tim SOC PT XYZ. Hal ini menunjukkan bahwa sistem ini dapat diandalkan untuk

menggantikan metode manual sebelumnya dan mampu berfungsi secara efektif dalam lingkungan operasional yang dinamis.



Gambar 2. Pengawasan (watchdog_script)



Gambar 3. Send_notification

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan dan menguji algoritma *Isolation Forest* untuk mendeteksi anomali pada *log* keamanan jaringan dari *platform Wazuh* di PT XYZ. Berdasarkan hasil evaluasi, algoritma ini menunjukkan kinerja yang sangat baik dalam mendeteksi anomali dengan akurasi 100% pada sampel data *log* sebanyak 2000 data. Sistem ini juga berhasil mengirimkan notifikasi melalui pesan WhatsApp dengan waktu pengiriman rata-rata hanya 5 detik, dan mendeteksi 17 anomali yang teridentifikasi sebagai *true positives*. Kelebihan utama dari sistem ini adalah keakuratan dalam mendeteksi anomali, kecepatan pengiriman notifikasi yang efisien, serta integrasi yang berhasil antara modul-modul program *detect_anomalies.py*, *send_notification.py*, dan *watchdog.py*. Selain itu, *watchdog.py* mampu memantau perubahan pada file *alerts.json* dan memicu deteksi anomali setiap 5 menit, memberikan respon mendekati *real-time*. Penelitian ini membuktikan bahwa sistem deteksi anomali yang dikembangkan mampu menjadi solusi yang andal dalam mendukung pengelolaan keamanan jaringan di PT XYZ yang mana dapat menggantikan metode manual dengan pendekatan otomatis yang lebih efisien. Namun, terdapat beberapa kekurangan yang perlu diperhatikan. Pengujian saat ini masih menggunakan dataset dengan volume terbatas dan kondisi yang relatif sederhana. Untuk memastikan ketahanan dan efektivitas sistem dalam skenario yang lebih kompleks, diperlukan uji coba tambahan dengan *dataset* yang lebih besar dan variatif. Selain itu, meskipun kecepatan pengiriman notifikasi sudah baik, analisis lebih lanjut diperlukan untuk menangani situasi dengan volume data yang sangat besar. Untuk penelitian lebih lanjut, disarankan untuk menguji sistem menggunakan data yang lebih beragam dan pada skenario operasional yang lebih kompleks. Pengembangan fitur tambahan seperti analisis prediktif dan optimasi algoritma juga dapat meningkatkan efisiensi serta keandalan sistem secara keseluruhan, sekaligus mengurangi latensi dalam proses deteksi dan pengiriman notifikasi.

DAFTAR PUSTAKA

[1] P. P. Putra, “Pengembangan Sistem Keamanan Jaringan Menggunakan Rumusan Snort Rule (HIDS) untuk Mendeteksi Serangan Nmap,” *SATIN - Sains dan Teknol. Inf.*, vol. 2, no. 1, pp. 15–21, 2016.

[2] S. A. Harjanto, M. Nurhaliza, and J. H. T. Sagala, "Optimalisasi Deteksi Anomali Untuk Pemfilteran Log dan Integrasi Dengan SIEM Menggunakan Machine Learning," *Madani J. Ilm. Multidisiplin*, vol. 2, no. 7, pp. 266–275, 2024.
[3] G. I. Sahhara, Windarto, T. Fatimah, and J. C. Chandra, "Implementasi Matching Rules Pada Sistem Pakar Web- Based Untuk Troubleshooting Jaringan Hotspot Universitas Budi Luhur," in *2nd Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, Jakarta: Fakultas Teknologi Informasi Universitas Budi Luhur, 2023, pp. 296–303.
[4] G. M. G. Bororing, "Pengembangan Algoritma Machine Learning Untuk Mendeteksi Anomali Dalam Jaringan Komputer," *J. Rev. Pendidik. dan Pengajaran*, vol. 7, no. 1, pp. 1361–1368, 2024.
[5] J. Chandra, H. Hermanto, and A. Rahman, "Deteksi Serangan Port Scanning Menggunakan Algoritma Naive Bayes," STMIK MDP, pp. 1–12, 2021. [Online]. Available: https://core.ac.uk/download/pdf/153523864.pdf
[6] M. I. Manalu and F. P. Hutabarat, "Network Traffic Anomaly Detection Using the Decision Tree Method," *J. Media Tek. Elektro dan Komput.*, vol. 01, no. 01, pp. 37–44, 2024.
[7] S. Anwar, F. Septian, and R. D. Septiana, "Klasifikasi Anomali Intrusion Detection System (IDS) Menggunakan Algoritma Naïve Bayes Classifier dan Correlation-Based Feature Selection," *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, pp. 135–140, 2019.
[8] R. R. Widalala, et al, "Dampak Penggunaan Artificial Intelligence pada Keamanan Siber: Sebuah Kajian Terhadap Potensi Keuntungan dan Ancaman," *J. Pembelajaran dan Pengemb. Diri*, vol. 4, no. 8, pp. 1541–1552, 2024.
[9] A. Shafiyyah, G. F. Nama, and R. A. Pradipta, "Implementasi Wazuh Menggunakan Metode Ppdioo Di Sistem Keamanan Jaringan Psdku Universitas Lampung Waykanan Sebagai Deteksi Dan Respon Serangan Siber," *J. Inform. dan Tek. dan Elektro Terap.*, vol. 12, no. 2, pp. 970–982, 2024.
[10] R. Aditya, Y. Muhyidin, and D. Singasatia, "Implementasi Security Information And Event Management (SIEM) Untuk Monitoring Keamanan Server Menggunakan Wazuh," *J. Ris. Sist. Inf. dan Tek. Inform.*, vol. 2, no. 5, pp. 137–144, 2024.
[11] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422.
[12] A. Zulfikar, F. A. Rahmani, and N. Azizah, "Deteksi Anomali Menggunakan Isolation Forest Belanja Barang Persediaan Konsumsi Pada Satuan Kerja Kepolisian Republik Indonesia," *J. Manaj. Perbendaharaan*, vol. 4, no. 1, pp. 1–15, 2023.
[13] H. Chaidir, A. G. Putrada, and M. Abdurohman, "Perbandingan Metode One Class SVM dan Isolation Forest Dalam Mendeteksi Anomali Dalam Activity Recognition Pada Rumah Dengan PIR Sensor," in *e-Proceeding of Engineering*, 2021, pp. 11078–11087.